エージェントによる FEM プリプロセッシング*

西 恭一^{*1}, 邊 吾一^{*1}, 青山 敬三^{*2}

FEM Preprocessor by Using Agent

Yasukazu NISHI, Goichi BEN (O-Il BYON) and Keizo AOYAMA

This paper proposes one of automatic mesh generation method in preprocessing of a FEM by using an agent. In this paper, the 'agent' is defined as a kind of A-life (Artificial Life) system and it has three kinds of functions, 'split', 'deform' and 'mutate' (swap for diagonals). When FEM users give a number of mesh division, the agent repeatedly divides analytical region by using the function of 'split' and it changes the elements in the region to ideal shapes and areas by using other two functions. After this method is applied to a trapezoid model and its effectiveness is confirmed, several kinds models (a square and a hexagonal array models of microscopic composite material and a mode I model in fracture mechanics) are automatically divided to the fine shapes nearly equaled an equilateral triangle. These examples consequently show that the proposed automatic mesh generation method by using the agent is a quite useful and a suitable for FEM analyses.

Key Words: Finite Element Method, Automatic Mesh Generation, Agent, Artificial-Life, Emergence

1. 緒 言

工学分野で幅広く用いられている有限要素法(FEM)は、線形のみならず非線形・非定常問題や最適化問題など、従来の解析的手法では扱えなかった問題についても応用されており、最も有効な数値解析法として確立されている。しかし、FEMの多様化に伴う解析対象の複雑化・大規模化は、FEMのメッシュ生成に費やす労力の加速度的な増加を招き、人為的負荷の軽減と時間短縮が必要となっている。そこで、これまでにさまざまな自動メッシュ生成法が研究・開発され⁽¹⁾⁽²⁾、汎用FEMシステムの一部として市販もされているが、制限事項が多く、さらに、熟練ユーザが持つ解析精度向上のための知識を与えられないため、解析対象に適したメッシュ生成は困難である。

そこで本研究では、FEMメッシュ生成法に人工生命(A Life)の概念を取り入れ、細胞分裂のように自己増殖を起こしながらFEM要素を派生し、解析対象に適した理想的なメッシュ生成を要素自身に考えさせる

方法を試みた。

2. 人 工 生 命

人工生命とは、単純な動作メカニズムを持つ個体が多数集まり集団を形成した際に、集団として複雑かつ多様な挙動が発現する現象(創発: Emergence)について解析を行い、工学的に有用なシステムとして用いる研究の総称である⁽³⁾。

図1に示すように、集団内における一つの人工生命体をエージェントと呼び、次のように定義する。

「エージェントは置かれた環境(領域)に対して、知識として蓄えているルールに従い自律的に行動を選択し実行する。また、同じ領域内にエージェントが複数存在する場合、各エージェントは互いに独立に存在し、全エージェントの行動を決定するグローバルなルール(神)は存在しない。したがって、結果として得られる全体的な挙動は創発的となる。」

3. エージェントの仕様

3.1 概念 自動メッシュ生成法として広く用いられているデローニー三角分割法⁽⁴⁾では、節点を基準に要素を生成するため(図2)、解析対象に対する全節点データをユーザ自身が入力しなければならず、また、

* 原稿受付 1997年5月19日。

^{*1} 正員、日本大学生産工学部 (〒275-8575 習志野市泉町1-2-1)。

^{*2} 日本大学大学院 [現: (株)東芝]。

要素の形状・大きさは節点の配置状態に依存してしまう。これに対し、本手法では要素を基準とし、副産物として節点を発生させることを試みた。図3に示すように、はじめに、解析対象全体を一つのエージェントとし、そのエージェントに外部および内部境界等の環境情報を持たせる。そして、刻々と変化する自己の評価値を基に、自身を目的とする要素分割数まで分裂増殖させることで、新たに同じ機能を持った要素、すなわちエージェントを派生させるとともに節点を発生させ、解析境界内全体を分割する。これにより、本手法におけるユーザ入力値は、解析境界および目的要素分割数のみとなる。

3.2 保有データ エージェントは、メッシュ生成の条件および進行状況を把握するためのグローバル情報として、以下を保有する。

- (1) 目的要素分割数：ユーザが望む要素分割数。
- (2) 現在の全エージェント数：メッシュ生成開始から時刻 t 経過後の要素数。
- (3) 解析境界番号：解析対象が複数の外部境界から構成されている場合、本システムでは個々の外部境界ごとにエージェントを定義するため、その指標として用いており、メッシュ生成開始時に自動的に(1～)に割り振られる。
- (4) 平均面積：解析境界面積を目的要素分割数で割った値であるが、解析対象の形状が複雑になると、解析境界内面積の算出は困難となる。そこで、ある程度メッシュ生成が進み、全エージェントが四角形、または、三角形要素に分割された時点で、個々のエージェント面積の合計から解析境界内面積を求め、平均面積を計算している。

Fig. 1 Agent

Fig. 2 Delaunay triangulation

- (5) 目的要素形状：ユーザが望む要素の形状であるが、現在は3(三角形)に固定されている。

また、各エージェントにユニークなローカル情報としては以下の情報を保有させた。

- (1) 要素番号：自己の要素番号。
- (2) 世代数：メッシュ生成開始時(解析対象全体が一つのエージェントであるとき)を世代数1とし、分裂するごとに世代数は一つインクリメントされ、受け継がれる。
- (3) 親エージェントの要素番号：自己の一世代前のエージェントの要素番号。
- (4) 理想面積：自己面積の目標値(=平均面積)。
- (5) 節点番号と座標：自己が保有する全節点情報。
- (6) 各節点に接している他のエージェントの要素番号および接する境界線に関する事項：自己が保有する節点と同数組の情報。各節点に接する要素数や境界線数は刻々と変化するため、可変長の配列としている。
- (7) エンティティデータ：自己の節点間を結ぶ線の種類(直線:0, 円弧:1など)。節点と同数の情報。

これらのローカル情報を持つことにより、エージェントはあらゆる二次元幾何学形状を認識できる。

3.3 評価関数 エージェントが自己の動作を決定するための指標として、式(1)に示すように、各節点の内角の積から自己の形状が正三角形時に最大値1となる形状評価値(V_s)

$$V_s = (3 \text{ 頂角の積 } [\text{rad}]^3) / (\pi/3)^3 \dots\dots\dots (1)$$

そして、 R =自己の面積/理想面積(平均面積)を求め、式(2)に示すように自己の面積が理想面積と等しい場合に最大値 ZERO となる面積評価値(V_a)

$$V_a = -(R-1) \quad [R \geq 1 \text{ のとき}] \dots\dots\dots (2)$$

$$V_a = -(1/R-1) \quad [R < 1 \text{ のとき}]$$

を評価関数として保有させた。なお、自己の形状が内部境界を含む場合や四角形以上の多角形である場合の形状評価値は、最小値 ZERO としている。

3.4 機能概要 エージェントは、選択可能な行動として、表1に示すように、“Split”、“Mutate”、“Deform”の三つの機能を有する。

“Split”機能は、自己を分裂し、任意形状の解析境界を理想面積に近い三角形要素に分割する機能である。

Fig. 3 Mesh generation by using agent

なお、自己の形状により分裂方法は異なり、

(1) 三角形の場合：分裂開始点となる節点(基準点)とその対辺 L の中点 M (表 1 の 1)

(2) 四角形以上の場合：基準点とその節点から最も遠い節点(四角形の場合は対角をなす点となる：表 1 の 1)

を境界として二つのエージェントに分裂させる。なお、 L が既に他のエージェント(以下 A_o)の辺である場合、 M が発生することにより、 A_o にとっては要素の辺上に節点が作られるため、実際の FEM 計算に用いることができなくなる。そこで、 M および M と対角をなす A_o の節点を境界線として、 A_o を強制的に分裂させることとした。

また、内角の大きい節点を優先して基準点とするが(内角の大きさが同じである節点が複数存在する場合は、その中からランダムに選択)、解析対象に内部境界を含む場合は、内部境界上の節点をさらに基準点として優先させる。

“Mutate”機能は、辺を共有する二つのエージェント同士が互いに通信して情報交換し、 V_s が高くなる場合に限り、共有辺を交換する機能である(表 1 の 2)。これにより、FEM 解析に適したメッシュ生成が完了する間に実行される“Deform”機能(以下参照)の回数を大幅に削減することが可能となった。

“Deform”機能は、 V_s および V_a が高くなる方向へ節点を移動させ、自己エージェントを理想面積の正三角形に近づける機能である。エージェントが保有する節点を A, B, C とすると、“Deform”機能を実行する手順は以下のとおりになる(図 4)。

Table 1 Functions

	Functions	Examples
1	Split	
2	Mutate (swap a diagonal line)	
3	Deform

Fig. 4 Deform function

①移動節点 A をランダムに選択する。

②点 A 側に、辺 BC を基準とした正三角形が作成できる点 D を求める。

③点 B, C は固定のまま、点 D から辺 BC への垂線によるベクトル MD 上で、面積が理想面積となる三角形 EBC の点 E を求める。

④ベクトル AE より、 αAE となる点 F を求める。ただし点 F が、解析境界線または点 A を共有している他のエージェントの辺を越える場合、“Deform”機能を中止する。

⑤点 A を点 F に移動する。

⑥点 A を共有している他のエージェントへ移動先情報を送り、節点データを更新させる(点 A を移動させる)。

ここで、 α の値を決定しなければならない。 α を 0.05~0.75 まで変化させたときの全エージェントの平均 V_s および V_a への影響を調べた結果、 $\alpha=0.1\sim0.4$ で V_s, V_a ともに高い値を得られるため、本研究では $\alpha=0.1$ を用いることとした。

3・5 動作決定アルゴリズム 3・2~3・4 節で述べたデータや機能を持つ個々のエージェントに、できるだけ簡単なアルゴリズムでユーザが求める理想的なメッシュ生成を行わせるため、メッシュ生成開始とともに解析境界全体を一つのエージェントに設定し、次のルールに従って行動させた。なお、エージェントの動作をモニタリングするため、メッシュ生成開始時(エージェントの派生時)を ZERO とする LOOP を定義し、境界内に存在するすべてのエージェントが“Split”, “Mutate”, “Deform”機能のいずれかを実行し終えた段階で、LOOP を一つインクリメントさせている。以下に、エージェントの動作の流れを記す。

①解析境界に円(弧)が含まれる場合、エージェントの派生とともに円(弧)を自動的に正多角形に変換する。解析対象や解析条件に応じて円弧の分割数を決定することが理想であるが、現在はデフォルト値として 30° ごと(変更可)に節点を発生させ、等分割している。

②以下の条件に当てはまる場合、“Split”機能により自己分裂し、新たに同じ機能を持つ別のエージェントを派生する。

- ・内部境界を保有している
- ・自己の形状が四角形以上の多角形
- ・解析境界内のエージェント数がユーザ入力による目的要素分割数より少なく、かつ、自己の V_a が一定値 $\beta (= -0.7 \text{ Const.})$ より小さい
- ・自己の形状が三角形であり、2 節点と同じ解析境界辺上に存在し、かつ、その 2 節点間の距離が理想面

積と等しい正三角形における一辺の長さの係数 γ ($=1.5 \text{ Const.}$) 倍以上長い

ここまでにより、メッシュ生成開始から初期の段階では、エージェントは、“Split”機能を繰り返し実行することで自己増殖し、解析境界内を分割していく。

③②の条件に当てはまらない場合、自己が保有する最も長い辺を共有する他のエージェントと通信を行い(共有するエージェントが存在しない場合は→④)、互いの辺を入れ替えることで V_s が良くなるのであれば、“Mutate”機能を実行し、共有辺を交換する。

④②および③の条件に当てはまらない場合、“Deform”機能を用いてランダムに決定した節点を V_s および V_a が高くなる方向へ移動させる。

以上のエージェントによる自動メッシュ生成システムの構築には、Sun Microsystems 社製 SPARCstation10 において、インタプリタ型言語の Tcl/Tk⁽⁵⁾ および Tcl/DP⁽⁶⁾(フリーソフトウェア)を独自に改良して用いた。

3・6 アルゴリズムの検証⁽⁷⁾ アルゴリズムの評価対象として、図5に示す三つの正三角形から構成される台形モデルを用い、目的要素分割数は、解析境界内全体を幾何学的に正三角形によって分割可能な48に設定し、メッシュ生成を実行した。この結果得られた全エージェントの平均 V_s および V_a の推移を図6

Fig. 5 Trapezoid model

Fig. 6 Variations of evaluation value

Fig. 7 Variation of mesh shape

に示し、メッシュ形状の変化を図7に示す。なお、図6の横軸は LOOP、縦軸は各 LOOP において存在する全エージェントの V_s と V_a の平均値としている。本手法によるメッシュ生成はアルゴリズムの性質上創発的な結果となるが、LOOPを重ねることで各エージェントの評価値は上がり、最終的には、LOOP 数 35 で分割数 48 の正三角形によってメッシュ生成されており、本アルゴリズムの有効性が明らかとなった。なお、LOOP 数が 35 となるまでに要したエージェント動作時間は約 19 min であり、今後、動作速度の向上を図る必要がある。

4. 適用例

4・1 微視的複合材料モデル⁽⁸⁾ 内部に円弧境界を含む解析対象例として、複合材料の強度や破壊挙動を精度良く解析するために用いられる微視的モデルを用いた。

解析対象のモデル化は、図8(a)、(b)のように、2種類の二次元複合材料断面モデル(正方配列および六方配列モデル)について行った。表2に示す分割条件とするため、母材と繊維で要素面積が大きく異なり、

Table 2 Conditions of mesh generation

	Square array		Hexagonal array	
	Matrix	Fiber	Matrix	Fiber
Num. of division	200	100	250	70
LOOP number	60	60	60	60

(a) Square array model (b) Hexagonal array model

Fig. 8 Micro model of composite materials

(a) Square array model (b) Hexagonal array model

Fig. 9 Results of mesh generation

一般的には境界近傍の要素形状にひずみを起こしやすい。ここでは、LOOP=0 の段階でエージェントを独立に母材と繊維に定義し、並列に動作を開始させた。正方配列モデル、六方配列モデルにおける結果をそれぞれ図 9(a), (b) に示す。両モデルとも境界内が正三角形に近い三角形要素によって平均的に分割されていることが確認できる。このときの全エージェントの平均 V_s を表 3 に示すが、 V_s が 0.844 のときは直角二等辺三角形になることから、両モデルとも正三角形に近いメッシュが生成させていることがわかる。

4・2 引張試験片モデル⁽⁹⁾ 内部に円孔を含む解析対象の例として、破壊力学のモード I 用の試験片⁽¹⁰⁾ (図 10) について、目的要素分割数=250 の条件下でメッシュ生成を行った。

メッシュが生成、増殖する様子を図 11 に示す。図 11 より本エージェントは自己形状を認識しているため、内部境界の内側を避けてメッシュが生成されていくことがわかる。また、初めは形状の悪い三角形要素によって分割されているが LOOP を重ねることで、正三角形に近い要素によって平均的に分割されていくことが確認できる。なお、LOOP=16 で $V_s=0.911$ 、 $V_a=0.003$ となり、高い評価値を得ている。これにより、本研究で提案したエージェントによる方法は、内部境界を含む複雑な形状に対しても、目的要素数で形状の良い三角形によるメッシュ生成が行える。

5. エージェントの機能拡張

5・1 応力集中点の考慮 FEM 解析では、一般にメッシュサイズが小さくなるほど解析精度は高くなる。しかし、メッシュサイズを小さくすることは、解

析境界内のメッシュ数を増やすことになり、自由度が増し、解析に要するメモリー量および計算時間が増大する。したがって、FEM 解析における計算を精度良く、しかも速く行うためには、解析条件に応じて精度が要求される箇所(応力集中点等)だけを細かくする自動メッシュ生成がなされるべきである。

本研究では、エージェントに解析条件を考慮した理想的なメッシュ生成を自動的に考えさせ、行動させることを目標としているが、その初期段階として、ここでは応力集中点をユーザに入力させ、正三角形に近い形状を維持しながら、応力集中点近傍には細かい要素分割が行える拡張機能をエージェントに付加し、行動させることを試みた。

5・2 アルゴリズム 個々のエージェントの面積を変化させるためには、前述のアルゴリズムの理想面積値を操作すればよい。そこで、ユーザ入力による応力集中点(A)とエージェントの距離に応じて、存在する n 個のエージェントに対する理想面積(S_i ; $i=1 \sim n$)を次式により設定した。

$$S_i = (L_i / (\eta \cdot L_{\max}))^2 \times S'$$

S' : 平均面積(=解析境界内面積/目的要素分割数)

$L_{\max}$: 点 A と点 A から最も遠い節点間距離

L_i : 点 A とエージェントの図心間距離

η : 係数(=0.2 Const.)

すなわち、応力集中点を考慮した場合、応力集中点とエージェントの図心間距離の二乗に反比例して理想面積が小さくなるとした。ただし、解析条件に対してエージェントの面積が必要以上に大きく、または、小さくなりすぎることを防ぐため、 S_i の限界値として最

Table 3 Average values of V_s after mesh generation

	Evaluation value	
Square array	0.924	=1.00
Hexagonal array	0.915	=0.844

Fig. 10 Mode I model in fracture mechanics

Fig. 11 Process of mesh generation

Fig. 12 Process of mesh generation in the model with stress concentration

大値は S' の 3 倍, 最小値は S' の $1/30$ とした。なお, 応力集中点を考慮しない場合は, $S_i = S'$ となる。

5.3 適用例 図 12 に, 図 10 の試験片モデルを用い, 図中 A 点を応力集中点とした場合のメッシュ生成過程例を示す。図 12 から明らかなように, A 点に対しては細かいメッシュが集中しており, さらに, 全体的には形状の良い三角形でメッシュ生成されていることが確認できる。そこで, 応力集中点を考慮しない場合と比較した形状評価値 V_s の推移を図 13 に示す。図 13 より, LOOP の増加とともに V_s は高くなり, 応力集中を考慮した場合は, 応力集中点方向に小さい要素が集まるため面積的制限が起こるにもかかわらず, 考慮しない場合に比べても若干 V_s が低くなる程度で収まっている。

6. 結 言

本報では, 解析対象の複雑化・大規模化に伴う FEM メッシュ生成の人為的負荷の軽減と時間短縮が求められていることから, 人工生命の概念を取り入れた新しい FEM メッシュ生成法について述べた。また, アルゴリズムの評価を行い, 微視的複合材料モデルと引張試験片モデルの二次元モデルに適用した。さらに, 応力集中点の付加により, 大きさが連続的に変化するメッシュ生成を試みた結果, 以下のことが得られた。

- (1) 本エージェントに付加したアルゴリズムにより, FEM 自動メッシュ生成は可能である。
- (2) 解析対象の外形がわかれば自動メッシュ生成は行えるため, ユーザへの負荷を激減することができる。

Fig. 13 Variations of evaluation value

(3) 応力集中点を考慮した場合を含め, FEM 解析における精度と計算時間を考慮したメッシュ生成(形状は正三角形に近く, しかも, 大きさは妥当)が行える。

(4) メッシュ生成における知識は, 一つのエージェントのみの動作について考えればよい。機能の拡張が容易である。今後の課題として,

(1) 現在エージェントは Tcl 言語で記述されているため動作が遅く, さらに, 行動はエージェントごとのシーケンシャルな処理となっている。生命現象に見られる真の創発的な行動をさせるためには, 並列処理に適した言語にて構築し直す必要がある。そして, 大規模問題への適用を行う。

(2) 四角形要素や三次元要素の高次要素へ適用する。

(3) メッシュ生成時にイタレーションが起こるが, その終了をエージェントの評価値などを基に自動的に判断させるアルゴリズムを検討する必要がある。

文 献

- (1) 川村恭己, 日本計算工学会第 1 回講演会教材, (1996), 2.
- (2) 矢川元基・山田知典・河合浩志, 機講論, No. 95-1(1995-3), 210-211.
- (3) 中村真理・車谷浩一, 人工知能学会誌, 10-6(1995), 39-47.
- (4) 谷口健男, FEM のための要素自動分割, (1992), 9-27, 森北出版.
- (5) Ousterhout, J. K. (U. C. at Berkeley), *An Introduction to Tcl and Tk*, (1993), 1-320, Addison Wesley.
- (6) Rowe, L. A., Smith, B. C. (U. C. at Berkeley), *Tcl Distributed Programming*, (1993), 1-3, U. C. B.
- (7) 西恭一, 日本ファジィ学会, 日本計算工学会 A-Life, Fuzzy の新たな展開講演会講演集, (1996-12), 1-14.
- (8) 遠哲一・西恭一・青山敬三, 機講論, No. 96-15(1996-4), 63-64.
- (9) 遠哲一・西恭一・青山敬三, 機講論, No. 96-25(1996-9), 45-46.
- (10) 鷲津久一郎・宮本博, 有限要素法ハンドブック II 応用編, (1983), 210-212, 培風館.