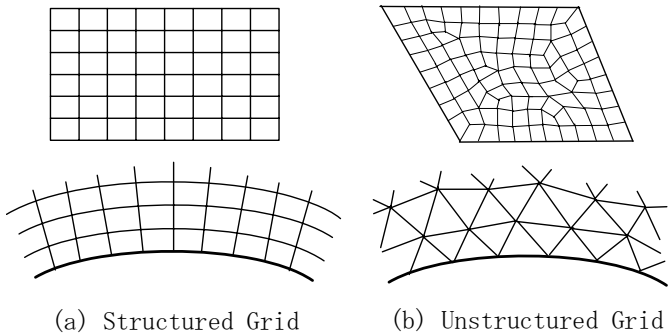
人工環境設計解析工学
メッシュ生成と
アダプティブリメッシング
（第４回）

東京大学
新領域創成科学研究科
鈴木克幸

構造メッシュと非構造メッシュ



(a) Structured Grid　(b) Unstructured Grid

マッピング法
バウンダリフィット法等

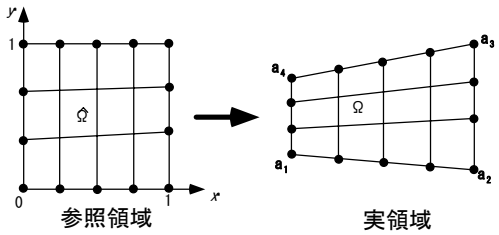
デラウニー法
アドバンシングフロント法等

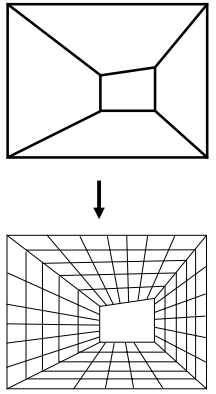
自動メッシュ生成技術

- 構造メッシュ生成（規則的に配置されたメッシュ）
  - マッピング法（マップドメッシュ）
  - バウンダリフィット法
- 非構造メッシュ生成
  - デラウニー法
  - 四分木、八分木法
  - フロント法
  - ペービング法
- 構造問題では非構造メッシュが主流。流体問題では構造メッシュが主流だが、非構造メッシュに移りつつある。

半自動メッシュ生成法

- マッピング法
- Transfinite マッピング法
- バウンダリフィット法

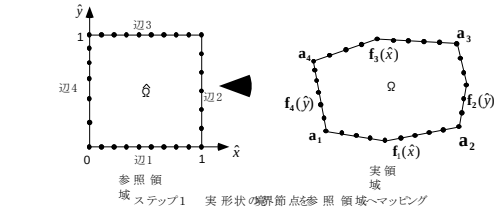




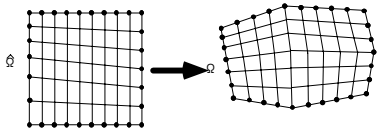
ブロック分割

Transfinite マッピング法

- マッピングを4角形以外の領域に拡張
- マッピング領域がゆがんでいるときはメッシュもゆがむ



ステップ1　実形状の境界節点を参照領域へマッピング



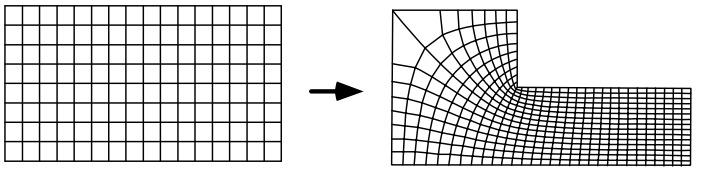
ステップ2　参照領域上でのメッシュ生成　　ステップ3　実領域へマッピング

バウンダリーフィット法

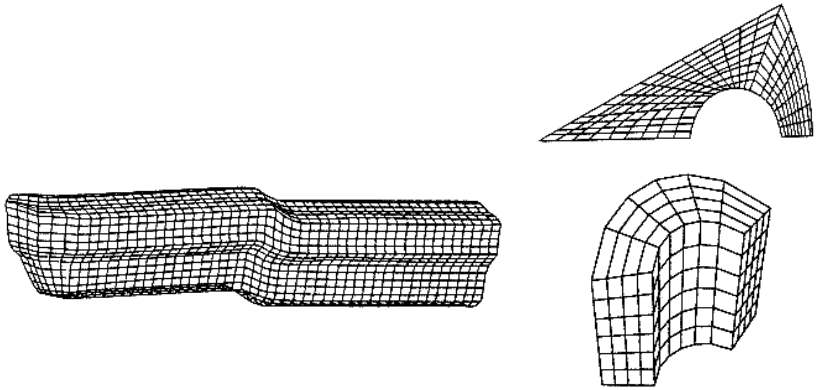
- 楕円型偏微分方程式を解く（Elliptic Mesh Generatorとも呼ばれる）
  - 非常になめらかな構造メッシュ
  - 凸部は疎に、凹部は密に

$$\xi_{xx} + \xi_{yy} = P$$

$$\eta_{xx} + \eta_{yy} = Q$$



マッピング法（2）

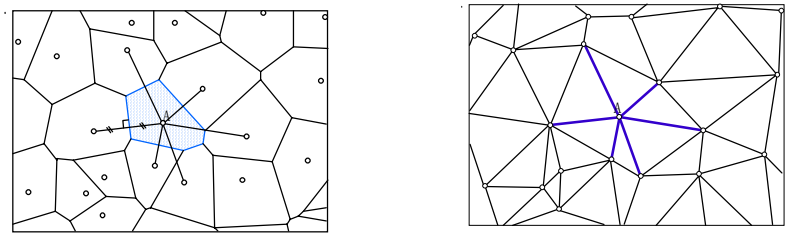


長所　良い形状の要素が多く生成される。

短所　対辺で節点数を等しくしなくてはならない等の制約があり、完全な自動化は行えず、複雑な形状を持つ領域にも適用できない。
要素の粗密化が難しい。

デラウニー法

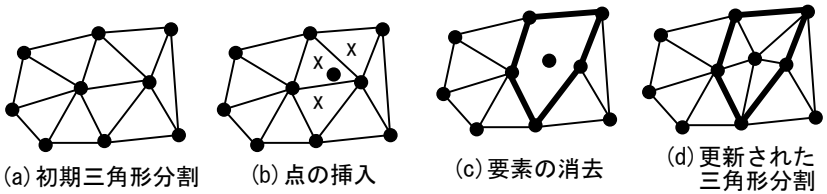
- ボロノイ分割に基づく
- 無限領域に対して解の存在が保証
- 有限要素メッシュとして質のよいメッシュ
- 3次元への拡張が容易

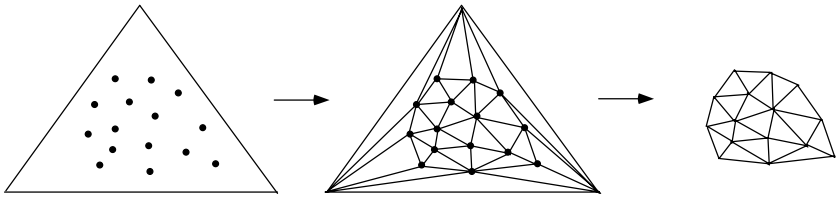


ボロノイ分割図　⟷　デラウニー三角形分割

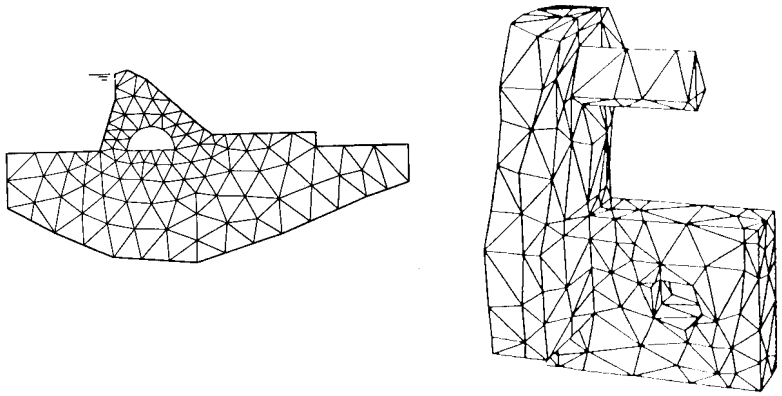
デラウニー法によるメッシュ生成法

- Watsonのアルゴリズム

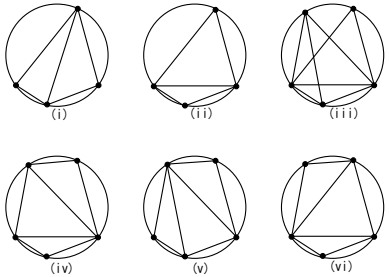




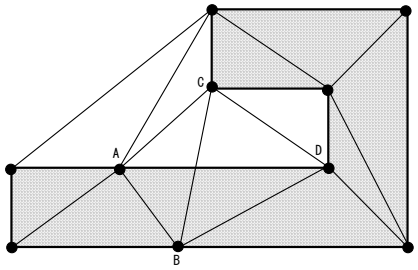
メッシュ生成例



デラウニー法の問題



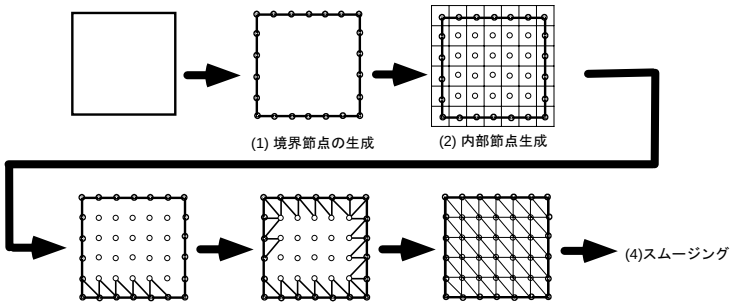
(a) Degeneracyが生じる例



(b) 境界と不整合が生じた例
(△ABC, △CBD)

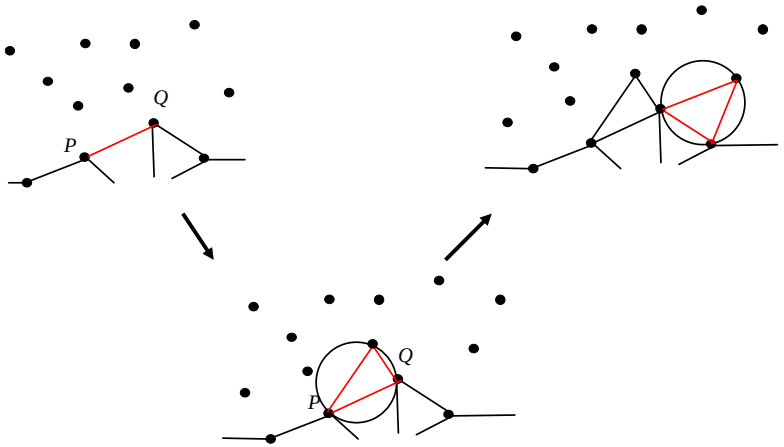
アドバンシングフロント法

- 境界から三角形(四面体)を生成
- 非凸形状に有利
- アルゴリズムは複雑

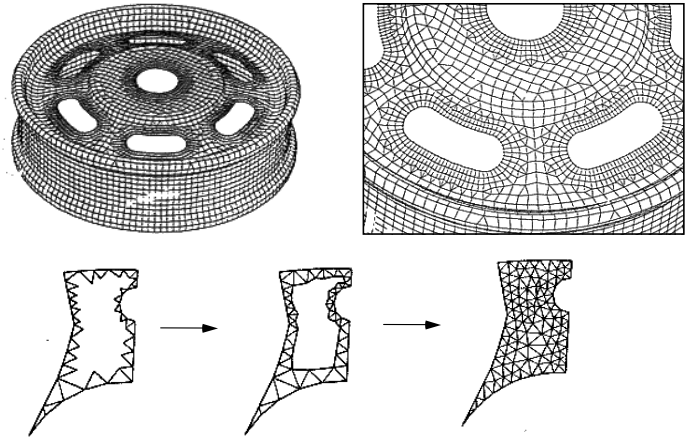


(3)フロント法による三角形要素の生成

アドバンシングフロント法での メッシュ生成



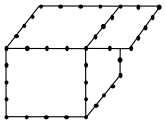
アドバンシングフロント法の例



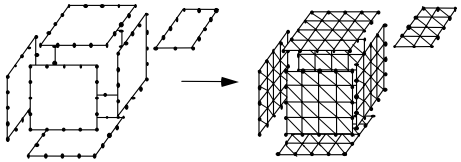
（アドバンシング）フロント法を用いたメッシュ生成

ユーザが要素の大きさの指標を与える

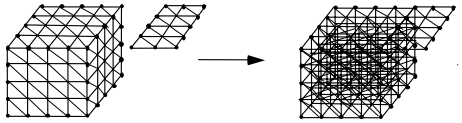
（１）境界節点発生

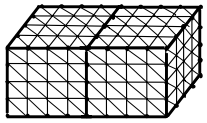


（２）面上でのメッシュ生成



（３）ソリッドのメッシュ生成





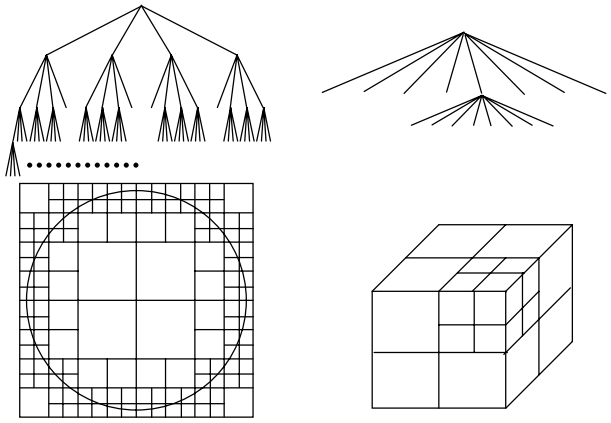
非多様体モデルや多連結の領域から成る問題において、
整合性の取れたメッシュ生成が容易

アドバンシングフロントvs デラウニー

- アドバンシングフロント法
  - 境界面近くでいいメッシュ
  - メッシュ形状の制御可能
  - 計算量が多い
- デラウニー法
  - 数学的ベース
  - 計算量小
  - 境界での不具合

4分木法、8分木法

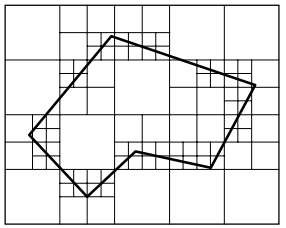
- 2次元、3次元



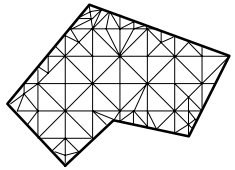
(a)　(b)

メッシュ生成例

4分木



メッシュ（スムージング前）



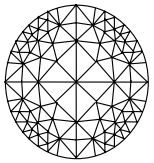
任意形状に対応
3次元への拡張が容易（8分木法）
要素形状があまりよくない
力学的な問題と別に粗密が生じる

四分木・八分木法（2）

（４）生成された、正方形のうち領域の外部にあるものは除去し、境界線上にある正方形については、境界に適合するように正方形ブロックの形状を修正する。

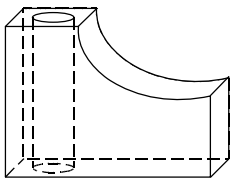
（５）適当な分割パターンを用いて、メッシュを生成する。

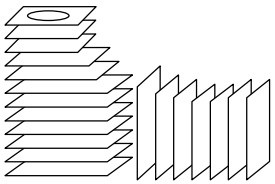
（６）スムージングを行いメッシュ全体の質を改善する。



スライスカット法（断面法）

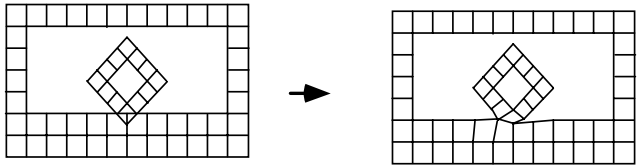
- 2.5次元体のメッシュ生成
- 完全な自動化は困難





ペービング法

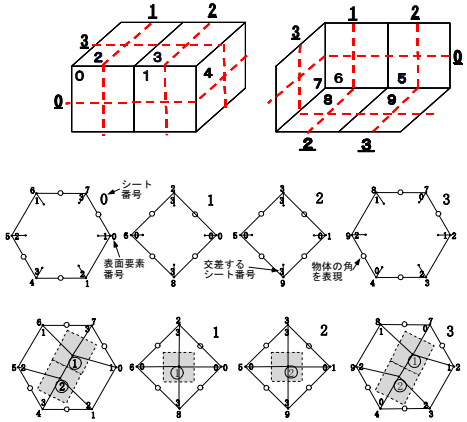
- 境界から順に4辺形メッシュを配置していく
- 矛盾が出たら修正する



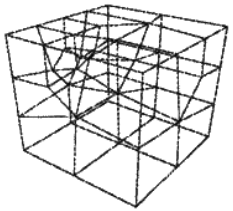
6面体メッシュ生成法

- Whisker Weaving法
  - 2次元のPaving法の3次元への拡張
- Grid Based法
  - 立体の内部に立方体のグリッドを配置
  - 表面を処理
- Medial Surfaceを用いた空間分割法
  - マッピングによる6面体生成が可能な基本領域に分割
  - すべての頂点において稜線が3本である必要

Whisker Weaving法

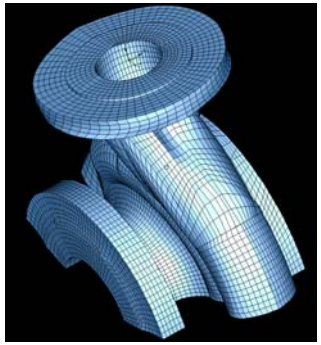


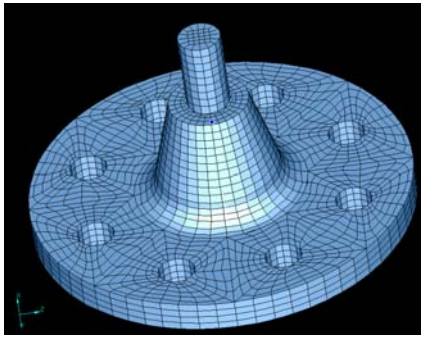
表面の四角形分割データから
内部に六面体を生成していく
STC空間上での分割
四角形分割によっては六面体に
切れない場合がある
現状では実用化レベルのものはない



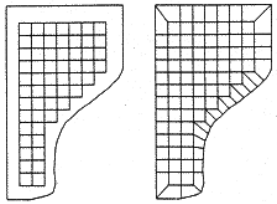
Whisker Weaving

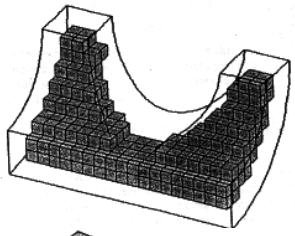
- 表面の四角形メッシュから、内部に六面体メッシュを作成していく
  （Muller-Hannemann）

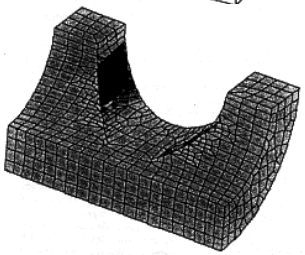




Grid Based法、オーバーレイ法







複雑な形状を有する問題には
適用できない。
市販されている物もある。

Medial Surfaceを用いた方法

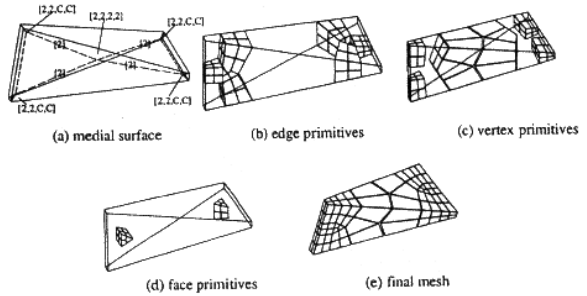
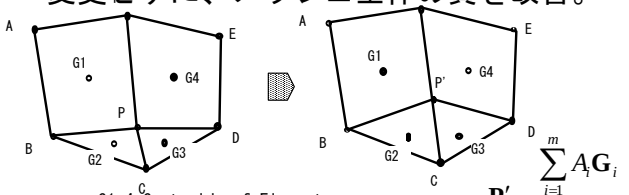


Figure 28. Example 1, showing stages in meshing the tapered brick

すべての頂点において稜線が3本である必要がある。
実用化されているものもある。

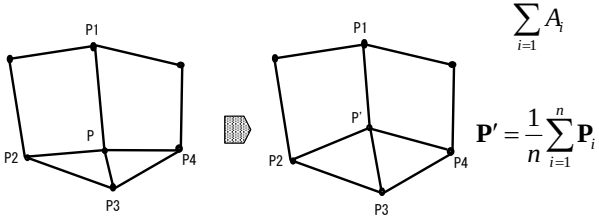
スムージング（Smoothing)

メッシュ生成が終了した後に、メッシュのトポロジーを
変更せずに、メッシュ全体の質を改善。



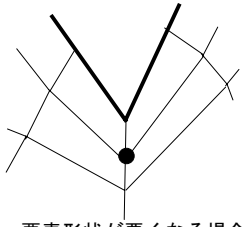
G1-4:Centroids of Elements

$$\mathbf{P}' = \frac{\sum_{i=1}^{m} A_i \mathbf{G}_i}{\sum_{i=1}^{m} A_i}$$



$$\mathbf{P}' = \frac{1}{n}\sum_{i=1}^{n} \mathbf{P}_i$$

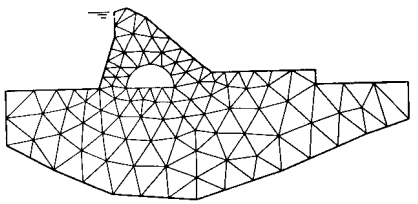
凹状の境界付近では、
注意が必要。

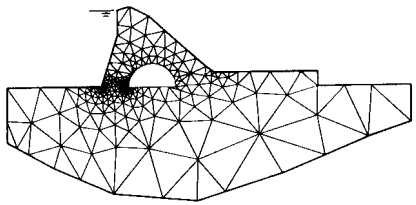


要素形状が悪くなる場合

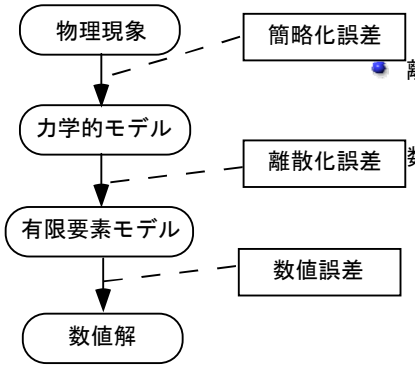
アダプティブ法とは

- 有限要素法の解の精度はメッシュに依存
  - 応力の急激に変化する場所は細かい要素に分割
- 「良い」メッシュの生成
  - 経験豊富なエンジニアが手で生成
  - 粗密付けの機能を持った自動メッシュジェネレータの利用
- アダプティブ法
  - 解析結果に基づき離散化誤差の分布を推定し、誤差を減らすようにメッシュを改善→解析専門家でなくても精度のよい解析が可能



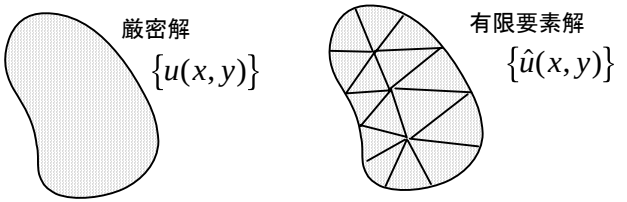


解析における誤差



- 簡略化誤差
  - 解析問題決定時のモデルの簡略化によって生じる誤差
  - 現状では定量的な評価は困難
- 離散化誤差
  - 連続体の有限要素離散による補間の不正確さによって生じる誤差
- 数値誤差
  - 数値積分の誤差は理論的背景がほぼ確立
  - 丸め誤差は、現在の計算精度ではあまり問題とならない

離散化誤差の計り方



誤差 $\{e(x,y)\} = \{u(x,y)\} - \{\hat{u}(x,y)\}$

ベクトル量のスカラー化（エネルギーノルム）

$$\|\{e\}\| = \int_{\Omega} \nabla\{u\}^T [D] \nabla\{u\} d\Omega$$

要素ごとの解析を行うとき、剛体運動成分は正確に求めることができない

事前誤差評価と事後誤差評価

- 事前誤差評価 (A-Priori Error Estimation)
  - 有限要素解析を行う前に、メッシュの代表長さhに基づいて誤差評価を行う
  - オーダーレベルの誤差評価のみ
  - 誤差の分布はわからない

$$\|\{u\} - \{\hat{u}\}\|_m \le Ch^{k-m+1} \|\{u\}\|_{k+1}$$

- 事後誤差評価 (A-Posteriori Error Estimation)
  - 有限要素解析の解を利用して誤差解析を行う
  - 定量的な誤差の評価が可能
  - 誤差の領域内での分布がわかる

事後誤差評価法の種類

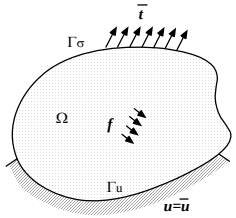
- 応力場のスムージングによるもの
  - 応力の不連続量を評価
    - Zienkiwicz, et. al.
- 残差によるもの
  - 基礎方程式に基づく
    - Oden, et. al.
    - 大坪、北村 et. al.
- サブドメインで残差を評価するもの
  - Babaska, et al.
- 双対法によって誤差の上下限を求めるもの
- etc.

残差に基づく方法

厳密解：釣合い方程式、境界条件を完全に満足する

$$\nabla\cdot[\sigma]+\{f\}=\{0\}$$

領域内で、力は常に釣り合う



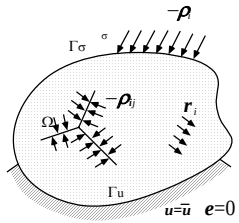
有限要素解：基礎方程式に残差が生じる

$$\nabla\cdot[\hat{\sigma}]+\{f\}=\{r\}$$

要素間、境界の反力に不釣合いが生じる

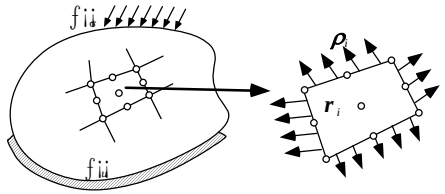
$$\{t\}_i+\{t\}_j=\{\rho\}_{ij}$$

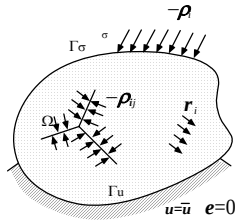
$$\{t\}_i-\{\bar{t}\}=\{\rho\}_{i\sigma}$$

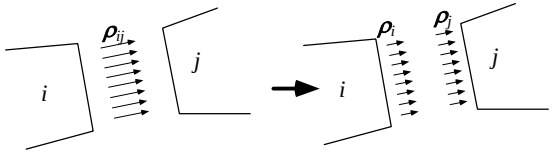


残差に基づく誤差の計算

- 誤差の満足すべき方程式を導く
- 要素ごとの解析

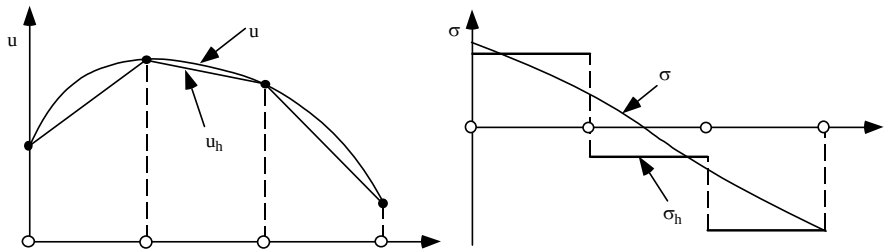






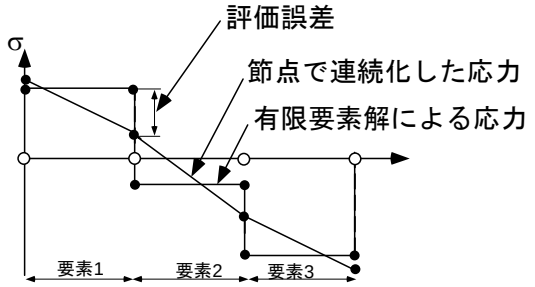
変位と応力の連続性

- 2階の微分方程式
  - 1階微分（応力、歪）は連続



スムージングによる方法

- 1次の近似関数を用いた場合、変位は連続であるが応力は不連続になる。



スムージングした応力

$$\sigma^* = N\overline{\sigma}^*$$

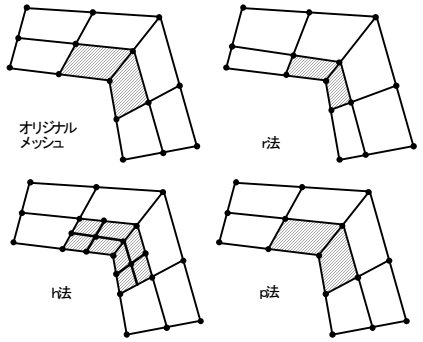
$$\int_\Omega N^T(\sigma^* - \sigma_h)d\Omega = 0$$

$$\overline{\sigma}^* = A^{-1}\left(\int_\Omega N^T D\nabla^* N\tilde{u}_h d\Omega\right)$$

$$A = \int_\Omega N^T N d\Omega$$

アダプティブ法の種類

- r法：節点を移動し、誤差の大きい領域の要素寸法を小さくする
- h法：要素を細分割し、要素寸法を小さくする
- p法：近似関数の次数を上げる



アダプティブ法の指針1

$$\min f(h) = \sum_{i=1}^{N_{el}} \|e(h)\|_i^2$$

subject to

$$g(h) = \int_\Omega n(h)d\Omega - \overline{N} \le 0$$

Lagrange乗数法

$$L = f(h) + \lambda g(h)$$

$$-\left(\frac{\partial f(h)}{\partial h} \Big/ \frac{\partial g(h)}{\partial h}\right) = \lambda$$

アダプティブ法の指針2

$$-\left(\frac{\partial f(h)}{\partial h} \Big/ \frac{\partial g(h)}{\partial h}\right) = \lambda$$

$$f(h) = \overline{m}h^\sigma$$

$$g(h) = \overline{n}h^{-2}$$

$$-\frac{\sigma}{2}\frac{\overline{m}h^{\sigma-1}}{\overline{n}h^{-3}} = -\frac{\sigma}{2}\frac{\overline{m}h^{\sigma}}{\overline{n}h^{-2}} = \lambda$$ 領域内で一定

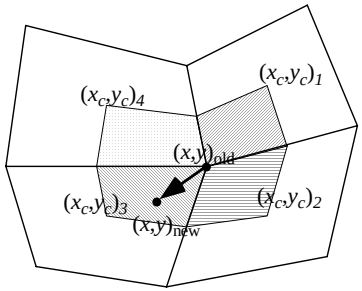
$$-\int_{\Omega_i} \overline{m}h^\sigma d\Omega = \frac{2\lambda}{\sigma}\int_{\Omega_i} \overline{n}h^{-2}d\Omega = \mu$$ 各要素で一定

r-アダプティブ法

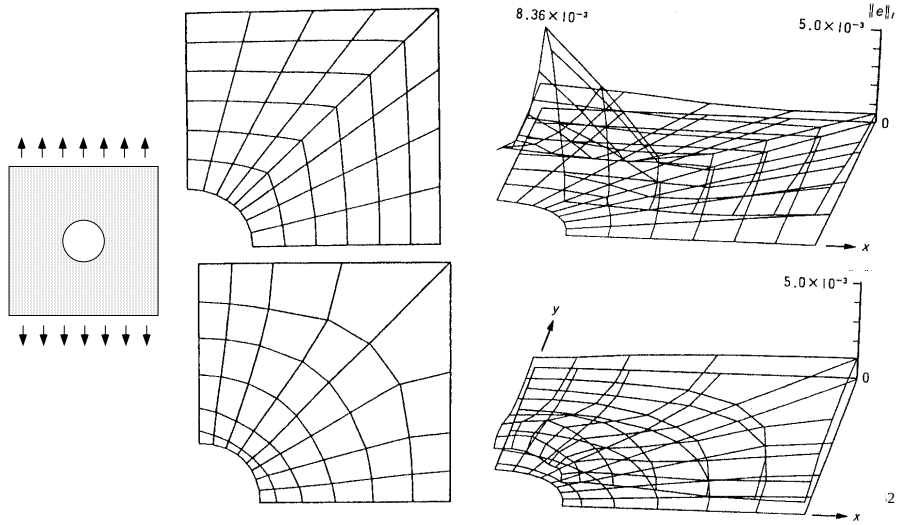
- まわりの要素の誤差に応じて節点を移動
- メッシュのコネクティビティーは変化なし
- メッシュのゆがみによる問題→大きな移動は難しい
- 目的とする精度を指定することは難しい

$$x_n = \frac{\sum x_i^C \left(\|\boldsymbol{e}\|_i / A_i\right)}{\sum \left(\|\boldsymbol{e}\|_i / A_i\right)}$$

$$y_n = \frac{\sum y_i^C \left(\|\boldsymbol{e}\|_i / A_i\right)}{\sum \left(\|\boldsymbol{e}\|_i / A_i\right)}$$

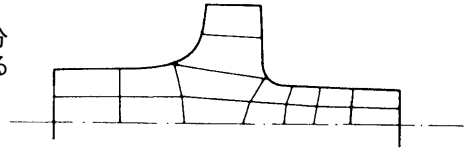


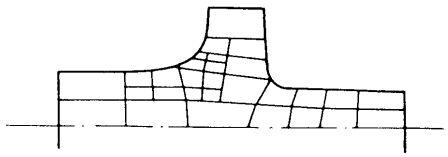
r-アダプティブ法の例



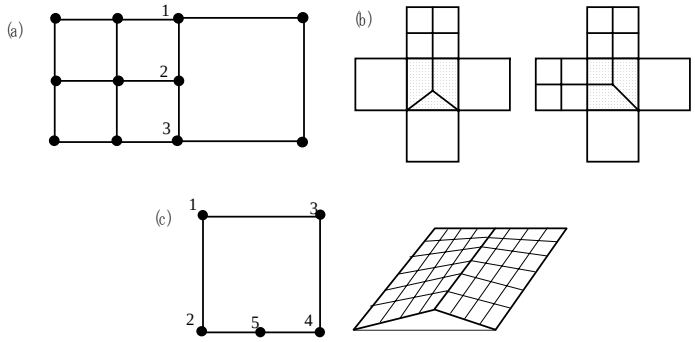
h-アダプティブ法

- 誤差の大きい要素を細分割
- 精度を指定することによってのコントロール可
- 要素の細分化する部分としない部分の境界で、特殊な処理が必要になる

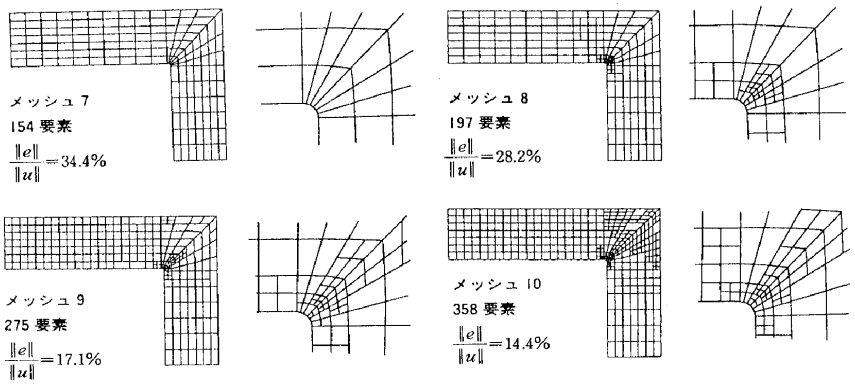

Original mesh


h Refinement (subdivision)

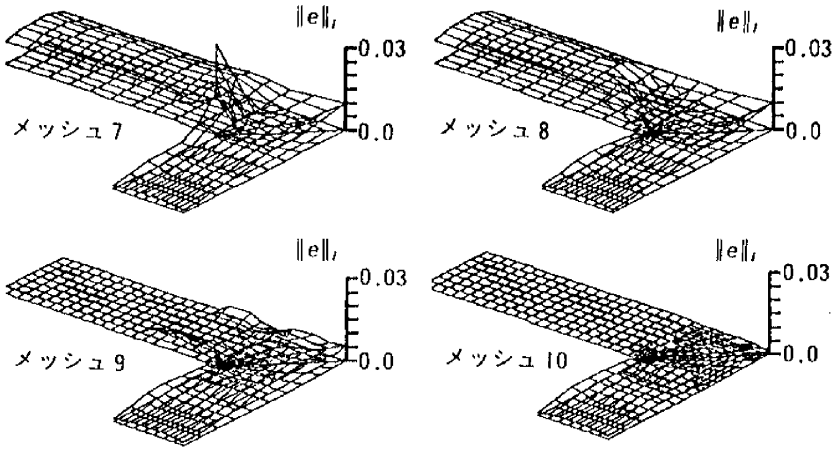
h-アダプティブ法の形状関数の連続性



h-アダプティブ法の例

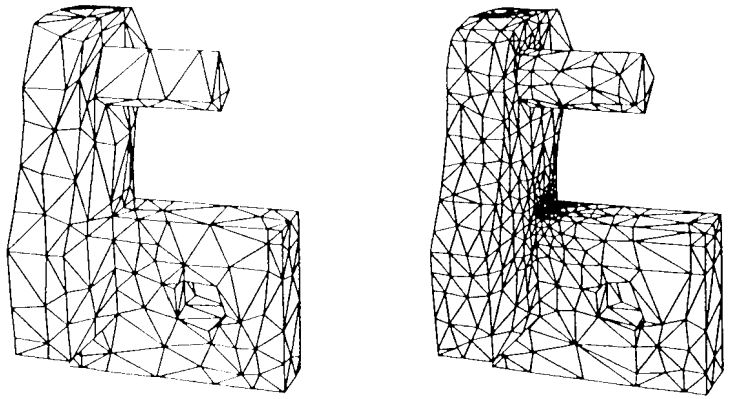


h-アダプティブ法の例(誤差分布)



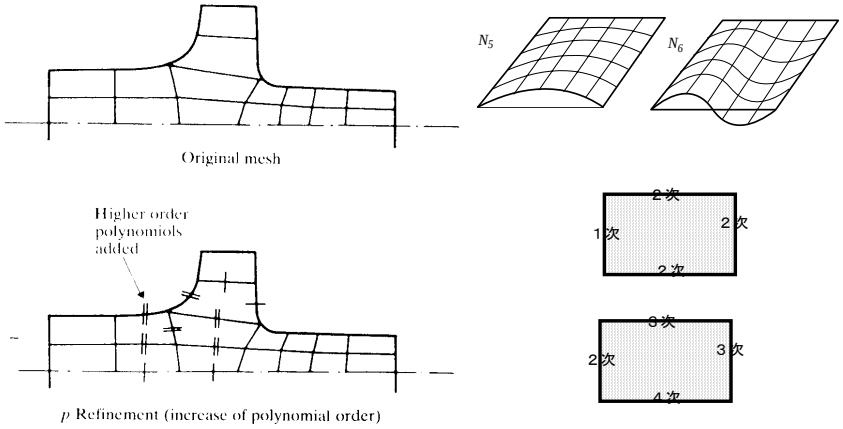
切り直しを行うh法

- メッシュの切り直しを行う

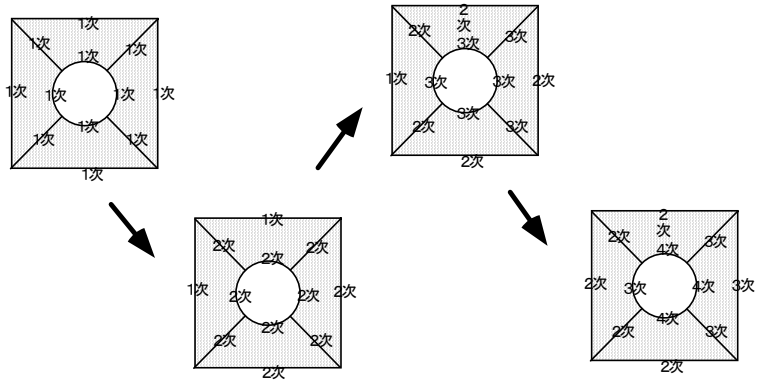


p-アダプティブ法

- 要素内の誤差が大きい要素の辺の形状関数の次数を上げる



p-アダプティブ法のストラテジー



各方法の比較

	r 法	h 法	P 法
長所	・要素数、節点数一定 ・比較的少ない自由度でよい精度	・複雑な形状への適用が容易 ・応力集中に対する精度良	・単純な入力データ ・少ない要素数で程々の精度
短所	・境界での節点移動が複雑 ・比較的少ない自由度でよい精度	・要素数の急激な増加 ・中間節点の発生	・急激な応力集中などに対する精度が悪い