領域分割法の動的有限要素解析への適用*

大石 篤 哉^{*1}, 山田 勝 稔^{*1}
吉村 忍^{*2}, 矢川 元 基^{*2}

An Application of Domain Decomposition Method to Dynamic FEM

Atsuya OISHI, Katsutoshi YAMADA,
Shinobu YOSHIMURA and Genki YAGAWA

Recently, in order to replace experiments by computer simulations, the required scale of the model to be analyzed has become extremely large. To solve such a large-scale problem, various parallel FEM algorithms have been investigated to date. Among them, the domain decomposition method (DDM) is suitable for parallel computing because of the small memory requirement, arbitrary decomposition of the domain, and easy implementation on parallel systems. In this study, we applied the DDM algorithm to dynamic FEM analyses, and implemented it on a single CPU system. It is well demonstrated through some numerical examples that the present DDM-based dynamic FEM code is very efficient and superior in speed and memory requirements.

Key Words: Parallel Processing, Dynamic FEM, Domain Decomposition Method, Conjugate Gradient Method, Computational Mechanics, Elasticity

1. 緒 言

近年、マイクロエレクトロニクスの急激な進歩により、コンピュータの性能が飛躍的に向上してきた。それに伴い、これまでは模型や実機を用いて行っていた実験を、有限要素法などの数値解析手法による数値シミュレーションに置き換える動きが盛んになっている。

実験が往々にして高価であり、条件の設定などにも困難を伴うのに対し、数値シミュレーションは比較的安価で、パラメトリックな解析も容易であるという利点がある。

数値シミュレーションに対しては、「高精度化」の要求がますます強くなり、そのため解析対象は急激に「大規模化」してきた。すなわち、メッシュ分割を細かくし、精度を上げようとする、自由度は格段に増加する。三次元の問題を考えると、1 辺のメッシュ分割数を N 倍にすると、全体のメッシュ分割数は N^3 倍になり自由度も同じく増加する。立方体を 1 次要素で分

割する場合を考えると、 $10 \times 10 \times 10$ のメッシュでは約 3 000 自由度、それを $100 \times 100 \times 100$ のメッシュに細かくすると、約 300 万自由度という大規模な問題になる。

このような大規模な問題を許容しうる速さで解くために、スーパーコンピュータが使われてきた⁽¹⁾。スーパーコンピュータは、パイプラインプロセッサによりベクトル処理性能を飛躍的に高めたコンピュータであり、最新機種では 10 GFLOPS (1 秒間に 100 億回の浮動小数点演算が行える) を超えるものまで出現している。当然ながら、スーパーコンピュータ用に特化し、その演算能力を最大限に利用するよう設計された高速計算プログラムも数多く開発され成功をおさめてきた^{(2)~(4)}。しかし、これまでスーパーコンピュータの性能向上を支えてきたデバイス素子の性能向上が限界に近づきつつあり、計算速度を高めるための新しい手法として並列処理が注目を集めている⁽⁵⁾。

並列処理を有効に利用するためには、並列処理に適したアルゴリズムが必要とされ、これまでに数々のアルゴリズムが発表されているが、それぞれに課題を有している。例えば、各節点とプロセッサを 1 対 1 に対応させる Jordan ら⁽⁶⁾ の Finite Element Machine は、特定の問題にはよい性能が得られるがはん用性に

* 平成 3 年 11 月 14 日 第 4 回計算力学講演会において講演、原稿受付 平成 3 年 12 月 20 日。

^{*1} 正員、徳島大学工業短期大学部(〒770 徳島市南常三島町 2-1)。

^{*2} 正員、東京大学工学部(〒113 東京都文京区本郷 7-3-1)。

劣る。また、Farhat ら⁽⁷⁾の並列サブストラクチャー法は、部分構造の境界上の節点に対応する剛性マトリックスを解く過程で並列アルゴリズムが複雑となり、効率の面で難点がある。

それに対し、本研究で対象とする領域分割法⁽⁸⁾は、大規模な問題を解くのに適し、ハードウェアに依存せず、高い並列処理効率を得られる。このため、静弾性問題についてはすでにトランスピュータ、EWS ネットワーク、スーパーコンピュータネットワークなど各種の並列環境上で動作し、良好な結果が報告され⁽⁸⁾⁽⁹⁾、さらに静弾塑性問題への適用⁽¹⁰⁾も行われている。

本研究では、領域分割法を動弾性問題に適用し、精度、収束性、速度等の基本的な特性について1 CPU 上で検討を行ったので、その結果について報告する。

2. 理 論

2・1 基礎式および弱形式 簡単のために、図1のような2領域分割の場合の動弾性問題を考える。

領域 Ω 内部において体積力 $\bar{B}$ 、境界 Γ_σ 上で表面力 $\bar{T}$ 、境界 Γ_u 上で変位 $\bar{u}$ が、それぞれ境界条件として与えられているとする。これを2領域に分割すると、

$$\Omega = \Omega^{(1)} + \Omega^{(2)} \quad \dots\dots\dots (1)$$

となり、各領域の境界は、

$$\Gamma^{(k)} = \Gamma_\sigma^{(k)} + \Gamma_u^{(k)} + \gamma_{12} \quad \dots\dots\dots (2)$$

となる。ここで、 γ_{12} は二つの領域間の境界を表す。

このとき、各部分領域における基礎式は、

$$\varepsilon_{ij}^{(k)} = \frac{1}{2} [u_{i,j}^{(k)} + u_{j,i}^{(k)}] \quad \text{in } \Omega^{(k)} \quad \dots\dots\dots (3)$$

$$\tau_{ij}^{(k)} = C_{ijmn} \varepsilon_{mn}^{(k)} \quad \text{in } \Omega^{(k)} \quad \dots\dots\dots (4)$$

$$\tau_{ij,j}^{(k)} + \bar{B}_i^{(k)} = \rho \ddot{u}_i^{(k)} + c \dot{u}_i^{(k)} \quad \text{in } \Omega^{(k)} \quad \dots\dots\dots (5)$$

$$\tau_{ij}^{(k)} \nu_j^{(k)} - \bar{T}_i^{(k)} = 0 \quad \text{on } \Gamma_\sigma^{(k)} \quad \dots\dots\dots (6)$$

$$u_i^{(k)} = \bar{u}_i^{(k)} \quad \text{on } \Gamma_u^{(k)} \quad \dots\dots\dots (7)$$

であり、さらに、領域間の境界 γ_{12} において、力の釣合いと変位の連続性、すなわち、

$$\tau_{ij}^{(k)} \nu_j^{(k)} + \tau_{ij}^{(l)} \nu_j^{(l)} = 0 \quad \text{on } \gamma_{12} \quad \dots\dots\dots (8)$$

$$u_i^{(k)} = u_i^{(l)} = \mu_i \quad \text{on } \gamma_{12} \quad \dots\dots\dots (9)$$

ただし、

図1 2領域分割

$$l = k + (-1)^{k-1} \quad \dots\dots\dots (10)$$

が成り立つ。ここで、 u_i , ε_{ij} , τ_{ij} はそれぞれ変位ベクトル、ひずみテンソル、応力テンソルであり、 C_{ijmn} はフックの法則の係数テンソルである。 ν_j は境界における外向き単位法線ベクトルであり、

$$\nu_j^{(k)} = -\nu_j^{(l)} \quad \text{on } \gamma_{12} \quad \dots\dots\dots (11)$$

が成り立つ。 $\dot{u}$, $\ddot{u}$ はそれぞれ速度・加速度ベクトル、 ρ は質量密度、 c は減衰の係数を意味する。 $(\cdot)_{,j}$ は座標系の j 方向に関する微分を、右肩の添え字 (k) , (l) は、注目している部分領域番号(ここでは、1 または 2)を示す。

以上の方程式に重み付き残差法を適用し、弱形式に定式化する。弱形式での定式化に際して、式(8)、(9)の取扱いに関し、次の2とおりの定式化が考えられる。

① 式(8)を厳密に満足させ、式(9)を積分形式で満足させる(応力法)

② 式(9)を厳密に満足させ、式(8)を積分形式で満足させる(変位法)

二つの定式化を比較すると、応力法は領域分割の仕方によっては全く変位拘束を受けない領域(完全 Neumann 境界条件)が生じ解が定まらない場合があるため、領域の分割形式に制約がある。一方、変位法では領域間境界に常に変位拘束が付加されるため任意の領域分割が許される⁽¹¹⁾。したがって、ここでは変位法を採用して定式化を進めることにする。

変位法では、式(3)、(4)、(7)、(9)を厳密に満足させ、式(5)、(6)、(8)を積分形式で満足させる。すなわち、

$$\sum_{k=1}^2 \left[- \int_{\Omega^{(k)}} (\tau_{ij,j}^{(k)} + \bar{B}_i^{(k)} - \rho \ddot{u}_i^{(k)} - c \dot{u}_i^{(k)}) \delta u_i^{(k)} d\Omega \right. \\ \left. + \int_{\Gamma_\sigma^{(k)}} (\tau_{ij}^{(k)} \nu_j^{(k)} - \bar{T}_i^{(k)}) \delta u_i^{(k)} d\Gamma \right. \\ \left. + \int_{\gamma_{12}} (\tau_{ij}^{(k)} \nu_j^{(k)} + \tau_{ij}^{(l)} \nu_j^{(l)}) \delta u_i^{(k)} d\Gamma \right] = 0 \quad \dots\dots\dots (12)$$

を式(3)、(4)、(7)、(9)のもとで評価すればよい。ここで、式(12)の第1項に発散定理を用い、さらに式(9)に注意すると、

$$\sum_{k=1}^2 \left[\int_{\Omega^{(k)}} \tau_{ij}^{(k)} \delta \varepsilon_{ij}^{(k)} d\Omega + \int_{\Omega^{(k)}} \rho \ddot{u}_i^{(k)} \delta u_i^{(k)} d\Omega \right. \\ \left. + \int_{\Omega^{(k)}} c \dot{u}_i^{(k)} \delta u_i^{(k)} d\Omega - \int_{\Omega^{(k)}} \bar{B}_i^{(k)} \delta u_i^{(k)} d\Omega \right. \\ \left. - \int_{\Gamma_\sigma^{(k)}} \bar{T}_i^{(k)} \delta u_i^{(k)} d\Gamma \right] \\ + \int_{\gamma_{12}} (\tau_{ij}^{(1)} \nu_j^{(1)} + \tau_{ij}^{(2)} \nu_j^{(2)}) \delta \mu_i d\Gamma = 0 \quad \dots\dots\dots (13)$$

という弱形式表示が得られる。

式(13)は、各部分領域 $\Omega^{(k)}$ の一般的な弱形式表示

の和に領域間境界 γ_{12} 上での制約条件が付加されたものである。領域分割法では、この制約条件の部分に共役こう配法などの反復的な手法により満足させ、解を得る。

2.2 共役こう配法 共役こう配法を用いた領域分割法のアルゴリズムを示す。まず、対称・正定値である演算子 A ,

$$A(\mu_i) = -(\tau_{ij}^{(1)} \nu_j^{(1)} + \tau_{ij}^{(2)} \nu_j^{(2)})|_{\gamma_{12}} \dots\dots\dots (14)$$

を導入する⁽¹²⁾。演算子 A は、具体的には「領域間境界に変位条件を与え、各領域ごとに有限要素解析を行い、領域間境界上での節点反力の和を取る」ことを行う演算子である。この A に対して以下のように共役こう配法を用いる⁽¹³⁾。

step 0

$$\mu_i^r = \mu_i^0 (\mu_i^0 : \text{constant}) \dots\dots\dots (15)$$

$$g_i^r = A(\mu_i^0) \dots\dots\dots (16)$$

$$w_i^r = g_i^r \dots\dots\dots (17)$$

$$r \leftarrow 0 \dots\dots\dots (18)$$

なお、式(16)の右辺は、部分領域境界間に適当な変位の初期値 μ_i^0 を与えて、有限要素解析を行い、式(14)より得られる。

step 1

$$\tau_{ij}^{(k)} = \rho \ddot{u}_i^{(k)} + c \dot{u}_i^{(k)} \quad \text{in } \Omega^{(k)} \dots\dots\dots (19)$$

$$\tau_{ij}^{(k)} \nu_j^{(k)} = 0 \quad \text{on } \Gamma_{\sigma}^{(k)} \dots\dots\dots (20)$$

$$u_i^{(k)} = 0 \quad \text{on } \Gamma_{u}^{(k)} \dots\dots\dots (21)$$

$$u_i^{(k)} = w_i^r \quad \text{on } \gamma_{12} \dots\dots\dots (22)$$

を解いて得られた $\tau_{ij}^{(1)}, \tau_{ij}^{(2)}$ より、

$$A(w_i^r) = -(\tau_{ij}^{(1)} \nu_j^{(1)} + \tau_{ij}^{(2)} \nu_j^{(2)})|_{\gamma_{12}} \dots\dots\dots (23)$$

を計算する。

step 2

$$\mu_i^{r+1} = \mu_i^r + \alpha_r w_i^r \dots\dots\dots (24)$$

$$g_i^{r+1} = g_i^r - \alpha_r A(w_i^r) \dots\dots\dots (25)$$

$$\alpha_r = \frac{w_i^r g_i^r}{w_i^r A(w_i^r)} \dots\dots\dots (26)$$

$$w_i^{r+1} = g_i^{r+1} + \beta_r w_i^r \dots\dots\dots (27)$$

$$\beta_r = -\frac{g_i^{r+1} A(w_i^r)}{w_i^r A(w_i^r)} \dots\dots\dots (28)$$

step 3

収束（領域間境界で反力が釣り合うこと）すれば step 4 へ。未収束の際は、 $r \leftarrow r+1$ として step 1 へ戻る。（具体的な収束判定条件については解析例の項で述べる。）

step 4

領域間境界の変位の収束値 μ_i が得られたので、これを用いて再び式(3)～(7)および式(9)を解くことにより、全体の解が得られる。（step 0-step 3 では領域間境界の節点の解しか得られていないため、この step

が必要となる。）

本アルゴリズム中の $A(w)$ の計算に現れる、各領域における動的有限要素解析とは、後述する時間積分法を用いたものである。そこでは、直前の時間ステップにおける変位の値などのデータが境界条件として使われる。すなわち、step 0 における $A(\mu)$ の計算のときは各点の直前の変位等を用い、step 1 における $A(w)$ の計算のときは、それらはすべてゼロとして扱う。

2.3 離散化

2.3.1 空間離散化 式(13)の [] 内を空間座標に関して離散化し、マトリックス表示すると、

$$[M]\{\ddot{\mathbf{u}}\} + [C]\{\dot{\mathbf{u}}\} + [K]\{\mathbf{u}\} = \{\mathbf{F}\} \dots\dots\dots (29)$$

となる。ここで、 $[M]$ は質量マトリックス、 $[C]$ は減衰マトリックス、 $[K]$ は剛性マトリックス、 $\{\mathbf{F}\}$ は外力ベクトルであり、 $\{\ddot{\mathbf{u}}\}$ は加速度ベクトル、 $\{\dot{\mathbf{u}}\}$ は速度ベクトル、 $\{\mathbf{u}\}$ は変位ベクトルを表す。

2.3.2 時間離散化（直接時間積分法：陰解法）

動的問題の解析法としては直接時間積分法、モード解析法などがあるが、本研究では最もはん用性のある直接時間積分法を扱う。

まず、直接時間積分法の中でも、マトリックスの反転操作を必要とし、陽解法に比べ並列化が難しいとされる陰解法について、Newmark- β 法を例として考える。

Newmark- β 法は、 δ, β をパラメータとして、

$$\begin{aligned} & \left([K] + \frac{\delta}{\beta \Delta t_n} [C] + \frac{1}{\beta (\Delta t_n)^2} [M] \right) \{\mathbf{u}\}_{n+1} \\ &= \{\mathbf{F}\}_{n+1} + [M] \left\{ \left(\frac{1}{2\beta} - 1 \right) \{\dot{\mathbf{u}}\}_n \right. \\ & \quad + \frac{1}{\beta \Delta t_n} \{\dot{\mathbf{u}}\}_n + \frac{1}{\beta (\Delta t_n)^2} \{\mathbf{u}\}_n \Big\} \\ & \quad + [C] \left\{ \left(\frac{\delta}{2\beta} - 1 \right) \Delta t_n \{\dot{\mathbf{u}}\}_n \right. \\ & \quad \left. + \left(\frac{\delta}{\beta} - 1 \right) \{\dot{\mathbf{u}}\}_n + \frac{\delta}{\beta \Delta t_n} \{\mathbf{u}\}_n \right\} \dots\dots\dots (30) \end{aligned}$$

となる。ここで、 $\{\mathbf{u}\}_n, \{\mathbf{u}\}_{n+1}, \{\dot{\mathbf{u}}\}_n, \{\dot{\mathbf{u}}\}_{n+1}, \{\ddot{\mathbf{u}}\}_n, \{\ddot{\mathbf{u}}\}_{n+1}$ はそれぞれ、時刻 t_n, t_{n+1} における変位および速度、加速度ベクトルである。このように、Newmark- β 法は各部分領域での有限要素解析において、系の左辺マトリックスに剛性マトリックス $[K]$ を含むため、マトリックスの反転が必要な陰解法である。

陰解法（例：Newmark- β 法）の領域分割法のアルゴリズムをまとめると次のようになる。

- (0) $n \leftarrow 0$
- (1) 共役こう配法 (step 0 から step 4)
- (2) $n+1$ 時間ステップの解の出力
- (3) $n \leftarrow n+1$

(4) 最終時間ステップでなければ(1)へ戻る。

(5) 終了

共役こう配法の step 0 で行う各領域の有限要素解析は式(30)であり, step 1 における各領域の有限要素解析は式(30)の右辺をゼロとおいた式である。

図2にフローチャートを示した。時間積分のループの内側に共役こう配法のループを有している。

2・3・3 時間離散化(直接時間積分法:陽解法)

陽解法の場合, 各自由度の独立性を利用した部分領域への分割方法(例えば, 各節点はただ一つの領域に属し, 二つ以上の領域に共有されない。)により, 比較的容易に並列化ができる。それに対し, ここでは, 「領域間境界の節点変位を与え, それによって生じる反力の不釣合を共役こう配法等で解消して解を得る。」という領域分割法のフレームワーク内でも十分効率的に解けることを示す。

直接時間積分法として中心差分法を用いることにする。中心差分法は,

$$\begin{aligned} & \left(\frac{1}{(\Delta t)^2} [M] + \frac{1}{2\Delta t} [C] \right) \{u\}_{n+1} \\ &= \{F\}_n - \left([K] - \frac{2}{(\Delta t)^2} [M] \right) \{u\}_n \\ & - \left(\frac{1}{(\Delta t)^2} [M] - \frac{1}{2\Delta t} [C] \right) \{u\}_{n-1} \dots \dots \dots (31) \end{aligned}$$

となる。 $\{u\}_n, \{u\}_{n+1}$ は, 時刻 t_n, t_{n+1} における変位である。中心差分法は, 集中質量マトリックスを用い, 減衰マトリックスが対角マトリックスであれば, 左辺マトリックスが対角マトリックスとなり, マトリックスの反転操作が不要な陽解法となる。ただし, 陽解法は条件安定(それに対し一般に陰解法は絶対安定)であり, 適用には制限がある。

図2 動的領域分割法のアルゴリズム

陽解法の領域分割法においては, 各領域の有限要素解析(式(31))の際にマトリックスの反転が不要となり計算量が少なくなる。さらに, 式(31)から明らかに, $\{u\}_{n+1}$ の各成分は独立に求めることができるため, さきに示した共役こう配法のアルゴリズムの step 0 で得られた初期の不釣合量から直接に領域間境界上の変位を求めることができる。(当然, 共役こう配法を用いた, 陰解法と同じ定式化もできる。)

アルゴリズムは次のようになる。

演算子 A は式(14)で定義されているとする。

step 0E

$$\mu_i = \mu_i^0 (\mu_i^0 : \text{arbitrary}) \dots \dots \dots (32)$$

$$g_i = A(\mu_i^0) \dots \dots \dots (33)$$

式(33)の右辺は先と同じく, 部分領域間境界の変位を μ_i^0 として有限要素解析を行い, 式(14)より求める。この時, 領域間境界以外の節点については解が決まる。

step 1E

$$\mu_i^{\text{solution}} = \mu_i + \frac{g_i}{s_i} \dots \dots \dots (34)$$

ここで

$$s_i = \frac{1}{(\Delta t)^2} \sum_{k=1}^2 m_i^{(k)} + \frac{1}{2\Delta t} \sum_{k=1}^2 c_i^{(k)} \dots \dots \dots (35)$$

ただし, $m_i^{(k)}, c_i^{(k)}$ はそれぞれ, 領域間境界の i 番目の自由度に関する, k 領域の質量マトリックス $[M]$, 減衰マトリックス $[C]$ の成分である。これで領域間境界の節点変位が決まるので, step 0E で求めたほかの節点変位と合わせて全節点の変位が決まる。

陽解法の領域分割法のアルゴリズムをまとめると次のようになる。

(0) $n \leftarrow 0$

(1・1) step 0E

(1・2) step 1E

(2) $n+1$ 時間ステップの解の出力

(3) $n \leftarrow n+1$

(4) 最終時間ステップでなければ(1)へ戻る。

(5) 終了

2・4 動的領域分割法の特徴 以上のように定式化した動的領域分割法の特徴は次のようにまとめることができる。

a) 有限要素解析を個々の部分領域に分割して行うため, 記憶容量が少なく済み, 大規模解析に適している。

b) 各部分領域の有限要素解析という大きな単位で並列に処理でき, 高い並列効率が得られる。

c) 各部分領域の有限要素解析は従来のコードが使えるので, 資産の継承が可能であり, インプリメント

も容易である。

d) 領域数, 分割方法が任意に選べるため, はん用性が高く, ハードウェア依存性が低い。

3. 解析結果および考察

3.1 解析対象 例題として, 二次元弾性体の衝撃応答解析を行った。図3に要素分割および領域分割を示す。要素数は全体で256とし, $2 \cdot 4 \cdot 16 \cdot 64$ の領域に分割した四とりの場合について解析した。また, 領域分割数が4の場合については, 全体の要素数を $64 \cdot 1024$ に変えた解析も行った。有限要素は四角形4節点平面ひずみ要素である。材料定数は, ヤング率200 GPa, ポアソン比0.3, 質量密度 $7.88 \times 10^3 \text{ kg/m}^3$ とした。荷重は, 上端の片側半分に5 MPaのステップ荷重をかけた。減衰はないものとした。時間刻み $\Delta t = 1 \times 10^{-7}$ (秒)とし, 1000ステップを解析した。

3.2 解析結果 (陰解法: Newmark- β 法)

3.2.1 収束条件と精度 Newmark- β 法のパラメータは, $\beta = 0.25$ (定加速度法)とした。(減衰なしとした場合 δ は不要になる。)

陰解法では, 各時間ステップごとに, 共役こう配法により反復的に解を求めていく。適当な判定条件により反復を打ち切るのであるが, 本解析では, 次の判定式を用いた。

図3 要素および領域分割図

$$\frac{\max_i |g_i^r|}{\max_i |g_i^0|} < \epsilon \quad (36)$$

ここで, g_i^r は共役こう配法の r 回目の反復における節点反力ベクトルの i 番目の自由度の成分を示す。 ϵ が小さいほど正確な解析となるが, 共役こう配法の反復回数が増えて各時間ステップの計算時間が増大する。このため, ϵ の値は必要精度や時間ステップ数などによって適切に設定する必要がある。

まず, $\epsilon = 1 \times 10^{-7}$ として解析を行い, 従来法(全体を一つの領域として解析する)と比較した結果を図4に示す。ここでは, 図中に示したA点について各時間ステップにおける y 方向の変位を比較した。領域分割法による計算結果は従来法による結果と非常によく一致しており, 時間ステップが進んでも誤差はあまり増えていない。ときおり誤差が大きいステップが見られるのは, 領域分割法による解析値をA, 従来法による解析値をBとし, 誤差を $(A-B)/B \times 100$ と定義したため, Bがゼロに近い値をとるステップでは誤差が大きくなるように見えてしまうためである。各時間ステップにおける共役こう配法の反復回数を図5に示す。共役こう配法の反復回数はほとんど一定である。

図4 領域分割法と従来法の結果の比較

図5 領域分割法における共役こう配法の反復回数

これは、途中で外力が変化する条件でも同様であった。

図6に共役こう配法の収束状況を示す。動解析は時間ステップ=3での値である。参考値として静解析の結果も示してある。静解析の場合、共役こう配法の収束のスピードは各部分領域間の境界自由度数の平方根の値で決まるため⁽¹¹⁾、領域数が増加すると収束に向かうスピードがかなり低下している。一方、動解析は、領域分割数によらずに静解析に比べてかなり速いスピードで収束している。

各種物理量の収束の様子を図7に示す。時間ステップ=50での結果である。応力や変位も非常に速く収束していることがわかる。なお、このグラフでは6回程度の反復回数(残差ベクトルにして 1×10^{-4} 程度)で十分な精度が得られているようであるが、動的解析では時間ステップの進行にともない誤差が累積しやすいので、各回の収束判定は厳しくしたほうがよいと思われる。

そこで、収束判定条件 ε を変化させて1000時間ステップの解析を行い、各時間ステップにおける変位の値を比較した。結果を図8に示す。時間ステップが進むにつれて ε の影響が出てきている。ここでは、 $\varepsilon=1$

$\times 10^{-5}$ 程度でよい結果が得られている。しかし、本解析では、若干厳しい条件である、 $\varepsilon=1 \times 10^{-7}$ を採用することとした。

3・2・2 初期値選定と精度 各時間ステップにおいて共役こう配法の反復計算を行うのであるが、最初に与える領域間境界の変位の選び方は任意であるため、実際には次のように何とおりかの方法が考えられる。

(a) 直前の時間ステップにおける解を初期値とする(直前値法)。

(b) 直前の時間ステップにおける解(変位および速度・加速度)から外挿される値を初期値とする(予測値法)。

(c) ゼロを初期値とする(ゼロクリア法)。

なお、予測値法における外挿式には次式を用いた。

$$\{\mu\}_{n+1} = \{\mu\}_n + \{\dot{\mu}\}_n \Delta t_n + \frac{1}{2} \{\ddot{\mu}\}_n (\Delta t_n)^2 \quad \text{on } \gamma_{kl} \quad \dots\dots\dots (37)$$

これらについて評価を行ったところ、共役こう配法の反復回数はほぼ同じであった。直前値法および予測値法は、あらかじめ解の近似値を初期値として用いていると思われる、より速く収束するように考えられる。ところが、本解析で採用した収束判定条件は、式(36)のように「初期の不釣合量との比」という相対的な判定条件であり、また、図6からもわかるように収束のスピードが一定に近い。初期値をよくしても反復回数の改善にはあまり効果がないと考えられる。見方を変えると、本解析で採用した判定条件においては、 ε が同じであれば、ゼロクリア法よりも直前値法や予測値法が結果の誤差が小さくなる。このことは、 ε をある程度大きくとったときに顕著に現れる。三つの方法による結果を比較したところ、比較的大きな ε (たとえば、 $\varepsilon=0.1, 0.01$)に対して、ゼロクリア法の誤差が最も大きく、予測値法の誤差が最も小さいという結果

図6 共役こう配法の収束の様子

図7 物理量の収束の様子

図8 収束判定条件と精度

であった。直前値法はその中間であった。しかし、 ε が 1×10^{-3} 以下になると、精度のよい順序は同じであるが、三つの方法とも実用上十分な精度が得られた。以上のことを考慮すると、反復の初期値の与え方としては直前値法あるいは予測値法がよく、Newmark- β 法では予測値法が最も精度がよいと考えられるが、ほかの時間積分法（陰解法の中心差分法、Houbolt法など）においては、速度、加速度の計算が余分の手間になるなどの難点がある。それに対し、変位のみを用い、精度も悪くない直前値法はすべての時間積分法に共通して使える利点がある。本解析では直前値法を採用している。

3・2・3 領域分割数・要素数と収束性 領域分割数と共役こう配法の反復回数との関係を図9に示した。集中質量マトリックスを用いた場合、および整合質量マトリックスを用いた場合ともに静解析と比べて、反復回数ははるかに少なくなっている。静解析では領域分割数の増加（部分領域間境界の自由度数の増加：要素数256の場合、2領域分割では34自由度、64領域分割では364自由度であり、その平方根の比は3.3倍）に伴って反復回数が増加（2領域では31回、64領域では96回であり、その比は3.1倍）しているのに比

図9 領域分割数と共役こう配法の反復回数

図10 全要素数と共役こう配法の反復回数

べ、動解析の場合は反復回数の増加は非常に少なくなっている。

また、全要素数と共役こう配法の反復回数との関係を、領域分割数が4の場合について、図10に示した。ここでも、静解析が、全要素数の増加（部分領域間の自由度数の増加：領域分割数4の場合、64要素では32自由度、1024要素では128自由度であり、その平方根の比は2倍）に伴って反復回数が増加（64要素では29回、1024要素では63回であり、その比は2.17倍）しているのに比べ、動解析は、集中質量マトリックス、整合質量マトリックス、どちらの場合も反復回数の増加は非常に少ない。

領域分割法が動解析において、共役こう配法の収束スピードに、静解析とは全く異なった振る舞いを示すことは次のように説明される。

$Ax=b$ を共役こう配法で解く際の収束の速さは、マトリックス A の性質に依存する。領域分割法においては、共役こう配法にかけられる演算子 A は「領域間境界に変位条件を与え、各領域ごとに有限要素解析を行い領域間境界上での節点反力の和を取る」という複雑なものである。しかし、その性質は系全体を記述した方程式の左辺マトリックスの性質を強く反映すると考えるのが自然である。静解析の場合、系の方程式の左辺マトリックスは剛性マトリックス $[K]$ であるのに対し、動解析(Newmark- β 法)の場合には式(30)から明らかなように、系の左辺マトリックスは剛性マトリックス $[K]$ と質量マトリックス $[M]$ の線形結合になっている。すなわち、収束スピードの向上は、質量マトリックスにより全体としての左辺マトリックスの性質が改善されたことによるものであるといえる。

3・2・4 記憶容量および実行時間 最も記憶容量を必要とする剛性マトリックス $[K]$ の容量を表1に示す。（整合質量を用いる場合は $[M]$ も同じ容量を占める。）領域分割数を増やすことにより、所要記憶領域を削減でき、小さな容量で大規模な解析が可能となる。例えば、本解析で使用しているスカイライン法では、要素数が256のときには、全体を一つの領域とする従来法の場合は剛性マトリックス $[K]$ の記憶に必

表1 必要記憶容量（スカイライン法）

Number of Elements	256				
Number of Nodes	289				
Number of Domains	2	4	16	64	1*
[K] Size (per Domain)[KB]	69	30	5	1	207
[K] Size (total)[KB]	138	121	81	67	207
Ratio	0.67	0.58	0.39	0.32	1.00

* Conventional FEM

表 2 実行時間 (陰解法)

Time Integration Method	Newmark- β				
Number of Nodes	289				
Number of Domains	2	4	16	64	1 ^{*1}
CPU Time ^{*2} (s)	141	107	77	90	70

*1 Conventional FEM

*2 Total time of 50 Time Steps

表 3 実行時間 (陽解法)

Time Integration Method	Central Difference (Explicit)				
Number of Nodes	289				
Number of Domains	2	4	16	64	1 ^{*1}
CPU Time ^{*2} (s)	197	136	85	95	670

*1 Conventional FEM

*2 Total time of 1000 Time Steps

要なのは 207 KByte (倍精度) であるのに対し, 64 領域に分割した場合は 67 KByte であり, 従来法の 3 分の 1 で済むことになる。

実行時間を表 2 に示す。集中質量マトリックスを用い, $\epsilon=1 \times 10^{-7}$ の結果である。16 分割では従来法と同等の実行時間が得られている。1 CPU における動的領域分割法の実行時間は, 各部分領域の FEM 解析に要する時間, 領域数, 共役こう配法の反復回数, 境界条件の更新に要する時間などによって決まる。領域分割数を多くすれば, 各部分領域の FEM 解析時間は短縮されるが, 反復回数が増加 (本解析の場合, 64 領域では, 16 領域のときよりも反復回数が 1 回多い) し, 領域間境界の自由度の増加により境界条件の更新に要する時間が増大する。このため, 実行時間の面では, 問題の規模に依存して, 最適な領域分割数が存在することになる。

3・3 解析結果 (陽解法: 中心差分法) 時間ステップ数 1000 回の解析を行い, 従来法の結果と比較した結果, 従来法とよく一致した。

実行時間を表 3 に示す。共役こう配法の反復がないため, 領域分割によるマトリックスのサイズ縮小の効果により高速になっている。共役こう配法を用いた場合でも 5, 6 回の非常に少ない反復で収束するため,

従来法よりは高速である。

記憶容量に関しては陰解法と同様に, 領域を分割することにより大幅に必要な記憶容量が減少する。

4. 結 言

大規模・並列処理に適した解法である領域分割法を動的問題へ拡張し, 1 CPU 上で例題を通して有効性を検証した。

1 CPU 上の動的有限要素解析において, 領域分割法は,

(1) 精度は全体を 1 領域として解く従来法と変わらない。

(2) 解析に必要な計算機の記憶容量は従来法よりはるかに小さい。

(3) 実行速度については, 陰解法では領域分割数を選ぶことにより従来法と同等, 陽解法では領域分割数によらず従来法の数倍のスピードが得られる。のような特徴を有している。

さらに, 並列処理環境においては, 実行速度に関して一層の高速化が期待できる。

文 献

- (1) 村田・ほか 2 名, スーパーコンピュータ, (1985), 1-52, 丸善。
- (2) 津田, 数値処理プログラミング, (1988), 137-148, 岩波。
- (3) 三好・ほか 2 名, 第 3 回計算力学講演会講演論文集, **900**-69, (1990), 127-128。
- (4) 矢川・赤星, シミュレーション, **8**, (1989), 49-55。
- (5) 高橋(編), 並列処理機構, (1989), 1-66, 丸善。
- (6) Jordan, H., ほか 1 名, *Comput. Struct.*, **10** (1979), 21-29。
- (7) Farhat, C., ほか 1 名, *Int. J. Num. Methods Eng.*, **24** (1987), 1771-1792。
- (8) Yagawa, G., ほか 2 名, *Comput. Struct.*, **38** (1991), 615-625。
- (9) 吉岡・ほか 3 名, 機論, **57**-541, A (1991), 1964-1972。
- (10) Soneda, N., ほか 2 名, *Trans. 11th SMiRT Conf.*, B, (1991), 243-248。
- (11) 矢川・曾根田, パラレルコンピューティング, (1991), 41-44, 培風館。
- (12) Glowinski, R., ほか 2 名, *Comput. Meth. Appl. Mech. Eng.*, **40** (1983), 27-109。
- (13) 矢川, 共役こう配法, (1977), 127-133, 教育出版。