

非構造格子に基づく大規模超並列計算法 とその有効性について

Studies on the Efficiency of Massively Parallel Computational Method
for Large-Scale Computation based on Unstructured Grid

斎藤克矢*、樫山和男**

By Katsuya SAITOH and Kazuo KASHIYAMA

Massively parallel finite element method based on unstructured grid for unsteady large-scale computation is presented. The shallow water equation is used for the basic equation. Parallel implementation is performed on the Fujitsu AP1000 and Connection Machine CM-5. The efficiency of the parallelization techniques is investigated. The present method is shown to be a useful and powerful tool for large-scale computations.

Keywords : massively parallel computation, large-scale computation, unstructured grid, explicit scheme

1. はじめに

近年、並列計算機のハードウェアの発達はめざましく、多数のプロセッサを有する商用の並列計算機が多数開発されている。また、一方で急速に普及したワークステーションを複数台接続して並列に計算する方法も注目されている。並列計算機は、現行のベクトル計算機の数十倍から数百倍の演算速度のピーク性能を持ち、構造解析に比べて離散化自由度数を圧倒的に必要とする流体解析の分野でも実用に供する計算が可能であるとして期待されている。特に水工学の分野でも洪水氾濫や津波、高潮などの緊急性の高い予報解析や環境問題などの長期間に渡る計算などに、並列計算機は多大なる威力を発揮するものと考えられる。しかし、並列計算機を用いる場合には、解法のアルゴリズムを逐次処理から並列処理に変更しなければならないため、アルゴリズムの根本的な変更が必要となる。また、並列計算機を効率的に用いるためには、各プロセッサの計算負荷の均等化やプロセッサ間の通信量の最小化が必要となる。今後は、並列計算機が主流を占めることから、並列計算機を用いて大次元の連立一次方程式を高速にかつ効率的に解くための並列計算法を構築することは急務であり、また水工学上重要であると考えられる。これまで、水工学分野への並列計算機の適用に関する研究は、構造格子（格子の形状が一定）に対して西村ら^[1]により行われている。しかし、非構造格子（格子の形状が任意）を前提とした大規模超並列計算に関する研究はなされていない。

* 学生員 中央大学大学院 理工学研究科土木工学専攻

** 正会員 工博 中央大学助教授 理工学部土木工学科

(〒112 東京都文京区春日 1-13-27)

本論文は、任意形状への適合性に優れている非構造格子に対する並列計算手法について提案し、複数の超並列計算機を用いてその効率化について検討を行ったものである。基礎方程式としては浅水長波方程式を用い、基礎方程式に対する空間の離散化には有限要素法を、時間の離散化には差分法を適用した。また、連立一次方程式の解法としては陽的解法を採用した。なお、並列計算機としては、並列計算専用機である富士通社の AP1000 とシンキングマシン社の CM-5 を用い、数値解析例としては高潮の大規模計算を取りあげた。

2. 並列計算機と並列計算の概要

本研究で使用した並列計算機は、富士通社の AP1000（富士通並列処理研究センター）とシンキングマシン社の CM-5（ミネソタ大学高性能計算研究センター）である。どちらの計算機も各プロセッサにメモリーが搭載された（MIMD 型）コンピュータである。AP1000 には、1021 個のプロセッサ（1 プロセッサあたりメモリー 16MB、スピード性能 8.33Mflops）がトーラス型のネットワークにより接続されており、CM-5 には、896 ノード（1 ノード 4 プロセッサ）で合計 3584 個のプロセッサ（1 プロセッサあたりメモリー 32MB、スピード性能 32Mflops）がファットツリー型のネットワークにより接続されている。使用した AP1000 では最大 512 個のプロセッサ、CM-5 では最大 512 個のノード（2048 プロセッサ）による並列計算が可能となっている。図-1 に AP1000 の構造を、表-1 にその仕様を示す。AP1000 はセルと呼ぶ 1024 個のプロセッサとホストと呼ぶワークステーションで構成されており、ホスト及びセル間でのデータ通信や同期は 3 種類のネットワークを介して行われる。

並列計算の手順は以下ようになる（図-2 参照）。

- 1) ホストより各セルに入力データ（領域データ、計算データ）を送信する。
- 2) 各セルで各時刻レベルの計算を行う。
- 3) 同期をとりながらセル間の境界上のデータを送受信する。（計算が終了するまで 2）、3）を繰り返す）
- 1) 計算結果をホストへ送信する。

3. 連立一次方程式の解法

本論文では、基礎方程式として以下に示す非定常の浅水長波方程式を用いた。

$$\dot{u}_i + u_i u_{i,j} + g(\zeta - \zeta_0)_{,i} + \frac{\tau_{ib}}{\rho(h + \zeta)} - \frac{\tau_{is}}{\rho(h + \zeta)} - A_l(u_{i,j} + u_{j,i})_{,j} = 0 \quad (1)$$

$$\dot{\zeta} + ((h + \zeta)u_t)_{,i} = 0 \quad (2)$$

ここに、 u_i :平均流速、 ζ :水位変動量、 h :水深、 g :重力加速度、 ζ_0 :気圧降下に対応する海面上昇量、 τ_{is} :海面でのせん断応力、 τ_{ib} :海底面でのせん断応力、 A_l :渦動粘性係数である。

基礎方程式を離散化するにあたり、空間方向の離散化にはガラーキン法に基づく有限要素法を、時間方向の離散化には差分法を用いた。なお、要素としては三角形一次要素、時間微分としては三段階の陽的な解法 [2] - [4] を採用した。すると、最終的に各時刻レベルにおいて次式で与えられる連立一次方程式が得られる。

$$\bar{M}X = F \quad (3)$$

ここに、 $\bar{M}$ は集中化された係数行列であり、 X は未知ベクトル、 F は既知ベクトルである。いま、図-3 に示す

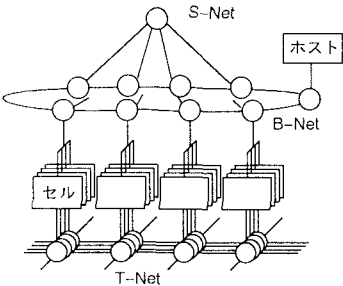


図-1 AP1000 構成

表-1 AP1000 諸仕様

プロセッサ	SPARC IU+FPU
性能	8 33Mflops
メモリ	16MB
キャッシュ	128KB
セル台数	64～1024 台
システム性能	0.53～8.53Gflops

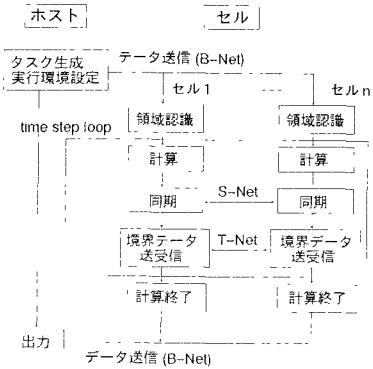


図-2 並列計算の手順

例において、節点 A と節点 B の未知量の並列解法を説明する。なお、図-3 において要素 (1)-(4) はセル 1 に含まれ、要素 (5),(6) はセル 2 に含まれている。

節点 A の未知量の求解法

ステップ 1) 節点 A を共有している各要素において、 $\bar{M}_{A(n)}, F_{A(n)}$ を計算する。(図-4) ここに、 n は節点 A を共有する要素を示す。

ステップ 2) $\bar{M}_{A(n)}, F_{A(n)}$ を足し合わせることで、節点量 $\bar{M}_A, F_A$ が求まる。

$$\bar{M}_A = \bar{M}_{A(1)} + \bar{M}_{A(2)} + \bar{M}_{A(3)} + \bar{M}_{A(4)}$$

$$F_A = F_{A(1)} + F_{A(2)} + F_{A(3)} + F_{A(4)}$$

ステップ 3) 節点 A における未知ベクトル X_A は次式から得られる。

$$X_A = F_A / \bar{M}_A$$

このように、節点が領域内に含まれる場合には、セル間の通信は必要ない。

節点 B の未知量の求解法

ステップ 1) 節点 B を共有する各セルでは、次式のように各セル内での節点量 $\bar{M}_{Bm}, F_{Bm}$ が求まる。(図-5 上) ここに、 m は節点 B を共有するセルを表す。

$$(\text{セル 1}) \quad \bar{M}_{B1} = \bar{M}_{B(3)} + \bar{M}_{B(4)}, \quad F_{B1} = F_{B(3)} + F_{B(4)}$$

$$(\text{セル 2}) \quad \bar{M}_{B2} = \bar{M}_{B(5)} + \bar{M}_{B(6)}, \quad F_{B2} = F_{B(5)} + F_{B(6)}$$

ステップ 2) セル間で境界節点データの送受信をした後、次式により節点 B における節点量 $\bar{M}_B, F_B$ を得る。(図-5 下)

$$\bar{M}_B = \bar{M}_{B1} + \bar{M}_{B2}, \quad F_B = F_{B1} + F_{B2}$$

ステップ 3) 節点 B における未知ベクトル X_B は次式から得られる。

$$X_B = F_B / \bar{M}_B$$

このように、陽的な解法では、データ送受信は、隣接するセル間のみで行えば良いので、並列化が容易である。また、以上のデータ送受信は、時刻ステップ毎に行われるが、集中係数行列 $\bar{M}$ は不変であるため送受信は 1 度で済む。なお、境界条件の処理についても、並列処理を行っている。つまり、入出力を除く全ての処理が並列化されている。

4. 効率化に関する検討

前節で示した並列計算法の有効性を検討するために、浅水長波流れ計算についていくつかの問題を設定して計算を行った。

4.1 負荷の均等化に関する検討

負荷の均等化に関する検討を行うために、各プロセッサに領域データが均等に配分された場合(図-6.A)とそうでない

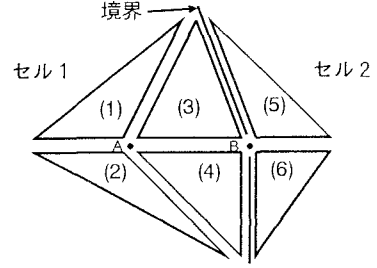


図-3 メッシュモデル

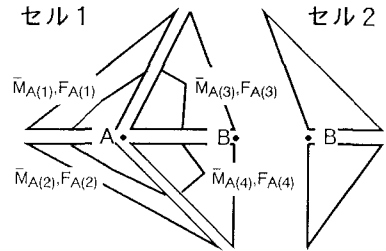


図-4 節点 A の未知量

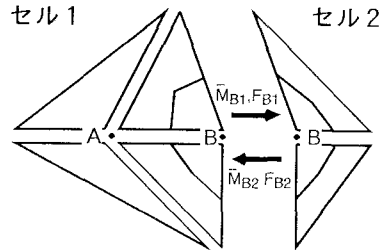
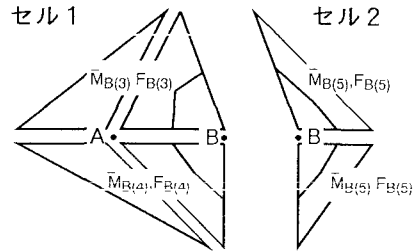


図-5 節点 B の未知量

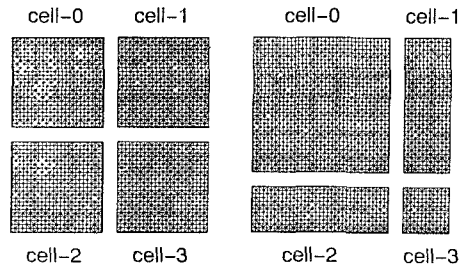


図-6.A

図-6.B

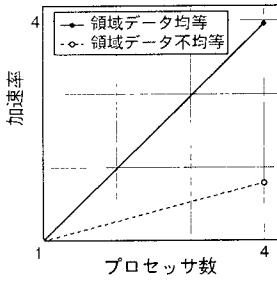


図-7 加速率

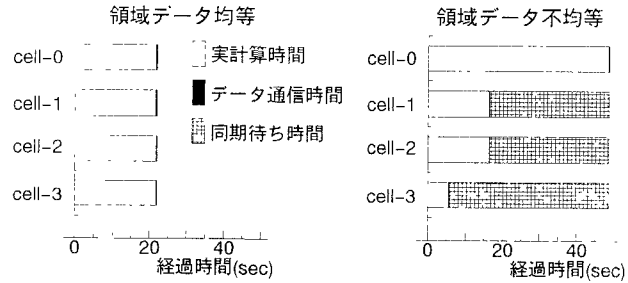


図-8 計算時間

場合（図-6.B）をそれぞれ4プロセッサを使用して計算を行った。加速率とは1プロセッサ使用時に対する演算速度の倍率であり、次式で定義されるものである。

$$\text{加速率} = \frac{\text{プロセッサ数1の経過時間}}{\text{プロセッサ数}N\text{の経過時間}} \quad (4)$$

ここに、 N はプロセッサ数を示す。図-7より、負荷を均等化することにより、プロセッサ数と同じ加速率が得られていることがわかる。また、図-8は計算時間全体に占める実計算時間、データ通信時間、同期待ち時間の割合を示したものである。図より、プロセッサ間負荷が不均一な場合には、多大な同期待ち時間が生じ、これが計算速度を著しく遅くさせる原因となることがわかる。このように並列計算を効率的に行うためには、プロセッサの負荷の均等化を図り同期待ちを最小にする必要がある。このため、非構造格子に対していかに各セルに領域データ（メッシュデータ）を均等に配分することが重要となる。ここでは、Farhat^[5]によって提案された領域分割法を用いる。この手法を用いることにより、複雑な境界形状をもつ領域も均等に分割することが可能となる。また、解析領域全体の要素分割は楢山^[6]による方法を用いている。図-9に、非構造格子分割例（要素数181991、節点数96610）とそれに対する領域分割例（512プロセッサ）を示す。

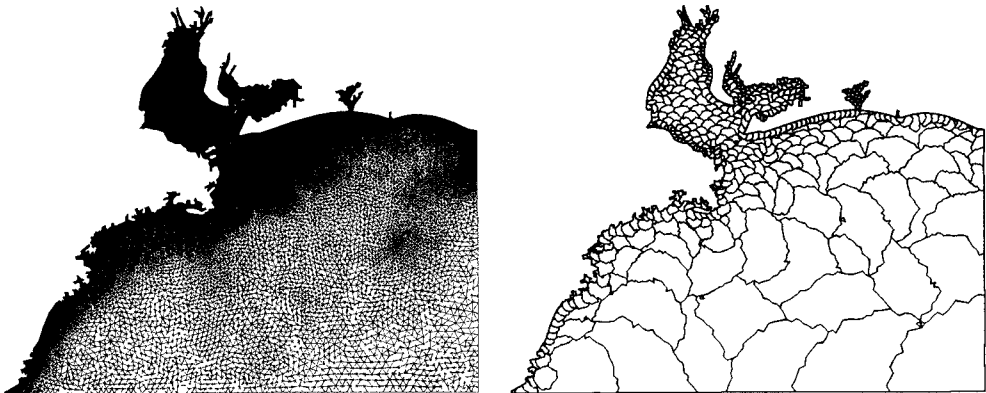


図-9 非構造格子分割例及び領域分割例

4.2 通信時間に関する検討

次に、通信時間が並列化効率に及ぼす影響を検討するために、総要素数が異なるメッシュを3種類（meshA:要素数32166、meshB:要素数21095、meshC:要素数11228）用意して、それぞれを16から最大512プロセッサ使用して同様の計算を行った。なお、例題としては伊勢湾の高潮解析を行った。図-10に加速率及び演算速度、図-11に並列化効率を示す。ここで、並列化効率とは加速率をプロセッサ数で割ったものであり、次式で定義される。

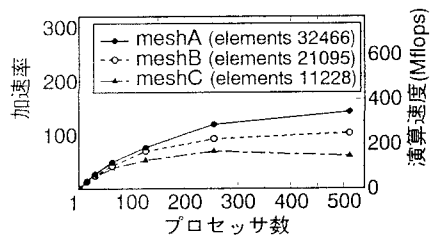


図-10 加速率及び演算速度

$$\text{並列化効率} = \frac{\text{加速率}}{\text{プロセッサ数}} \quad (5)$$

図より、プロセッサ数の増加に伴い、各メッシュとも並列化効率が低下していることがわかる。図-12 に、512 プロセッサ使用時に加速率が低下している meshC の、各プロセッサ使用時の全 CPU 時間の内訳を示す。図から、並列化効率の低下は、1 プロセッサが受け持つ要素数が減少することで、全 CPU 時間に対する通信時間の割合が増加することによって起る。このことより、問題の規模に応じて使用するプロセッサ数を選択することが計算効率の上では重要であるといえる。また、プロセッサ数を同一とした場合には、総要素数の増加により加速率及び並列化効率が向上していることがわかる。すなわち、大規模な問題になるほど効率は向上するといえる。

5. 大規模問題への適用

AP1000 では、プロセッサ単体は 16Mbyte のメモリを所有していることから、領域データの分散化（各セルに各領域データのみを記憶させる）を施すことにより、512 セルを用いた場合には最大 $16 \times 512 = 8.192 \text{ Gbyte}$ の計算容量を必要とする大規模問題を扱うことが可能である。

ここでは、領域データの分散化を行い、大規模問題への本並列計算法の適用を行う。数値解析例としては、1959 年の伊勢湾台風による高潮計算をとりあげた。なお、計算は 3 種類のメッシュ（meshL:要素数 184991、meshM:要素数 134263、meshS:要素数 79496）を用いて計算を行った。図-9 に meshL の要素分割図と領域分割図（512 セル）を示す。なお、要素分割はクーラン数が一定になるように分割されており^[6]、これにより大きな時間増分をとることができる。図-13 に台風経路、図-14 に名古屋港における気象潮位を観測結果と比較したものを示す。図より両者は良好な一致を示している。図-15 にプロセッサ数を変化させた場合の加速率及び演算速度を示す。また、図-16 には並列化効率を示す。これらの図より、大規模な問題になるほど高い性能が得られることがわかる。最も大規模な meshL では、512 プロセッサを用いた場合、約 400 の加速率を得ている。また、図-17 に meshL の場合について、使用するプロセッサ数を変化させた場合の全 CPU 時間の内訳を示す。図より、各プロセッサに割り当てられる要素数の増加に比例して、計算効率が向上しているのかわかる。また図-18 は、CM-5 用いて大規模問題への適用を行ったときの演算速度を示したものである。これらの図から、並列計算機は機種によらず、大規模問題に対して有効性を発揮することが確認される。

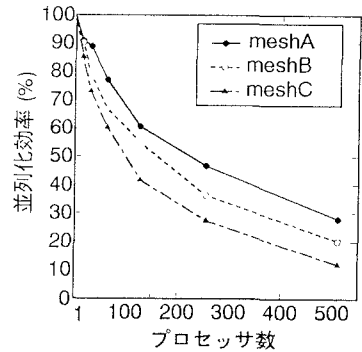


図-11 並列化効率

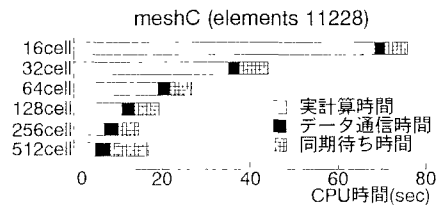


図-12 CPU 時間 (meshC)

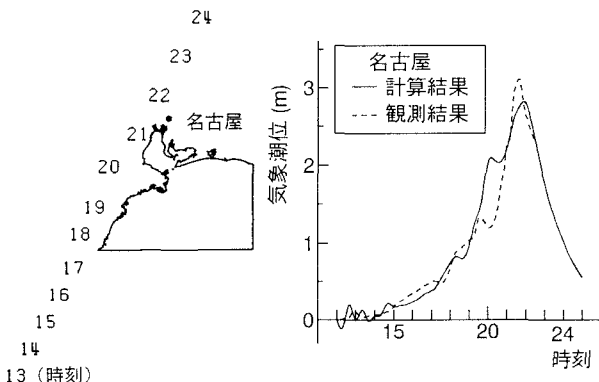


図-13 台風経路

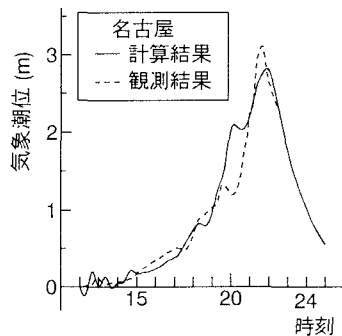


図-14 名古屋港における気象潮位

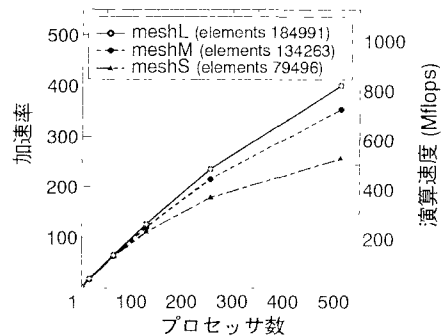


図-15 加速率及び演算速度

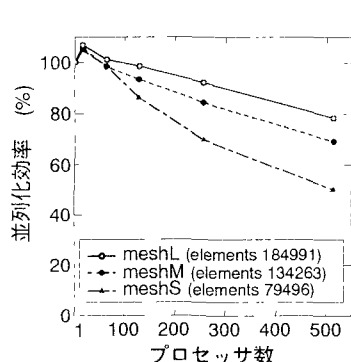


図-16 並列化効率

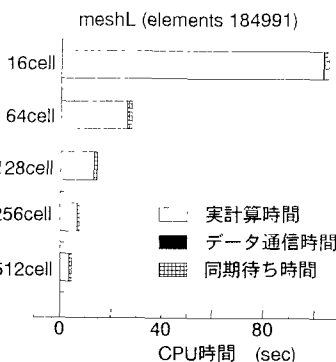


図-17 全 CPU 時間の内訳

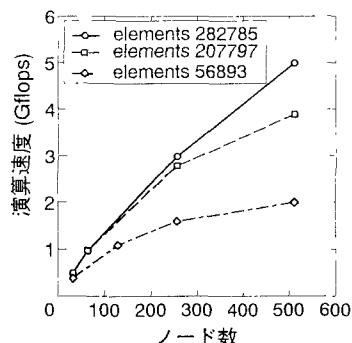


図-18 演算速度 (CM-5)

6. おわりに

本論文によって、非構造格子に基づく陽的な大規模超並列計算法を提案し、効率化に関する検討を行った。その結果、次のことが明らかとなった。

- 1) 陽的な計算法では、データ送受信は、隣接するセル間のみで行えば良いので、並列化が容易である。
- 2) プロセッサあたりの連立一次方程式の未知数の数の増大に伴い、並列化効率は向上する。このことは、大規模問題になるほど並列計算機の有効性を発揮することを意味する。なお、大規模計算（約 18 万要素）の場合、512 プロセッサを用いて加速率約 400 を得た。
- 3) 並列計算の効率化向上には、各プロセッサの負荷の均等化が重要である。また、問題の規模に応じて使用するプロセッサ数を選択することが肝要である。

なお、本論文では並列専用機を用いたが、本論文で述べた手法はワークステーションを用いた PVM (Parallel Virtual Machine) による並列計算にも有効である。今後は、陰的解法の並列化手法について検討を行う予定である。

謝辞: 本研究を行うにあたり、並列計算機の使用を許可された富士通並列処理研究センターおよびミネソタ大学高性能計算研究センターに感謝の意を表します。

参考文献

- [1] 西村仁嗣、星野力、武若聰、波の場の計算における並列シュミレータの応用、第 32 回海岸工学講演会論文集、pp.100-103、1985。
- [2] 樫山和男、伊藤英恵、Tezduyar, T., 浅水長波流れの大規模超並列計算、第 8 回数値流体力学シンポジウム講演論文集、pp.609-612、1994。
- [3] Kashiya, K., Ito, H., Behr, M. and Tezduyar, T., Three-step explicit finite element computation of shallow water flows on a massively parallel computer, *Int. J. Num. Meth. Fluids*, Vol.21, pp.885-900. 1995
- [4] Saitoh, K. and Kashiya, K.: Finite Element Computation of Storm Surge on a Massively Parallel Computer, *Int. Conf. Tech. Marine Env. Pre., Tokyo, JAPAN, Sept. 24-29, 1995* pp.678-685.
- [5] Farhat, C.: A simple and efficient automatic FEM domain decomposer, *Computers & structures*, 28(1988), pp.576-602.
- [6] 樫山和男、複雑平面領域の水環境流れ解析のための自動要素分割法の提案、水工学論文集、第 37 巻、pp.781-786.1993。