ハイパフォーマンス
コンピューティング 55−5
（1995. 3. 9）

輸送計画を用いたFEM並列要素分割法

河野 洋一　西松 研　福盛 秀雄　村岡 洋一
早稲田大学

本研究では、FEM(有限要素法)の三角形要素分割を並列計算機上で行なう手法を提案する。従来の並列分割法である<粗分割-細分割>法は、粗分割においてPE数に比例した三角形数を逐次に生成する必要があり、高い並列化率を達成できないという問題がある。そこで本研究では粗分割を並列化する手法を提案する。本手法では、始めに最低1個の三角形を各PEに割り当て、各PEで並列に粗分割を行なう。つぎに輸送計画にしたがって各PEの担当領域を変更し、負荷分散を行なう。そして最後に細分割を並列に行なう。本手法を富士通の並列計算機AP1000に実装した結果、PE32台のとき、逐次処理に比べて約15倍のスピードアップを達成した。

Parallel Mesh Generation of FEM with the Transportation Problem

Yoichi KONO　Ken NISHIMATSU
Hideo FUKUMORI　Yoichi MURAOKA
School of Science and Engineering , WASEDA University
3-4-1 Ohkubo Shinjuku-ku , Tokyo 169 , JAPAN

The traditional method of parallel mesh generation cannot achieve high performance because it generates proper meshes sequentially for work load distribution before domain decomposition. Presented here is a new algorithm for parallel mesh generation of the FEM on distributed memory parallel computers that uses the transportation problem for work load distribution among PEs to generate meshes in parallel before domain decomposition. By reassigning subdomains with the transportation problem, a speedup ration of 15 has been achieved as compared to sequential time. A work load balance ration of 1.6 has also been achieved for a 32 PE configuration.

1　はじめに

本稿では領域分割法 FEM(有限要素法) の並列三角形分割法を提案する。

従来の粗分割–細分割法 [1][2] では、粗分割の処理を逐次に処理していた。この粗分割処理では、あとに続く処理において PE(プロセッサ) 間の負荷分散を達成するため PE 数に比例した三角形数を生成する必要がある。そのため、PE の数が多くなると、粗分割処理が多くなり高いスピードアップが望めないという欠点があった。

そこで本稿では、輸送計画を用いて、粗分割も含めて並列化する手法を提案する。本手法では、始めに PE 間の負荷分散を考えずに最低 1 個の三角形を各 PE に割り当て、各 PE が担当する領域 (1 個以上の三角形) を並列に粗分割する。次に PE 間の負荷分散を達成するため、輸送計画を適用する。これは、各 PE の担当領域の接続関係からコストを決定し、このコストと各 PE の持つ三角形数から作成された輸送行列を解くものである。この輸送行列に従って各 PE は並列に三角形を交換し負荷分散を行なう。

本手法を、富士通の分散メモリ型並列計算機 AP1000 に実装した結果、逐次処理の実行時間に対して、PE32 台で約 15 倍のスピードアップが得られた。また負荷分散の指標として、全 PE のなかで 1 個の PE の担当する最も多い節点数と、全 PE の担当する節点数の平均値の比を調べると、輸送計画によって平均 1.6 倍に抑えることができた。

2　従来の手法

FEM の並列要素分割の手法には、粗分割–細分割法がある [1][2]。

この粗分割–細分割法は以下の 3 処理からなる (図 1 に従来法を PE4 台に適用した例を示す)。

1 逐次粗分割

まず解析領域を三角形で覆い、この三角形を節点数の平均化に必要な三角形数まで粗分割する ([2] では PE64 台の時、粗分割数約 1000 個)。この処理は従来は逐次処理されていた。

2 PE への割り当て

後に続く細分割の負荷分散、ソルバ部の負荷分散を達成するため、あらかじめ細分割

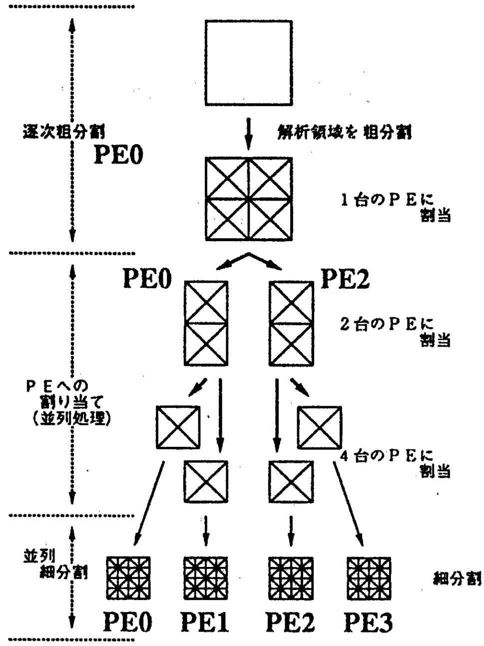


図 1: 従来法

後の節点数を計算し、最終的な節点数が平均化されるように、粗分割三角形を PE に割り当てる。PE への割り当てはほぼ並列に処理できる (図 1 のように粗分割後の三角形数を半分にする処理を繰り返す)。

3 並列細分割

各 PE は与えられた粗分割要素を解析に必要な細かさの要素になるまで並列に要素分割する。

この粗分割–細分割法を、多くの PE を持つ並列計算機に適用する場合、粗分割の要素数 (PE 数に比例) が多く必要であるため逐次処理 (粗分割) が増加しスピードアップが望めない ([2] において PE64 台に適用したとき、平均 10 倍の高速化が得られている)。

3　提案する手法

本手法は輸送計画により、すべての要素分割を並列に処理するため高速化が可能である。提案する並列化手法の流れを図 2 に示す。入力情報は、解

析領域の頂点座標、細かくしたい部分の座標、最終的な三角形の総数とする。また本手法では、PE間の節点数を平均化すると、負荷分散が達成されると仮定する。

1 三角形分割

まず解析領域を三角形に分割する。

2 PE への割り当て

図3のように分割と PE への割り当てを同時に行ない、各PE に担当領域として最低1個の三角形を割り当てる。この処理は並列に実行できる。第4章において詳細を述べる。

3 並列粗分割

全 PE の三角形の総数が節点数の平均化に必要な数になるまで、各 PE は担当領域の三角形を分割する。第5章において詳細を述べる。

4 輸送計画

各PE の担当している三角形数と、PE の担当領域の接続関係から求めたコストを使って、輸送行列を作成する。この輸送行列から最適な三角形の各PE への配置が求まる。本研究では、計算量が $O(PE^3)$ の効率的な解法 [3] を用いている。この処理は逐次に処理する。第6章において詳細を述べる。

5 節点数の平均化

輸送行列の解にしたがって、各PE の担当する三角形を通信によって変更し、各 PE 間の節点数を平均化する。この処理は並列に実行できる。第7章において詳細を述べる。

6 並列細分割

変更後の担当領域を、各 PE は解析に必要な総数 (入力情報) まで、並列に三角形分割する。

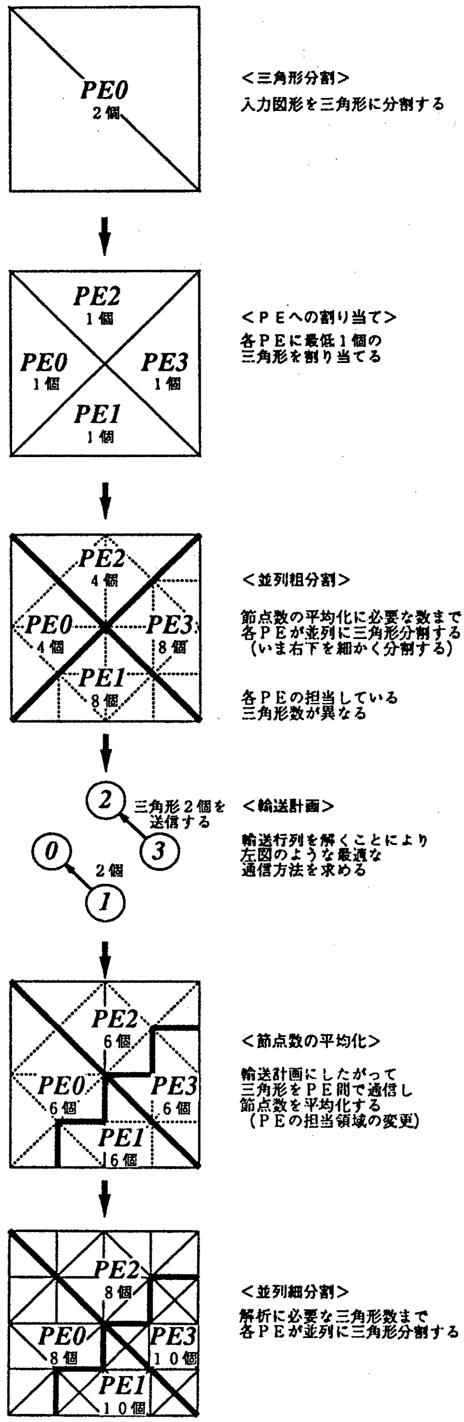
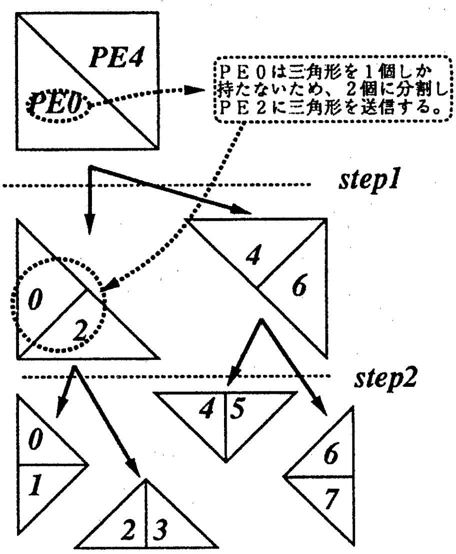


図 2: 本手法のながれ

4 PE への割り当て

PE への割り当て処理では、図3 のように1台のPE(親PE) が担当する三角形の半分を、もう1台の別の PE(子 PE) に送信していき、すべてのPE に最低1個以上の三角形を割り当てていく。も

し子 PE へ送信する時に親 PE が三角形を 1 個しか持たない場合は、すべての PE の持つ三角形をさらに分割し、三角形を 2 個以上にしてから、子 PE に三角形を送信する。1 回の送信ごとに、各 PE は親 PE を介して隣接する PE 番号の情報 (境界情報) を変更していく (例えば図 3 *step1* では、PE2 と PE6 が接するという境界情報を、PE4 と PE0 を介して、PE2 と PE6 は知る)。これにより全体の通信を行なわずに、局所的な通信のみで境界情報の変更ができる。



三角形の数字はＰＥ番号を表す

図 3: PE への割り当て

5 並列化粗分割

並列化粗分割では、細かくしたい座標からの距離によって各三角形の分割数を決定し、各 PE ごとに三角形分割を並列に行なう (細かさの違う三角形を生成する)。このとき、1 回の分割ごとに PE 間の通信によって三角形の総数を求め、節点数の平均化に必要な総数になるまで粗分割処理を繰り返す。

6 輸送計画の適用

6.1 輸送計画

輸送計画は以下のように定義される。

輸送計画とは、非負の数 x_{ij} の $m \times n$ 大きさの表 $x = (x_{ij})$ で、
制約条件

$$\sum_{j=1}^{n} x_{ij} = a_i \qquad (i = 1, 2, \cdots, m)$$

$$\sum_{i=1}^{m} x_{ij} = b_j \qquad (j = 1, 2, \cdots, n)$$

のもとで、目的関数 $\sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij}$ を最小にするものを求める問題である [3]。ここで、a_i、b_j は与えられた非負の整数であって、$\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j$ をみたす。

以後、c_{ij} をコストと呼ぶ。本研究では、Ford-Fulkerson アルゴリズム [3] によって輸送計画を解く。

6.2 節点数の平均化への適用

この輸送計画を PE 間の節点数を平均化の処理に適用する。

具体的な輸送計画の例を図 4 に示す。各生産地の生産量と、各消費地の消費量、および各生産地ー消費地間のコストから最適解が求まる。

これを節点数の平均化に適用すると図 5 のようになる。生産地と消費地には PE が相当する。生産量には節点数の平均化前の担当する三角形数、消費量には節点数の平均化後の担当する三角形数 (全 PE の持つ三角形数の平均値) が相当する。ここで、節点数が十分多いとき、節点数と三角形数が比例関係にあると仮定し、三角形数の平均化をすることにより節点数の平均化を達成する。

この輸送計画を解くことにより、PE 間の節点数の平均化を行なうための、PE 間の通信方法 (三角形の送信先 PE と受信元 PE の組合せ) が導かれる。

6.3 コストの計算

次に輸送計画におけるコスト c_{ij} の決定方法を述べる。

コストは各 PE(の担当領域) の接続関係：PE 間の距離から求める。ある PE が、1 個以上の PE と

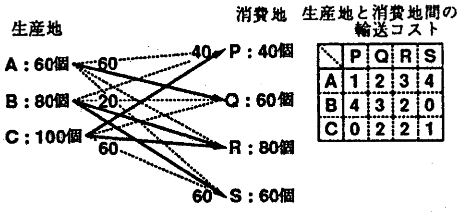


図 4: 生産地－消費地の輸送計画の例

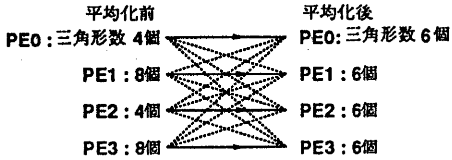


図 5: PE 間の輸送計画

隣接している場合、この隣接する PE との距離を 1 とする。自分自身の PE の距離は 0 である。次に、今考えている PE の隣接している PE が、さらに隣接している PE を持つ場合、この新しい PE との距離を 2 とする… 。このように、各 PE 間の距離を求めたのち、この距離の値を 2 乗したものをコスト c_{ij}(PEi と PEj の距離の 2 乗) とする。

かりに隣接PE 間のみコスト 1、それ以外の PE との通信の組はコストを無限大とすると、各 PE は隣接 PE 以外の PE に三角形を送信できなくなる。このままでは、平均化前に担当する三角形の数が多く、隣接 PE のみと三角形の通信を行なっても、平均化できない場合も考えられる。この場合隣接 PE のさらに隣の PE にも三角形を送る必要がある。したがってコストは PE 間の接続距離に関係した値が望ましい。

またコスト c_{ij} を PE 間の距離と等しくしたまま輸送計画を解くと、各 PE は平均化前の自分の担当する三角形を自分自身に送信するという輸送計画をつくり出してしまう。そして、前述のように隣接 PE に三角形を送っただけでは平均化できない場合、このような PE は「一番遠くの PE」に三角形を送るような輸送計画をつくり出してしまう。この輸送計画を実行した場合、「一番遠くの PE」は担当する三角形が 2 領域以上になってしまう。FEM の前処理部である要素分割部は、各 PE の境界節点数を減らすため、担当する三角形を 1 領域にまとめておく必要がある。したがって、遠くの PE との通信コストを極端に高くし、なるべく隣接 PE 間の通信を解にもつ輸送計画を作成する必要がある。こうすることにより比較的に PE の担当する三角形を 1 領域にまとめることが容易になる。本手法では計算が簡単なため、またコストの値がオーバーフローになるのを防ぐため、コストを距離の 2 乗にした (コストを距離の 3 乗にしても、結果に大差はなかった)。

コストの決定は $O(PE^2)$ の計算量で処理可能である。計算に必要な情報は、各 PE に隣接する PE の番号である。この処理を並列化するには、各 PE が全ての PE の隣接する PE 番号を持たなくてはならない。このために非常に多くの通信が必要になる。したがって本手法では、コスト c_{ij} を 1 台の PE で逐次に計算する。

例を示す。仮に図 6 -(a) のように、各 PE に三角形が割り当てられたとする。コストの計算に必要な情報は

$PE0$ の隣接する $PE : \{PE1, PE4\}$
$PE1$ の隣接する $PE : \{PE0, PE2\}$
$PE2$ の隣接する $PE : \{PE1, PE3\}$
⋮

である。この時の距離 1 の PE 間の接続関係は図 6 -(b) のようになる。距離 2 の PE 間の接続関係は図 6 -(c) のようになり、このときのコストを 4 とする…。

このようにして求めた距離を表 1 に、コストを表 2 に、また最適解を計算した結果を表 3 に示す。この最適解をすべての PE に送信し、全 PE が表 3 の結果を共有する。表 3 の輸送計画にしたがって各 PE は節点数の平均化の処理を行なう。

表 1 : 距離

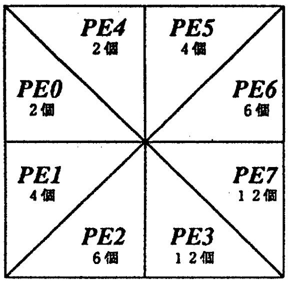
PE	(0)	(1)	(2)	(3)	(4)	(5)	(6)	(7)
(0)	0	1	2	3	1	2	3	4
(1)	1	0	1	2	2	3	4	3
(2)	2	1	0	1	3	4	3	2
(3)	3	2	1	0	4	3	2	1
(4)	1	2	3	4	0	1	2	3
(5)	2	3	4	3	1	0	1	2
(6)	3	4	3	2	2	1	0	1
(7)	4	3	2	1	3	2	1	0

表 2 : コスト

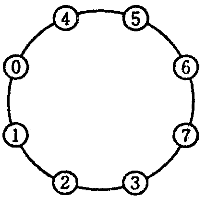
PE	(0)	(1)	(2)	(3)	(4)	(5)	(6)	(7)
(0)	0	1	4	9	1	4	9	16
(1)	1	0	1	4	4	9	16	9
(2)	4	1	0	1	9	16	9	4
(3)	9	4	1	0	16	9	4	1
(4)	1	4	9	16	0	1	4	9
(5)	4	9	16	9	1	0	1	4
(6)	9	16	9	4	4	1	0	1
(7)	16	9	4	1	9	4	1	0

表 3 : 輸送計画における最適解 (空白は 0 を示す)

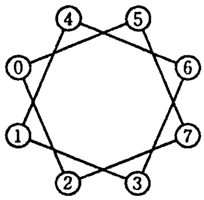
PE	(0)	(1)	(2)	(3)	(4)	(5)	(6)	(7)	a_i
(0)	2								2
(1)	4								4
(2)		6							6
(3)			6	6					12
(4)					2				2
(5)					4				4
(6)						6			6
(7)							6	6	12
b_j	6	6	6	6	6	6	6	6	



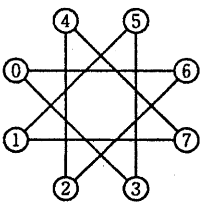
(a) 各PEの担当領域と、担当三角形数



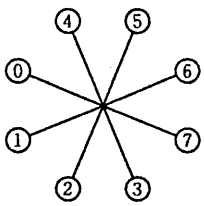
(b) 距離1の各PEの接続関係
(番号はPE番号を表す)



(c) 距離2の各PEの接続関係
(コスト＝4)



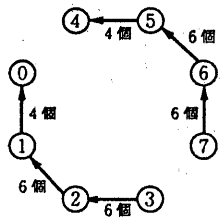
(d) 距離3の各PEの接続関係
(コスト＝9)



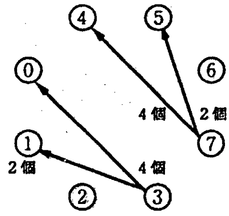
(e) 距離4の各PEの接続関係
(コスト＝16)

図 6: コストの計算例

ここで、コスト c_{ij} を PE 間の距離と等しくすると、図 7 -(b) のように、本来望ましくない輸送計画が最適解になってしまう。図 7 -(b) の結果を実行した場合、PE0 と PE4 は、離れた領域を担当することになる。したがって、図 7 -(a) のように輸送計画を行なうためには距離とコストを等しくしてはならない。



a) コスト＝距離の2乗
の場合



b) コスト＝距離
の場合

図 7: コストと距離を等しくしたときの問題

7 節点数の平均化

三角形数が十分に多い場合、節点数と三角形数はほぼ比例関係にあるため、三角形数を平均化することで節点数の平均化を行なう。

前述のようにコストの計算において、コストを PE 間の距離の 2 乗にすることによって、輸送計画による PE の担当領域の変更後も、各 PE の担当する三角形をなるべく 1 領域になるようにしていた。

節点数の平均化の処理においても、PE の担当する三角形が 1 領域になるように、以下のように送信する三角形を決定する。

- 輸送計画の解で、比較的少ない三角形数の通信の組合せは、送信先の PE の三角形を 1 領域にまとめるのを困難にする。そのため輸送行列の解のなかで、小さい値の解は比較的大きな値の解に組入れる。

- 自分以外の PE へ三角形を送信する PE は、まず自分の担当する重心座標 (x,y) を、相手の受信側 PE へ送信する。受信側ではそれらの (x,y) 座標から、節点数の平均化のあとに受信側が担当する領域の重心座標 (x,y) を計算する。この座標を送信側 PE に返し、送信側 PE ではその複数の座標を自分の担当領域に写像する。写像された座標に一番近い三角形から、送信する三角形を決定する (図 8)。

この処理を行なっても、PE の担当する三角形が 2 領域以上になってしまう場合がある。このときには、担当する複数領域のうち、一番大きい領域以外を周辺の PE に割り当てなおす。

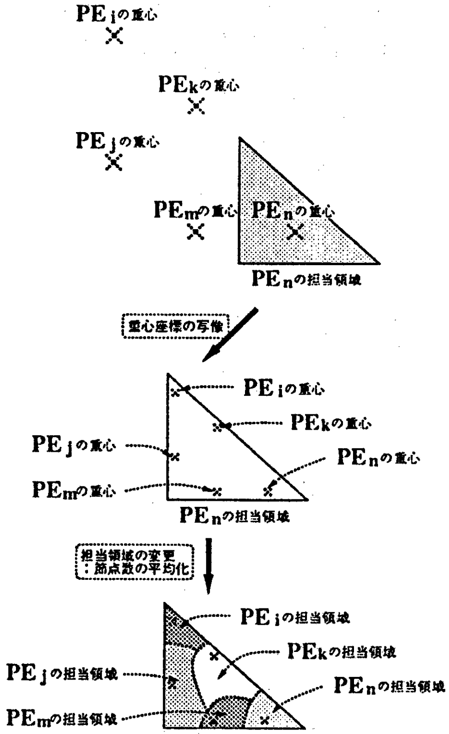


図 8: 送信する三角形の決定

8 評価

本手法を分散メモリ型並列計算機に実装し評価した。使用した富士通の AP1000 は 8×8 の 64 台構成であり、ピーク性能 5.56MFLOPS/cell である。評価する図形として数種類の図形を入力し、粗分割三角形数約 2000 個、最終三角形数約 10000 個とした。

本研究で採用した輸送計画の解法 Ford-Fulkerson アルゴリズム [3] の処理時間を図 9 に示す。輸送計画の処理時間は、図 9のように PE の増加とともに急激に増加する。この結果から、従来法 [2] の分割にかかる全処理時間は約 1 秒強である (PE 64 台のとき) ため、本手法は PE 数 64 台以上では、逆に遅くなってしまう。

次に本手法を数種類の図形に適用した結果の処理時間を図 10 に、スピードアップを図 11 に示す。PE32 台で、逐次処理に比べて約 15 倍のスピードアップを達成した。また PE32 台を使った時、節点数の平均化の処理により、全 PE の中で最大の節点数と平均の節点数の比は平均 1.6 倍となった。

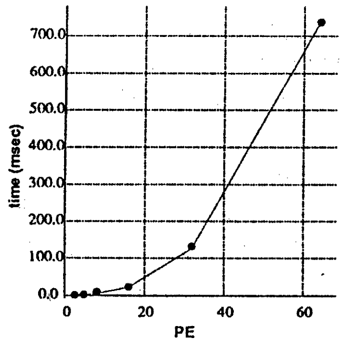


図 9: 輸送計画の処理時間

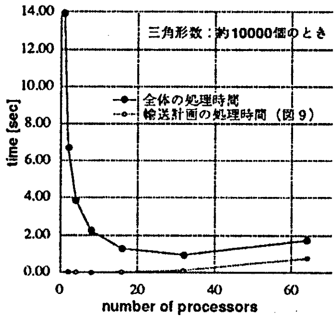


図 10: 処理時間

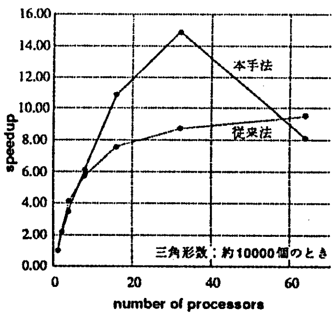


図 11: スピードアップ

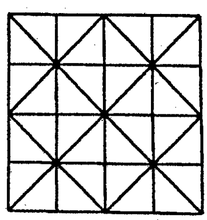


図 12: PE の初期割り当て

従来法 [2] では、平均 1.3 倍であるため、従来法より悪い結果になっている。この理由として

- 節点数の平均化の処理において、三角形を通信する時に、少ない三角形の通信を行なっていないため、完全に輸送計画にしたがっていないこと
- PE が極端に1ヶ所に集中するような解析図形では、三角形を遠くの PE へ通信しなくてはならない。この場合、各 PE の担当する三角形が 2 領域になりやすいこと

が考えられる。

輸送計画により、各 PE の担当領域が変化する様子を示す。例として、右下の付近を細かく分割するように指定した正方形をしめす。PE への割り当て処理を終了した時の様子を図12 に示す。各 PE は三角形を 1 個だけ担当している。

並列粗分割が終了した時点の各 PE の担当領域と、粗分割の様子を図 13 に示す。

輸送行列の解にしたがって、PE の担当領域を変化し、細分割していく様子を図 14 に示す。右下に PE が集中している様子が分かる。

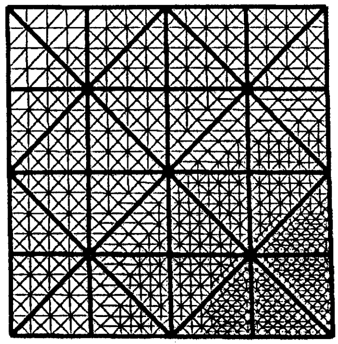


図 13: 粗分割状態と PE の初期割り当て

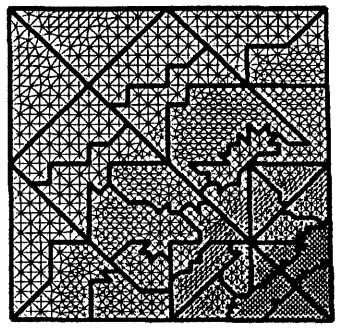


図 14: 細分割例と各 PE の割り当て

9 まとめ

本研究では、有限要素法の要素分割部の並列化を提案・評価した。従来の粗分割–細分割法において、逐次に処理されていた粗分割を、本手法では並列に実行し、さらに輸送計画を用いて、各PEの担当する三角形数を平均化した。富士通の分散メモリ型並列計算機AP1000に実装した結果、PE32台のとき平均 15 倍のスピードアップが達成できた。しかし 64 台以上の PE に本手法を適用した場合、$O(PE^3)$ の計算量を必要とする輸送計画の処理が無視できなくなる。したがって、本手法を 64 台以上の PE に適用する場合、(1) 輸送計画は PE64 台以上に適用すると遅くなるため PE32 台ごとに階層化して輸送計画の計算量を減らす、(2) 輸送計画の並列化する、などが必要になってくる。

本研究では、輸送計画による PE の担当領域の変更を、粗分割処理のあとに 1 回しか行なっていない。この処理を粗分割後のみではなく、細分割の後にも行なうことで、最終的な節点数の平均化の達成も可能である。また本研究では 2 次元の解析図形を対象としているが、本手法は 3 次元分割の並列化にも有効である。

謝辞

本研究に当たり、AP1000 を使わせて頂いたことを (株) 富士通および並列処理センターに感謝いたします。

参考文献

[1] I.St.Doltsinis and S.Noelting : Studies on parallel processing for coupled field problems , *Computer Methods in Applied Mechanics and Engineering 89* , pp. 497-521 (1991).

[2] 河野 他：分散メモリ型並列計算機上の有限要素法のための要素自動分割、情処第 48 回全大、1-135、1994.

[3] George B. Dantig : 線形計画とその周辺 Linear Programming and Extensions 、廣済堂、pp. 503-513(1983).