JAIST Repository

https://dspace.jaist.ac.jp/

Title	Natural Element Method に適した並列アルゴリズムの スケーラビリティーの検討							
Author(s)	大原,健一							
Citation								
Issue Date	1998-03							
Туре	Thesis or Dissertation							
Text version	author							
URL	http://hdl.handle.net/10119/1154							
Rights								
Description	Supervisor:松澤 照男, 情報科学研究科, 修士							



Japan Advanced Institute of Science and Technology

修士論文

Natural Element Method に適した 並列アルゴリズムのスケーラビリティーの検討

指導教官 松澤照男 教授

北陸先端科学技術大学院大学 情報科学研究科情報システム学専攻

大原健一

1998年2月13日

Copyright © 1998 by Keni di Oohara

目 次

1	はじ	めに	1					
2	Natural Element Method							
	2.1	NEM の概要	3					
	2.2	基礎方程式	4					
	2.3	方程式の離散化・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	4					
	2.4	要素分割....................................	7					
		2.4.1 Delaunay Triangul ation	7					
		2.4.2 Del aunay Tri angul ati o の手順	8					
		2.4.3 Del aunay fli p	9					
	2.5	補間関数	10					
		2.5.1 Voronoi Cell	10					
		2.5.2 Natural Neighbour Interporation	12					
	2.6	節点の追加・削除・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	15					
		2.6.1 節点の追加	15					
		2.6.2 節点の削除	15					
	2.7	速度と圧力の計算方法	16					
		2.7.1 速度の計算	16					
		2.7.2 圧力の計算	16					
3	NEN	I の並列化	17					
	3.1	領域分割方法	17					
		3.1.1 領域分割の手順	18					

	3.2	境界条件について	19
	3.3	並列計算の手順・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	19
4	実験	及び考察	21
	4.1	解析例	21
	4.2	並列計算....................................	25
		4.2.1 評価基準について	25
		4.2.2 検討方法	25
		4.2.3 各並列計算部分における速度向上比	31
		4.2.4 全実行時間の速度向上比	43
5	±۶	<i>w</i>	51
0	80	v2	91

5 まとめ



2.1	要素分割の比較・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	7
2.2	Lawson のスワッピングアルゴリズム	9
2.3	Delaunay flip	10
2.4	Voronoi 分割図	11
2.5	1st Voronoi Cell(左)と 2nd Voronoi Cell(右)	12
2.6	1st Voronoi Cellと 2nd Voronoi Cellの重ね合わせ	13
3.1	N = 4、M = 1 の場合の領域分割図	19
4.1	解析領域及び境界条件	22
4.2	初期節点配置	23
4.3	初期節点配置を基にした要素分割図	23
4.4	速度図	24
4.5	速度向上比 (PE=2)	26
4.6	並列計算部分の速度向上比 (冊4)	27
4.7	並列計算部分の速度向上比 (冊3)	28
4.8	並列計算部分の速度向上比 (田母6)	29
4.9	並列計算部分の速度向上比 (100step 平均)	30
4.10	中間流速及び速度計算部分における速度向上比 (田型)	32
4.11	中間流速及び速度計算部分における速度向上比 (田4)	33
4.12	中間流速及び速度計算部分における速度向上比 (田舎)	34
4.13	中間流速及び速度計算部分における速度向上比(田46)	35
4.14	中間流速及び速度計算部分における速度向上比(100step 平均)	36
4.15	圧力計算の速度向上比 (昭2)	38

4.16	圧力計算の速度向上比 (PE=4)			•	•	•	•		•	•	•	•	•	•	•	•		39
4.17	圧力計算の速度向上比(形部)	•	•	•	•	•	•		•	•	•	•	•	•	•	•		40
4. 18	圧力計算の速度向上比(円46)	•		•	•	•	•		•	•	•	•	•	•	•	•		41
4.19	圧力計算の速度向上比(100step 平均).	•	•	•	•	•	•		•	•	•	•	•	•	•	•		42
4.20	全実行時間における速度向上比(田翌)	•			•		•			•	•	•		•		•		44
4.21	全実行時間における速度向上比 (田4)	•			•		•			•	•	•		•		•		45
4.22	全実行時間における速度向上比(田3)	•			•		•			•	•	•		•		•		46
4.23	全実行時間における速度向上比(田46)				•		•			•	•	•		•		•		47
4.24	全実行時間における速度向上比(100step	2	Ŧ:	均)	•	•								•			48

表目次

第1章

はじめに

流体の運動の様子を調べる方法として、オイラーの方法とラグランジュの方法とがあ る。オイラーの方法は流れを場として解析する方法であるのに対し、ラグランジュの方法 は流れを粒子の集まりとみなし、その流体粒子の運動を追跡し、解析する方法である。

解析領域を三角形要素に分割したメッシュによって離散化し、近似解を求める数値解法 の場合にラグランジュ的方法を用いると、タイムステップが進行する毎に節点が移動する ことから、その度毎に三角形要素の分割を行ってメッシュを張り直すことが必要となって くる。このことから計算量が増大し、又、メッシュの更新を行なった場合に大きく歪んで 潰れてった要素が生成され面積が求められなくなり、解析が進められなくなる場合が起こ る。オイラー的方法で解析する場合はメッシュが固定されている為、この様な問題は起こ らない。

以上の様な理由から、オイラー的方法で解析することが盛んに行なわれている。しか し、移動境界問題や自由表面問題といった大きな変形が伴う問題を扱う場合にはラグラン ジュ的方法で解析することが望まれている。それは、時間が進行する毎に解析領域の境界 を追跡しながら解析を行なうことが出来るという特徴があるからである。

Jean Braun ら [1, 2, 3] によって新しく提案された Natural Element Method(以下 NEM) は、ラグランジュ的数値解析方法である。NEM では Del aunay triangul ati on と呼ばれる 要素分割方法を用いて大きく歪み、潰れてしまったメッシュが生成されない様にしてい る。また、Natur Neightur for at iounう補間方法を用いている。しかし、NEM は Del auguar i angul やNootnur nei goduri for opt at ionの計算時間 が増大するという問題があり、並列化を行ない計算時間を短縮することが望まれている。 本研究室の沼形は、並列計算機 CM-5 を用いて、NEM の並列化を行なった。並列化を 行なった事でプロセッサエレメント (以下 PE) が 2,4,8 の時には理論値の約 7 割から 8 割 の性能を得、PE が 16 の時には理論値の約 5 割の性能を得ることが確認された。

しかし、並列計算を行なっても流れの様子が逐次計算で行なった場合の結果と同じ流れ の様子を実現することが必要である。その為に、各 PE 間の領域境界部分でオーバーラッ プしている節点について何らかの境界条件を与えることが必要となってくる。

また、その並列アルゴリズムについて PE の台数の増加とその性能の関係(スケーラビ リティー)を検討する必要がある。また領域分割方法の違いで同じ PE を用いた時に、そ の性能にばらつきが見られ、どの様な領域分割を行なった時に最も良い性能が得られるの かどうかを検討する必要がある。

そこで本研究では、並列計算機 Cray-t3eを用い、また通信関数として Me ss a Bassing Int er fa 以下 MP I)を採用し、NEMの並列アルゴリズムを並列計算機 cray-t3e に移植 し、並列計算を行なった時に逐次計算と同じ流れが再現されるようにし、その並列アルゴ リズムについて、スケーラビリティーの検討を行なうことを目的とした。

その結果、8PE までは流れの様子を再現することが出来、領域分割を二次元方向に分割すことで、PE の増加に対して、より良い性能を得ることができた。

第2章

Natural Element Method

2.1 NEM の概要

流体の運動の調べる方法には、オイラーの方法とラグランジュの方法とがある。オイ ラーの方法は解析領域に対して固定点を設置し、流れの様子を調べる方法である。一方、 ラグランジュの方法は流体の運動を追跡する為に、節点を移動させて流れの様子を調べる 方法である。

ラグランジュ的方法で解析を行なうと、タイムステップ毎にメッシュを張り直すことが 必要となり、そのために計算量が増大するという問題がある。その上、あるタイムステッ プにおいては歪んで潰れてしまった要素が生成される場合があり、この歪んだ要素により 要素の面積が求められなくなり、精度の良い補間を行なうのに支障をきたす事になる。そ のため、流体の数値解析を行なう場合にはオイラーの方法が用いられることが多い。しか し移動境界問題や自由表面問題といった大きな変化が伴う問題を扱う場合、解析領域がタ イムステップの進行とともに変化するため、追跡して解析することの出来るラグランジュ 的解析方法を用いることが望まれている。

Jean Braun らによって新しく提案された Natural Element Method (以下 NEM) [1] は、ラグランジュ的数値解析を行なう場合においてこれらの問題点に対して有効に対処さ れた数値解析方法である。NEM の最も大きな特徴は、要素分割方法と補間方法にある。 解析領域の要素分割法としては Del aunay Triangul ati on [4]を用いている。この要素分割 方法を用いる事によって、正三角形に出来るだけ近い要素を生成することを可能にしてい る。そして NEM 独自の補間方法である Natur Nei gbour Interoplat i [ol a] は、この 要素を用いることで均一でない不規則な分布をした節点にたいしても精度の良い補間を 行なうことが可能になった。

2.2 基礎方程式

NEM の基礎方程式は、Navier Stoke s方程式の移流項が省かれた運動方程式と連続式から成る。

$$\frac{\partial u}{\partial t} + \frac{\partial p}{\partial x} - \frac{1}{Re} \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) = 0$$
 (2.1)

$$\frac{\partial v}{\partial t} + \frac{\partial p}{\partial y} - \frac{1}{Re} \left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right) = 0 \qquad (2.2)$$

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0 \tag{2.3}$$

ここで、u、v は流速、p は圧力、Re はレイノルズ数を表す。

2.3 方程式の離散化

時間方向の離散化には分離型解法 [6] である流速修正法を適用する。ただし、圧力 P に ついては陰的に解く。以下、n はタイムステップを表す。また、 Δt タイムステップ n か らは n+1 の間に経過する微小時間を表す。

$$\frac{u^{n+1} - u^n}{\Delta t} = -\frac{\partial P^{n+1}}{\partial x} + \frac{1}{Re} \left(\frac{\partial^2 u^n}{\partial x^2} + \frac{\partial^2 u^n}{\partial y^2}\right)$$
(2.4)

$$\frac{v^{n+1} - v^n}{\Delta t} = -\frac{\partial P^{n+1}}{\partial y} + \frac{1}{Re} \left(\frac{\partial^2 v^n}{\partial x^2} + \frac{\partial^2 v^n}{\partial y^2}\right)$$
(2.5)

式(2.)↓(2.)●の発散をとると、

$$div\left(\frac{u^{n+1}-u^n}{\Delta t}\right) = -\frac{\partial^2 P^{n+1}}{\partial x^2} + \frac{1}{Re}div\left(\frac{\partial^2 u^n}{\partial x^2} + \frac{\partial^2 u^n}{\partial y^2}\right)$$
(2.6)

$$\frac{\partial^2 P^{n+1}}{\partial x^2} = \frac{1}{\Delta t} div(u^n - u^{n+1}) + \frac{1}{Re} div(\frac{\partial^2 u^n}{\partial x^2} + \frac{\partial^2 u^n}{\partial y^2})$$
(2.7)

$$\frac{\partial^2 P^{n+1}}{\partial x^2} = \frac{1}{\Delta t} di \, v(v^n - v^{n+1}) + \frac{1}{Re} di \, v(\frac{\partial^2 v^n}{\partial x^2} + \frac{\partial^2 v^n}{\partial y^2}) \tag{2.8}$$

以上の式から圧力についてはポアソン方程式で求めていることが分かる。 この時、n+1のタイムステップにおける非圧縮条件は、

$$di\,\iota(\frac{\partial u^{n+1}}{\partial x} + \frac{\partial v^{n+1}}{\partial y}) = 0 \tag{2.9}$$

より、

$$\frac{\partial u^{n+1}}{\partial x} = 0 \tag{2.10}$$

$$\frac{\partial v^{n+1}}{\partial y} = 0 \tag{2.11}$$

となるので、

$$\frac{\partial^2 P^{n+1}}{\partial x^2} = \frac{1}{\Delta t} di \, v \tilde{u} \tag{2.12}$$

$$\frac{\partial^2 P^{n+1}}{\partial y^2} = \frac{1}{\Delta t} di \, v \tilde{v} \tag{2.13}$$

ただし、ũ、ũは中間流速を表し、次式で定義される。

$$\tilde{u} = u^n + \frac{\Delta t}{Re} \left(\frac{\partial^2 u^n}{\partial x^2} + \frac{\partial^2 u^n}{\partial y^2} \right)$$
(2.14)

$$\tilde{v} = v^n + \frac{\Delta t}{Re} \left(\frac{\partial^2 v^n}{\partial x^2} + \frac{\partial^2 v^n}{\partial y^2} \right)$$
(2.15)

以上で時間についての離散化を行なった。

次に、これを重み付き残差法を基にした近似解法である Galerkin 法により、空間につ いての離散化を行なう。自然境界条件を 0 とした場合、式(2.4)(2.5)(2.12)(2.13) (2.14)(2.15)の重み付き残差方程式は、それぞれ以下のように表される。

$$\int_{\Omega} W \tilde{u} d\Omega = \int_{\Omega} W u^n d\Omega - \frac{\Delta t}{Re} \int_{\Omega} di \, W \, di \, u^n d\Omega$$

$$\int_{\Omega} W \tilde{v} d\Omega = \int_{\Omega} W v^n d\Omega + \frac{\Delta t}{Re} \int_{d\Omega} W d \, i w^n n d \, S - \frac{\Delta t}{Re} \int_{\Omega} d \, i W d \, i v^n d\Omega$$

$$\int_{\Omega} d \, iW \, d \, ipv^{n+1}d\Omega = -\frac{1}{\Delta t} \int_{\Omega} W d \, i\tilde{u}d\Omega \tag{2.16}$$

$$\int_{\Omega} d \, iW \, d \, ipv^{+1}d\Omega = -\frac{1}{\Delta t} \int_{\Omega} W d \, i\tilde{v} d\Omega \qquad (2.17)$$

$$\int_{\Omega} W u^{n+1} d\Omega = \int_{\Omega} W \tilde{u} \, d\Omega - \Delta t \frac{\partial^2 u^n}{\partial x^2}$$
 (2.18)

$$\int_{\Omega} W v^{n+1} d\Omega = \int_{\Omega} W \tilde{v} d\Omega - \Delta t \frac{\partial^2 v^n}{\partial x^2}$$
 (2.19)

となり、以上で離散化方程式が求めることができる。

また、 u_i^{n+1} は各節点の次のタイムステップにおける速度を表している。タイムステップ n における節点の座標を x_i^n と表すことにすれば、節点の移動を次式で表す事とする。

$$x_i^{n+1} = x_i^n + \frac{\Delta t}{2} (u_i^n + u_i^{n+1})$$
(2.20)

2.4 要素分割

NEMでは、タイムステップ毎に要素分割を行なことが必要となる。NEMでは、Delaunay Triangul ati on[1, 4] と呼ばれる要素分割方法を用い、大きく歪んで潰れてしまった要素が 生成されることを防ぎ、より正三角形に近い要素を生成することで精度の良い補間を実行 することが可能となっている。

2.4.1 Delaunay Triagulation

Delauyn Tariangul aと呼ばれる要素分割法は、大きく歪んで潰れてしまった要素が 生成されることを防ぐ方法であり、本研究では三角形要素を用いた要素分割を行なう。歪 んだ要素や潰れてしまった要素を生成しない為の方法として、1つの三角形要素の外接円 の内部にはいかなる節点をも含ないようにしている。

まず、Delauyn Tariangul aがどのような要素を生成するか考えてみる。図に4点を つかった要素分割を2通り示す。左が Delauyn Tariangul aでありれ 右が四角形 abcd の対角線を入れ換えて出来る要素分割である。Delauyn Tariangul attよる要素の方が 正三角形に近いことがわかる。今は簡単のために4点の例を取り上げたが、多数の点に 対する Delauyn Tariangul att正正角形に近い要素をつくり出すことが確かめられて いる。要素分割では正三角形に近い形状の要素が望ましいので、Delauyn Tariangul ation は要素分割に適しているといえる。





図 2.1: 要素分割の比較

2.4.2 Delaunay Triagulation の手順

上の図で示した、Delaunay Triangul ati onを任意に設定された節点群に対して適応させ る手順として、Lowsonのスワッピングアルゴリズム / cited と呼ばれるアルゴリズムを 採用した。

以下に、そのアルゴリズムの手順を示す。

- 1. 追加した節点 p を含む三角形要素を探す。(図の三角形 abc)
- 2. その三角形要素の各頂点と節点 p を結び、三角形要素を3つの三角形要素に分ける。 この3つの各三角形要素に対して隣接する三角形のうち、節点 p に向かいあう三角 形要素をスタックに入れる。

(図の三角形 adb、bec、cfa を作り、bad、cbe、acfをスタックに入れる。)

- スタックから三角形要素を1つ取りだす。
 (図の三角形 adb。)
- 4. 節点 p が三角形要素の外接円内に含まれるか調べる。含まれる場合は 5 へ。そうで ない場合は 3 へ。
 (図は節点 p が三角形 adb の外接円内に含まれる場合を示している。)
- 5. 対角線を入れ換えて2つの三角形要素を作り替える。この2つの三角形要素に隣接する三角形のうち、節点pに向かいあう三角形要素ををスタックに入れる。3へ。
 (図では、四角形 padb の対角線 ab を取り除き、対角線 pd を加えた。このとき出来た三角形は pad と pdb である。)



図 2.2: Lawson のスワッピングアルゴリズム

2.4.3 Delaunay flip

上述した Delaunay Triangul ati onを用いて各タイムステップ毎に要素分割を行なうこ とは解析領域全体に対して毎回三角形要素分割を行なうことを意味する。これもし、三角 形の外接円の内部に節点を含んでいるような要素が少ないのであれば局所的な変更で済む のであれば、計算量の減少につながることになることから、要素分割方法の局所的な分割 方法として Delauyn bip/citenem, 凶呼ばれる方法がある。この方法は図に示されて いるように、隣接する二つの三角形要素で構成される四角形を考える。三角形要素の外接 円内部に他の節点を含む場合に、四角形の対角線を入れ換える操作が図に示されている。 これを、すべての要素について外接円内部に他の節点を含まなくなるまで行なえば、その 要素分割は Delauyn Triangul ata a co操作は Delauyn bip(1,2]と呼ばれる。 この方法は、最初に三角形要素分割が行なわれている事が必要とする。そして、NEM

を並列化する為の目的の1つは、計算時間を減らすことであるので、今回は、タイムス テップの2ステップ以降の計算についてはこのDelauynfai を用いることにした。



☑ 2.3: Delaunay flip

2.5 補間関数

NEM では、Natura Neighour Interoplation [4] 呼ばれる補間方法が用いられる。 その重みは Voronoi Cellと呼ばれる多角形の面積から求められる。

2.5.1 Vorma Cell

Delauyn Tariangul æ行なnた三角形要素の各辺の垂直二等分線を結んでいくと凸 多角形分割が出来る。これを Voron 分割といい、その多角形の1つ1つを Voron Gell という。図は不規則に配置した節点にたいする Voron 分割を表している。

各 Vor on Gel には 1 つの節点が含まれている。この隣接した Voronoi Cell の節点を つないでいくと、Del aunay Tri angul ati oが出来ることが知られている。即ち、Del a uyn a Tr i angul alti Von on Gel の隣接関係を表している。このとき、Vor on Gel の辺 は Del a uyn Tri angul aの三角形要素の辺の垂直二等分線の一部になっている。また、 Vor on Gel の頂点は Del a uyn Tri angul aの三角形要素の外接円の中心になってい る。このように、Del a uyn Tri angul aと Nonon Gel とは双対関係にあることが分 かる。



図 2.4: Voronoi 分割図

2.5.2 Natural Nigh bour Interportation

NEM では、要素分割に使っている Delaunay Triangul ati onから、Voronoi Cel lを求め、 その面積を使って補間する。

解析領域の Del a uyn Tri a ng ul aから求められた Vor on Cielを特に 1st Vor on oi Celと呼ぶ。次に、領域に補間したい点を1点加えて出来る Vor on Cielを 2nd Voronoi Cellと呼ぶ。また補間点を加えて出来た 2nd Voronoi Cell に対して隣接している 2nd Voronoi Cellに含まれている節点を補間点の Natur Nei geburと呼ばれる。

図に、補間点付近の 1st Voronoi 分割と 2nd Voronoi 分割の例を示してある。点線は Delauyn Tariangul aを表しれ その頂点の + 印は、節点である。2nd Voronoi 分割の 中央の + 印が補間する点である。更に、1st Voronoi 分割と 2nd Voronoi 分割を重ね合わ せたものを図に示した。



図 2.5: 1st Voronoi Cell(左) と 2nd Voronoi Cell(右)

1st Voronoi Gellと2st Voronoi Gellを重ね合わせた図より、補間点を加えた2nd Voronoi Gellが1st Voronoi 分割の5つの1st Voronoi Gellによって、5つの領域に分割されているのがわかる。注意してみると、この5つの1st Voronoi Gellは、どれも補間点のNatural Nei ghbour の1st Voronoi Gellである。このように、Natural Nei ghbour の1st Voronoi Gellによって分割されるのである。

ここで、補間点の 2nd Voronoi Cellの面積を 1 とし、分割された 5 つの領域の面積を重 みとして用いるのが、Natur Neighudnte ropratiである。すなわち、補間点の関数



図 2.6: 1st Voronoi Cell と 2nd Voronoi Cellの重ね合わせ

値は、補間点の Natural Nigh bour となっている節点の値が重み付けされて求められる。 なお一般には、補間点の 2nd Voronoi Cell は少なくとも 3 つの領域に分割される。補 間点の近傍の節点分布によって、いくつの領域に分けられるかが決まる。 以下に、点 x において、Natural Neigh bour Interportationの 補間関数値 f(x) を求める手順をまとめる。

- 補間点 x の Nat ur Nei gbb urになっている節点を全て捜し出す。
 要素分割を与えている Del aunay Tri angul ati on から、補間点の Nat ur Nei gbb ur を簡単に求めることが出来る。補間点を加えて Del a uyn Tri ang ul aを行なったとき、補間点と同じ三角形要素に属する節点が Nat ur Nei gbb urである。従って、結局、元の(補間点を加えていない) Del a uyn Tri ang ul al ab bhて、補間点を外接円に含む要素の各頂点が Nat ur Nei gbb urである。また、これらの要素はCircum Triange と呼ばれる。
- 2. Natural Nigh boar の lst Voroni Gl を求める。
 1st Voroni Gl の各頂点は Circumariangの外接円の中心である。
- 補間点の 2nd Voronoi Cellを求める。
 補間点の 2nd Voronoi Cell の各頂点は、Natural Neighbour と補間点で構成される
 三角形の外接円の中心である。
- 4. Natural Nigh bor の 1st Voronoi Cellと補間点の 2nd Voronoi Cellの重なり部分 の面積を求める。それらを $Vor_n(x)$ と表すことにする。
- 5. $Vor_n(\boldsymbol{x})$ の総和を求める。これを、 $S(\boldsymbol{x})$ で表すことにする。

$$S(\boldsymbol{x}) = \sum_{n} Vor_{n}(\boldsymbol{x})$$
(2.21)

6. 各 Natural Neigh bour の重み $W_n(x)$ を求める。

$$W_n(\boldsymbol{x}) = \frac{Vor_n(\boldsymbol{x})}{S(\boldsymbol{x})}$$
(2.22)

7. 各 Natural Neigh bour の関数値を f_n とすると、補間値 f(x) は次式で求められる。

$$f(\boldsymbol{x}) = \sum_{n} W_{n}(\boldsymbol{x}) f_{n}$$
(2.23)

2.6 節点の追加・削除

NEM では、タイムステップ毎に節点が移動することから三角形要素の形も変形する。 初期節点配置は速度や圧力の変化が著しい円柱近傍に多く配置させ、変化があまり見られ ない所には節点を疎らに配置させているといった不規則な配置をしている。このままタイ ムステップを進行させていくと局所的に節点が集まり過ぎ三角形要素の大きさが小さくな り過ぎ面積が求められなくなることがおこることが考えられる。また、その逆に三角形要 素の大きさが大き過ぎて精度の良い計算を行なう事に支障をきたす。

この問題点に対応するために、三角形要素の面積について最大値と最小値の基準を予め 決めておき、三角形要素の面積がその基準値よりも小さくなり過ぎた所には節点を削除す る事を考え、その反対に三角形要素の面積がその基準値よりも大きくなり過ぎた所には節 点を追加することを考えた。

2.6.1 節点の追加

節点の追加については、追加する節点の速度や圧力の値を補間してやり、追加する。その後は前述した Lawson のスワッピングアルゴリズムを用いて Dela unay Triangul ati on を行ない、節点を追加した新たな三角形要素分割を行なった。

具体的な手順は以下のようになる。

- 1. 新たに追加する節点 P の座標*x*を決定する。
- 2. Natural Nigh boar Interpolation を用いて座標*x*に関して補間して、節点 P に 必要な値を求める。
- 3. 座標*x*に節点 P を置いて、Delauyn Tariangul aの手順にしたがって要素分割する。

2.6.2 節点の削除

任意の Del a uyn Tar i a ng ul aが与えられているとする。そこから節点を1点削除す ると、その点を含む三角形要素も消失し、その部分には多角形の空白部分が生じる。この 空白の多角形内部を新しい三角形要素で埋めなくてはならない。しかも、それで与えた 要素分割が、領域全体として Del a uyn Tar i a ng ul att なのていなければならない。し かし、この空白の多角形だけを Delaunay Triangul ati onにすれば充分であることは明ら かである。なぜなら、節点の削除によって、空白の多角形の外側の三角形要素が外接円内 に節点を含むようになることはあり得ないからである。したがって、この後やるべきこと は空白になった多角形領域について、DelauynTriangul æ求めることである。これ は、多角形の端の領域から順次、外接円内に多角形の頂点を含まない要素を作っていけば 良いだけである。

2.7 速度と圧力の計算方法

2.7.1 速度の計算

速度を求める計算には、まず中間流速を求め、次に速度を求めるという方法をとる。これは式(??)および式(??)を解くことで求められる。速度を求める場合には、解を陽的に求めている。

2.7.2 圧力の計算

圧力は式(??)を解くことで求められる。この方程式系の係数行列を解く解法として、 直接解法(LU分解)を用いた。

第3章

NEM の並列化

本研究では、並列計算機 Cray-T3e を用い、並列計算を行なった。通信方式として Message Passing Interface (以下 MP I)を用い、並列化を行なう為の手法として領域分割法を用い る。この方法を用いて解析領域をプロセッサエレメント(以下 PE)数に分割する。オイ ラー的方法を用いると、タイムステップが進行してもメッシュは変わらない為、一度領域 分割を行なうと解析が終了するまで領域分割を再度行なう必要がない。ラグランジュ的解 析方法である NEM では節点がタイムステップ毎に移動するので、その都度領域分割を行 なう必要が生じ、また、各 PE が受け持つ節点数が均等になる様に負荷分散を行なう必要 がある。

3.1 領域分割方法

領域分割方法として、解析領域を微小領域に分割し、その微小領域内の節点を調べ、各 PEの受け持つ節点数が均等になるように微小領域単位で分割を行なう方法をとる。この 時領域境界部分の扱いについては、各PEが受け持った節点の三角形要素を調べ、その三 角形要素の1頂点でも領域外に存在するならばその頂点をその三角形要素を受け持って いるPEが受け持つ節点し、その節点をそのPEの領域境界として認識させる。すなわち 各PEの領域境界部分はお互いが同じ節点をオーバーラップして受け持つようにした。

3.1.1 領域分割の手順

上で述べた領域分割方法の手順を具体的に述べると以下の様になる。

- 1. 領域を図のような縦長の帯状の小領域に分割する。
- 2. 各帯状小領域に入っている節点数を調べる。
- 3. 各小領域に含まれる節点の数が均等になるように、帯状小領域単位で領域全体を N 個の小領域に分割する。
- 4. 解析領域を2次元方向に分割する場合には上で分割したN個の各小領域に対して、 横長の帯状領域に分割する。
- 5. N 個の各小領域において、各帯状小領域に入っている節点数を調べる。
- 6. N 個の各小領域において、含まれる節点の数が均等になるように、帯状小領域単位 で領域全体を M 個の小領域に分割する。
- 7. 各 PE で、割り振られた節点を頂点とする三角形要素を調べ、それらを受け持つ要素とする。このとき、三角形要素の1頂点でも割り振られた節点であるならば、受け持ちの要素とする。即ち、その要素の頂点には、他の PE の節点となっているものも存在する。この要素が、領域境界でオーバラップする部分をあたえる境界要素となる。
- 8. オーバラップ部分の要素の頂点となっている節点のうち、他の PE に割り振られた 節点を、新たに自分の受け持ちの節点として加える。



図 3.1: N = 4、M = 1 の場合の領域分割図

3.2 境界条件について

前述した領域分割方法により、各PEはその領域境界においてはお互いにある同じ節点 をオーバーラップして受け持っている。このまま中間流速、圧力、速度のについて並列計 算を行なった場合、逐次計算の時の流れの様子とは異なってしまう。そこで、各PEの領 域境界部分について何らかの境界条件を与えることが必要となる。そこで今回は境界条件 として、オーバーラップして受け持った節点については前のタイムステップで計算した値 を与えた。すなわちオーバーラップしている節点については未知数として扱わず、既知数 として扱うこととした。

3.3 並列計算の手順

並列計算の計算手順は、以下の手順で行なう。

- 1. 節点の初期データを入力する。
- 2. 節点データをもとに Delaunay triangul at i onにより要素分割する。(1 ステップ目の時のみ)

- 3. 各 PE に節点と要素のデータを転送する。
- 4. 領域分割し、各 PE が自分が受け持つ領域及び節点を決定する。
- 5. 各 PE において、中間流速 \tilde{u} 、 \tilde{v}] を求める。
- 6. 各 PE において、圧力 p を求める。
- 7. 各 E において、流速 *u*、[*v*] を求める。
- 8. 各 EE において、節点の位置を更新する。
- 9. 各 PE で求めた解を、1 つの PE に集める。
- 10.2 ステップ以降では集められたデータから新しい節点配置に対して Delaunay flip に より要素分割を更新する。

(計算終了のタイムステップ数を終えていれば計算終了。)

11. 節点の削除及び追加を行なう。

第4章

実験及び考察

4.1 解析例

本研究では2次元円柱周りの流れの解析を行なった。

通常、オイラー的手法を用いる場合には、一様な流れの中に円柱が置かれていると考 えて、問題を解析する。つまり、円柱は解析領域に固定され、流体が円柱周りを移動して いく問題として扱われる。ラグランジュ的手法においても、そのように問題を扱うこと はもちろん可能である。しかし、ラグランジュ的手法の特徴を考慮すると、むしろ静止流 体中を円柱が移動していく問題として扱う方が自然である。(この場合は移動境界問題と なる。)

今回は、この問題を後者のように扱い解析した。 初期条件を、以下の様に与える。

- 微小時間増分量 $\Delta t = 0.01$
- レイノルズ数 *Re* = 100
- 初期値は速度、圧力とも0とする。
- 境界条件は外壁の境界の速度を 0 とする。
- 各 PE 間の領域境界部分に対する境界条件として、オーバーラップして受け持った
 節点については前のタイムステップにおける値を用いた。



図 4.1: 解析領域及び境界条件

また、計算開始時には図のように節点を配置した。図から分かるように円柱の近傍に節 点を多く配置させ、円柱から離れた所では節点を疎らに配置させている。これは、速度や 圧力の変化が円柱近傍で著しく激しく、円柱から離れた所では与えられた節点集合に対し て、Delaunay Triangul ati onを行ない図のような要素分割を得た。



図 4.2: 初期節点配置



図 4.3:初期節点配置を基にした要素分割図



図 4.4: 速度図

この計算で タイムステップ数 100 における流れの様子を図に示す。速度図では、双子 渦のきっかけとなる渦が生じている様子が見られる。これはレイノルズ数に対応した流れ の特徴が出ていることが分かる。

4.2 並列計算

4.2.1 評価基準について

並列計算アルゴリズムの性能を評価するものとして速度向上比を用いる。速度向上比と は、N 個の PE を用いて並列計算を実行したときの時間 *T_N*と、逐次計算を行なった時の 時間 *T₁*とを比較してどれだけの性能が得られたのかを検討する基準となるもので、次式 のように定義される。

速度向上比 =
$$\frac{T_1}{T_N}$$

4.2.2 検討方法

この速度向上比を用いて、PE 数の増加と共に速度向上比にどのような変化が表れるの かを検討した。またこの並列アルゴリズムは、中間流速の計算、圧力の計算、速度の計 算、そして次のタイムステップにける節点の位置を更新する計算の部分について並列化さ れている。そこでまず並列化した部分についての速度向上比を示した。並列計算を行なう 際に領域分割方法を縦方向、横方向について変えていき速度向上比にどの様な変化が表れ るのか検討を行なった。図の decom N-M とは解析領域を縦にN 分割そして、横に M 分 割したことを示している。



all parallel calculation(PE=2)

図 4.5: 速度向上比 (PE=2)



図 4.6: 並列計算部分の速度向上比 (PE=4)



図 4.7: 並列計算部分の速度向上比 (PE=8)



all parallel calculation(PE=16)

図 4.8: 並列計算部分の速度向上比 (PE=16)



図 4.9: 並列計算部分の速度向上比 (100step 平均)

図より、PE 数が増加するにつれて速度向上比が、急激に大きくなっていることがわかる。そこで、並列化されている部分の内の何処の計算部分が影響を与えているのかをかを確認する必要がある。そこで各並列計算部分についての速度向上比を表しどの部分の影響が一番大きいのか検討を行なった。

また、各 PE において領域分割方法を変えると、速度向上比にばらつきが見られることが 分かる。これは、領域分割の方法を変えると、PE が受け持つ節点数の増減が見られるこ とから生じるものと考えられる。PE が受け持つ節点数にばらつきが見られるのは、領域 分割がうまく出来ていないこと、または領域分割の結果、オーバーラップ部分の節点数が 増加していることが挙げられる。

4.2.3 各並列計算部分における速度向上比

次に並列化を行なった各計算部分のそれぞれについての速度向上比を表し、PE数が増 大するにつれて、速度向上比にどの様な変化が表れるのか、またどの部分の計算量の変化 が速度向上比に大きく影響を及ぼしているのかどうかの検討を行なった。

中間流速の及び速度計算部分における速度向上比

まず、中間流速及び、速度計算部分、節点位置の更新を行なう部分についての速度向上 比の結果を次の図に示した。この図より、PE 数に比例した速度向上比が得られいること がわかる。これは、中間流速及び、速度の計算については圧力解の計算の様に行列を分解 する必要がない対角化行列を用いていることから、並列化することで減少した PE の受け 持つ節点数の割合に比例して速度向上比が大きくなったものと考えられる。節点位置の更 新を行なう部分についても同様のことが考えられる。また、領域分割方法の違いで速度向 上比にばらつきがみられる。これもやはり、各 PE 間でオーバーラップして受け持ってい る節点数の違いから起きているものと考えられる。

31



図 4.10: 中間流速及び速度計算部分における速度向上比 (PE=2)



図 4.11: 中間流速及び速度計算部分における速度向上比 (PE=4)



図 4.12: 中間流速及び速度計算部分における速度向上比 (PE=8)



図 4.13: 中間流速及び速度計算部分における速度向上比 (PE=16)



図 4.14: 中間流速及び速度計算部分における速度向上比 (100step 平均)

圧力計算部分における速度向上比

次に、並列計算部分である圧力解の計算部分についての速度向上比を図に示した。 この図より、PE 数の増加と共に速度向上比の変化が他の並列計算部分の速度向上比の変 化より非常に大きくなっていて、並列計算部分全体の速度向上比の変化に1番大きく影響 を与えていることが分かった。

これは圧力解を求める計算方法が、受け持つ節点のオーダリング、2次元の圧力行列の作成、圧力ベクトルの作成、圧力解の導出にはLU分解を用いるというように他の並列計算 部分以上に非常に節点数に対する依存が大きくなっていることが挙げられる。

それは、オーダリング、圧力ベクトルの作成については並列化することで減少した PE の 節点数の割合に比例すること、圧力行列の作成及び圧力解の導出(LU 分解)には並列化 をすることで減少した節点数の2 乗に比例して速度向上比が増大するからである。

また、中間流速、速度、節点位置の更新部分の時と同様に領域分割方法の違いにより、速度向上比にばらつきが見られ、そして2次元方向に領域分割を行なった方が1次元方向 に分割を行なった時よりも速度向上比が大きく増大していることが分かった。



図 4.15: 圧力計算の速度向上比 (PE=2)



図 4.16: 圧力計算の速度向上比 (PE=4)



図 4.17: 圧力計算の速度向上比 (PE=8)



図 4.18: 圧力計算の速度向上比 (PE=16)



図 4.19: 圧力計算の速度向上比 (100step 平均)

4.2.4 全実行時間の速度向上比

次に、この NEM の並列アルゴリズムの並列計算ではなく1 PE で計算が行なわれる部 分として、三角形要素分割の更新及び、節点の追加・削除を行なう部分が挙げられる。 この三角形要素の更新、節点の追加・削除といった並列計算ではない部分と、PE 数の増 加に伴ったメッセージパッシング量の増加が、この NEM の並列アルゴリズムの全実行計 算時間に、どの様な影響を与えるのか、そして PE 数の増加い伴う速度向上比の変化につ いて検討を行なう必要がある。

そこで、この NEM の並列アルゴリズムについての全計算実行時間に対する速度向上比を 次の図に表した。



図 4.20: 全実行時間における速度向上比 (PE=2)



図 4.21: 全実行時間における速度向上比 (PE=4)



図 4.22: 全実行時間における速度向上比 (PE=8)



図 4.23: 全実行時間における速度向上比 (PE=16)



図 4.24: 全実行時間における速度向上比 (100step 平均)

図より、計算の立ち上がり時は、初期節点配置の設定及び、三角形要素の作成にかなり の時間を必要とすること、タイムステップの進行に対して流れの様子が大きく変わる為に 節点の移動が激しくなり多くの三角形要素の形が大きく変化し、その三角形要素の更新に かなりの時間が掛かる。そしてそれに伴う節点の追加・削除についても同様にかなりの時 間を必要とすることから、速度向上比が安定しない。

一方、タイムステップが進むと流れは徐々に安定してくることから、要素分割や節点配置 の変更は少なくなり、全計算時間対する割合が減っていき、平衡状態となってくる。

逐次計算においては、圧力解を求める計算時間が全計算実行時間のほとんどを占めてい る。並列化を行なったことにより、圧力解の計算量が減少された為にこれほど大きな速度 向上比の変化が見られるものと考えられる。

また、100step 平均で表した速度向上比の図より、領域分割を2次元方向で行なった方が 1次元方向で領域分割を行なった時に比べ、より高い速度向上比が得られていることがわ かった。

2,4,8PEを用いて並列計算を行なった際に、一番解析を進める事が出来たのは、一番速度 向上比の大きい領域分割の時であった。しかしながら 16PE の時、領域分割 decomp8-2 の時の方が decomp4-4の時よりも速度向上比が高いが実際の計算においては decomp4-4 の時の方が解析が進められるといった事が起こった。これは、今回の計算時間の測定方 法が全各 PE の計算が終った時間を計算時間としている為、受け持つ節点が多いとそれ だけ計算時間がより多くかかってしまうことになる事に原因がある。decomp8-2の場合、 各 PE の受け持つ節点は 110-188の間であり、decomp4-4の場合、各 PE の受け持つ節点 は 165-198の間であった。decom9-2の場合には負荷分散が均等に行なわれておらず、そ の為、ある PE ではオーバーラップ部分の節点の影響が大きくなり、解析がすすめられな くなるといった事態が起こった。一方、decom4-4の場合には、負荷分散が均等に行なわ れている為に解析が進められることが出来た。しかしながら、decom9-2の場合、各 PE 内、一番多く受け持っている節点数が、de comp4-4の場合の一番多く受け持つ節点数よ りも少ない為、見かけ上 decom9-2の場合の方が、decom4-4の場合より良い速度向上 比を示すことになった。

この図と並列化を行なった部分における速度向上比とを比較すると、PE を多く用い計算 を行なうと、並列化が行なわれていない部分の影響及び、メッセージパッシングの影響が 大きくなり、PE 数の増加に対する速度向上比の増加の割合が少なくなっている。

49

従って、PEの増加に伴うメッセージパッシング量を抑える事には限界があるので、この NEMの並列アルゴリズムの内、更に並列化を行なうことが可能である部分について並列 化を行なえば、速度向上比の更なる向上が期待出来るものと考えられる。

並列化を期待できる部分として再要素分割の部分が考えられるが、この時も他の並列計算 と同様に各 PE の領域境界部分の扱いに問題が生じる。再要素分割を並列化しても、逆に 解析が進められなくなる場合が生じるのでこの PE の境界部分については、適切な境界条 件が求められる。

今回解析を行なった際に用いた節点数は2000 個程度であった。その為、32PE 以上を用い て解析を行なった場合には、各 PE が受け持つ節点数が少なくなり過ぎてしまい、その為 オーバーラップ部分の節点の影響が大きくなり解析が進められなくなった。また、16PE を用いた解析でも計算時間は非常に短かくなった。今回用いたモデルでは2000 個程度の 節点数で充分であったが、より複雑形状をした大規模な流れをモデルを用いた場合には、 より多くの節点数を必要となり計算時間が増大することから、32PE 以上を用いた解析を 行い、計算時間を短縮する必要が出てくるものと考えられる。

第5章

まとめ

本研究では、移動境界問題や自由表面問題に有効に対処出来ると期待されているラグラ ンジュ的解析方法である NEM について並列計算機 Cray-t3e を用いて並列化を行なった。 並列化を行なう為の手法として領域分割法を用い、各 PE が受け持つ節点数が均等にな る様に負荷分散を行なった。この際、各 PE 間の領域境界部分でオーバーラップして受け 持っている節点については境界条件として前のタイムステップで計算された値を与えた。 従ってオーバーラップして受け持っている節点については、既知数の扱いとした。こうす ることで、並列化を行う上での前提条件なる逐次計算を行なった場合と同じ流れの様子が 再現出来る様に努めた。

まず並列化を行なった部分に注目し、PE数の増加の割合と速度向上比の増加の関係について検討を行なった。次に並列化を行なった各部分について注目し、どの部分が速度向上比に一番影響を与えているのか検討を行なった。最後に計算の全実行時間にたいする速度向上比を表し、PEの増加と速度向上比の増加の関係、そして逐次計算部分が速度向上比に与える影響について検討を行なった。

この時、領域分割方法を変え、同じ PE 数を用いた時でも領域分割の仕方の違いで速度向 上比にどの様な変化が生じるのか検討を行なった。

51

謝辞

本研究を行なうにあたり、御指導を頂いた松澤教授に対し心から感謝します。

また、研究を行なうに当たり助言や指摘など各方面においてお世話になった沼形健児氏 (三井造船)並びに研究室の各位に対し、深く感謝致しますと共に更なる御発展、御活躍 を心から期待しております。

参考文献

- Jean Braun and Malcol m Sambridge, A numeri cal metho d for sol vi ngparti akli fferenti alequati onson hi ghl yi rregul aevol vi nggri ds, Nature, v376, pp655-660, 1995.
- [2] Jean Braun and Malcol m Sambridge, Dynamical Lagrangi an Remeshing (DLR): A new algorithm for solving argestrain deformation problems and itsapplication of fault-propagation dingEarth and PlanetaryScience Letter,s124, 211-220, 1994.
- [3] Mal col m Sambridge, Jean Braun and Herbert McQueen, Geophysicalparametrizationand interpolationofirreguladata usi ngnatural nei glbours, Geophys. J. Int 122, 837-857, 1995.
- [4] 谷口健男, FEM のための要素自動分割(デローニー三角分割法の利用), 森北出版, 1992
- [5] JJConr/CABellia
 共著, 奥村敏恵 訳, 流体解析への有限要素法の応用, サイ

 エンス社, 1978
- [6] 数值流体力学会 編, 非圧縮性流体解析, 東大出版会, 1995
- [7] 数値流体力学会 編,移動境界流れ解析,東大出版会,1955