

3次元電磁波散乱問題に対するモーメント法のハードウェアアクセラレーション

高橋 徹[†] 戎崎 俊一[†]

近年、筆者らは分子動力学シミュレーション専用計算機 MDGRAPE-2 による境界型解法（モーメント法、境界要素法等）のベクトル並列計算手法を提案している。本論文ではその手法を 3次元電磁波散乱問題に対するモーメント法に適用することを目的とした。まず、計算コスト上ネックとなる電磁場積分方程式の境界積分項を、MDGRAPE-2 がベクトル並列計算可能な形式に帰着できることを示した。続いて、実装したモーメント法コードを、3枚の MDGRAPE-2s ボードを搭載した PC 上でテストした。その結果、本コードは従来のコードと比べて境界要素数が約 5,000 以上の散乱問題では 2 桁程度高速となった。

A Hardware Acceleration of the Method of Moments for the Electromagnetic Scattering Problems in Three Dimensions

TORU TAKAHASHI[†] and TOSHIKAZU EBISUZAKI[†]

In the past few years, the present authors have proposed a technique to vectorise and parallelise the boundary-type solver, such as the method of moments (MoM) and the boundary element method, using the special-purpose computer MDGRAPE-2, which is a hardware accelerator for molecular dynamics simulations. The purpose of this study is to apply the hardware accelerating technique to the MoM for the electromagnetic scattering problems in three dimensions. To this end, it is shown that the boundary integral terms in the electric and magnetic field integral equations, which require the highest computational cost in the MoM, are rewritten in the form which MDGRAPE-2 can vectorise and parallelise. The implemented code herein was tested on a PC with three MDGRAPE-2s boards. The results demonstrated that the present code was nearly 100 times faster than the conventional code in the scattering problems discretised with 5,000 or more boundary elements.

1. 序

モーメント法あるいは境界要素法といった境界型解法は、解析すべき領域の境界だけを離散化すればよい。そのうえ、無限に広がる領域を有限に近似する必要がない。そのため、外部問題を主に扱う波動解析（電磁場解析、音響解析等）に対して有力な数値解法である^{1)~3)}。ところが、境界型解法は大規模解析への適用が容易ではない。計算コストが少なくとも未知数の 2 乗に比例するからである。

この計算コストの問題に対する 1 つのアプローチが解法のベクトル計算・並列計算である。従来よりベクトル計算機を用いたベクトル化¹⁾、近年では PC クラスタ・SMP クラスタを用いた並列化に関する研究が多数ある^{4)~6)}。

本研究が提案するアプローチも解法のベクトル並列化にほかならない。他の研究と異なる点は、汎用の計算機ではなくロジック固定型の計算機（専用計算機）を利用することである。一般に、専用計算機は価格・消費電力・サイズあたりの性能は汎用計算機に優る。ところが融通はきかない。しかし、分子動力学（Molecular Dynamics、以後 MD と略す）シミュレーション専用計算機 MDGRAPE-2^{7),8)} が境界要素法のアクセラレータとなることを筆者らは見出している。つまり、境界要素法と MD 法との間のアナログと MDGRAPE-2 が持つ自由度（式 (18) の関数 g を MDGRAPE-2 ユーザが自由に設定できること）を利用すれば、境界要素法のベクトル並列計算が MDGRAPE-2 によって実行できる。これまで、ポテンシャル問題⁹⁾、周波数域スカラ波動問題⁹⁾、静弾性問題¹⁰⁾ に対する各境界要素法の高速計算に成功した。現在、時間域スカラ波動問題にも取り組んでいる¹¹⁾。

本論文の目的は、MDGRAPE-2 によるハードウ

[†] 独立行政法人理化学研究所

The Institute of Physical and Chemical Research
(RIKEN)

アクセラレーション手法を、3次元電磁波散乱問題に対するモーメント法に適用することである。その定式化を示し、実装したコードの性能を評価する。

一方で、著者らは高速多重極法^{12)~15)}等の高速アルゴリズムに従う境界型解法を専用計算機によって加速する手法も研究している^{16),17)}。今後、このような併用的加速手法を目下のモーメント法に対しても構成する必要がある。本論文はその構成法を与えるものではないが、その構成要素の1つになりうる。

本論文の構成は次のとおりである。2章で、誘電体による電磁波散乱問題の積分方程式による定式化と、モーメント法による解法を述べる。3章でMDGRAPE-2の概略を述べる。4章でMDGRAPE-2によるモーメント法のベクトル並列化手法を構成する。5章で解析コードの実装について述べる。6章でその解析コードの性能評価を示す。7章で本研究の総括と今後の課題を述べる。なお、既報^{9),10)}と重複する情報も、読者の便宜を図って適宜述べる。

2. 誘電体による電磁波散乱問題に対するモーメント法の定式化^{2),18),19)},

2.1 電磁場積分方程式

3次元電磁場内に誘電体が占める領域を D_1 、それを取り巻く無限領域を D_2 と表す(図1)。 D_1 と D_2 の境界を S ($:= \partial D_1 = \partial D_2$) と表し、 S は自己交差しないとする。今、電荷密度と電流密度は零とする(斉次な Maxwell 方程式を扱う)。また、電束密度(磁束密度)は電場 $E(x, t)$ (磁場 $H(x, t)$) に対して誘電率 ε (透磁率 μ) を比例定数とした比例関係にあるとする。ここに、 $x \in \mathbb{R}^3$ は場所、 $t \in [0, \infty)$ は時間である。さらに、電磁場は角周波数 ω で調和振動しているとする。その時間因子は $e^{-j\omega t}$ (j は虚数単位) とする。以後、電磁場の時間因子は省略し、その振幅を $E(x)$ および $H(x)$ と書く ($E, H: \mathbb{R}^3 \rightarrow \mathbb{C}^3$)。このとき、領域 D_ν ($\nu = 1, 2$) において Maxwell 方程式は次のように書ける ($x \in D_\nu$):

$$\begin{aligned} \nabla \cdot E_\nu(x) &= 0, & \nabla \times H_\nu(x) &= -j\omega\varepsilon_\nu E_\nu(x), \\ \nabla \cdot H_\nu(x) &= 0, & \nabla \times E_\nu(x) &= j\omega\mu_\nu H_\nu(x), \end{aligned} \quad (1)$$

ここに、領域ごとに区別すべき諸量には下添字 ν ($= 1, 2$) を付した。

次に(1)の境界条件を示す。今、 D_ν に対する外向き単位法線ベクトルを n_ν と表す。ここに、 $n_1 = -n_2$

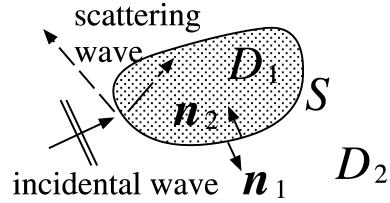


図1 無限領域 D_2 に取り囲まれた誘電体 D_1 (表面 S) による電磁波散乱問題

Fig.1 Electromagnetic scattering problems due to a dielectric scatterer D_1 with the boundary S surrounded by the infinite domain D_2 .

($=: n$) である。また、 S 上の表面電流密度 J_ν と表面磁流密度 M_ν を

$$\begin{aligned} J_\nu(x) &:= n_\nu(x) \times H_\nu(x), \\ M_\nu(x) &:= E_\nu(x) \times n_\nu(x) \quad x \in S \end{aligned}$$

と定義する。このとき、 S 上で電磁場の接線成分が連続であることから、境界条件は次のように書ける：

$$\begin{aligned} J_1(x) &= -J_2(x) (=: J(x)), \\ M_1(x) &= -M_2(x) (=: M(x)) \quad x \in S. \end{aligned} \quad (2)$$

また、 D_2 において電磁場は次の漸近条件を満たすとする：

$$\begin{aligned} E_2(x) &\rightarrow E^{\text{inc}}(x), & H_2(x) &\rightarrow H^{\text{inc}}(x) \\ & & & |x| \rightarrow \infty, \end{aligned} \quad (3)$$

ここに、 E^{inc} と H^{inc} は入射電場と入射磁場を各々表す ($E^{\text{inc}}, H^{\text{inc}}: D_2 \rightarrow \mathbb{C}^3$)。

式(1)~(3)によって定式化される電磁波散乱問題は、次の2つの境界積分方程式に帰着する：

$$\begin{aligned} n(x) \times \sum_{\nu=1}^2 \{(\mathcal{L}_\nu J)(x) - (\mathcal{K}_\nu M)(x)\} &= n(x) \times E^{\text{inc}}(x), \\ n(x) \times \sum_{\nu=1}^2 \{(\mathcal{K}_\nu J)(x) + \frac{\varepsilon_\nu}{\mu_\nu} (\mathcal{L}_\nu M)(x)\} &= n(x) \times H^{\text{inc}}(x) \quad x \in S, \end{aligned} \quad (4)$$

ここに、演算子 \mathcal{L}_ν と \mathcal{K}_ν ($\nu = 1, 2$) は次のように定義する：

$$\begin{aligned} (\mathcal{L}_\nu \varphi)(x) &:= \int_S \left[-j\omega\mu_\nu \varphi(y) \right. \\ &\quad \left. + \frac{1}{j\omega\varepsilon_\nu} \nabla_x \nabla_{S_y} \cdot \varphi(y) \right] G_\nu(x-y) dS_y, \quad (5) \\ (\mathcal{K}_\nu \varphi)(x) &:= \int_S \varphi(y) \times \nabla_x G_\nu(x-y) dS_y, \end{aligned} \quad (6)$$

ここに、 ∇_S は表面微分を表し、 φ は J あるいは M を表す。また、波数を $k_\nu := \omega\sqrt{\varepsilon_\nu\mu_\nu}$ と書くとき、関

本論文では、ベクトル $a \in \mathbb{R}^3$ or \mathbb{C}^3 について、その直交座標系での第 i 成分を $(a)_i$ と表す。

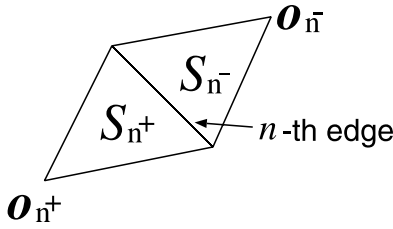


図 2 辺 n に関する記号の説明

Fig. 2 Explanation of symbols relating to the n -th edge.

数 G_ν は Helmholtz 方程式 $(\Delta + k_\nu^2)G_\nu = 0$ に対する基本解

$$G_\nu(x) := \frac{e^{jk_\nu|x|}}{4\pi|x|} = \frac{k_\nu}{4\pi}\Gamma(k_\nu|x|), \quad (7)$$

を表す。ここに、 $\Gamma(x) := e^{jx}/x$ と定義する。

2.2 モーメント法

式 (4) を未知の J と M についてモーメント法によって解く。モーメント法の離散化手法としては標準的なものを考える^{18),19)}。

今、境界 S を N 個の平面三角形境界要素で離散化する。要素 $b (= 1, \dots, N)$ を S_b と表し、その面積を $|S_b|$ と表す。また、 S を構成する辺の総数を N_e と表す。このとき、 S は自己交差しないので、 $N_e = 3N/2$ が成り立つ。さて、未知の J と M を、ある基底関数 $\{f_n\}$ を用いて次のように近似する ($x \in S$):

$$J(x) \approx \sum_{n=1}^{N_e} J_n f_n(x), \quad M(x) \approx \sum_{n=1}^{N_e} M_n f_n(x) \quad (8)$$

ここに、 $J_n, M_n \in \mathbb{C}$ は未知係数である ($n = 1, \dots, N_e$)。本論文では、Rao ら¹⁸⁾ が提案した次の基底関数を用いる (複合同順):

$$f_n(x) = \begin{cases} \frac{\pm l_n}{2|S_{n^\pm}|}(x - o_{n^\pm}) & \text{if } x \in S_{n^\pm}, \\ 0 & \text{otherwise,} \end{cases} \quad (9)$$

ここに、 l_n は辺 n の長さを表す。また、 n^+ と n^- は、辺 n を共有する 2 つの境界要素の番号を表す (図 2)。このとき、 $n^\pm \in [1, N]$ 。また、 S_{n^α} を構成する 3 つの頂点のうち、辺 n 上にない頂点を o_{n^α} と表す (図 2)。ここに現れる辺の符合 $\alpha (= +, -)$ は、未知係数 J および M の正負を決定するだけなので、辺ごとに任意に定めればよい (たとえば図 2 において符合を入れ換えて定義してもかまわない)。

式 (9) の基底関数による近似式 (8) を導入し、その基底関数をテスト関数にも採用する。このとき、式 (4) は J と M を未知係数とする次の $2N_e$ 元連立線形

方程式に帰着する ($m = 1, \dots, N_e$):

$$\sum_{\nu=1}^2 \sum_{n=1}^{N_e} (L_{m,n}^\nu J_n - K_{m,n}^\nu M_n) = E_m^{\text{inc}}, \quad (10)$$

$$\sum_{\nu=1}^2 \sum_{n=1}^{N_e} (K_{m,n}^\nu J_n + \frac{\varepsilon_\nu}{\mu_\nu} L_{m,n}^\nu M_n) = H_m^{\text{inc}},$$

ここに、

$$\begin{pmatrix} L_{m,n}^\nu \\ K_{m,n}^\nu \\ E_m^{\text{inc}} \\ H_m^{\text{inc}} \end{pmatrix} := \int_S f_m(x) \cdot \begin{pmatrix} (\mathcal{L}_\nu f_n)(x) \\ (\mathcal{K}_\nu f_n)(x) \\ E^{\text{inc}}(x) \\ H^{\text{inc}}(x) \end{pmatrix} dS_x. \quad (11)$$

式 (10) を解けば未知の J と M が求まる。この際、Gauss の消去法のような直接解法は $O(N^3)$ の計算コストと $O(N^2)$ のメモリを必要とする。そのため、大きな N に対しては現実的な解法とはいえない。そこで、反復解法を用いる。このときに計算コスト上ネットワークとなるのは、与えられたある J と M について式 (10) の左辺を構成すること、すなわち、次の 2 種類の行列・ベクトル積を計算することである:

$$\sum_{n=1}^{N_e} L_{m,n}^\nu \varphi_n, \quad \sum_{n=1}^{N_e} K_{m,n}^\nu \varphi_n \quad m = 1, \dots, N_e. \quad (12)$$

ここに、 $\nu = 1, 2$ および $\varphi = J, M$ である。この $O(N^2)$ の計算を高速に実行できれば、解法全体を高速に実行できる。

4 章において、ネットワークとなる式 (12) の行列・ベクトル積の計算を MDGRAPE-2 によってベクトル並列化する手法を与える。その準備として、式 (12) の具体形を導いておく。まず、その行列の係数 $L_{m,n}^\nu$ と $K_{m,n}^\nu$ の具体形を示す。ここで、基底 f_m の局在性により、式 (11) の右辺の積分範囲が実際には $S_{m^+} \cup S_{m^-}$ であることに注意する。その S_{m^α} ($\alpha = +, -$) 上の積分は、通例に従い、 P 次の Gauss-Legendre 積分公式^{3),20)} を用いて次のように数値的に評価する:

$$\int_{S_{m^\alpha}} (\cdot)(x) dS_x \approx \sum_{p=1}^P w_p^x (\cdot)(x_{m^\alpha,p}) |S_{m^\alpha}|, \quad (13)$$

ここに、 w_p^x は p 番目の重み係数、 $x_{m^\alpha,p}$ は要素 m^α 上に設けた p 番目の積分点を表す ($p = 1, \dots, P$)。式 (5), (6), (9), (13) より、式 (11) の $L_{m,n}^\nu$ と $K_{m,n}^\nu$ は次のように書ける ($m, n = 1, \dots, N_e$):

$$\begin{aligned}
L_{m,n}^\nu &\approx -j\omega\mu_\nu \sum_{\alpha=\pm} \sum_{p=1}^P \frac{\alpha l_m w_p^x}{2} (\mathbf{x}_{m^\alpha,p} - \mathbf{o}_{m^\alpha}) \\
&\cdot \sum_{\beta=\pm} \frac{\beta l_n}{2|S_{n^\beta}|} \int_{S_{n^\beta}} (\mathbf{y} - \mathbf{o}_{n^\beta}) G_\nu(\mathbf{x}_{m^\alpha,p} - \mathbf{y}) dS_y \\
&- \frac{1}{j\omega\varepsilon_\nu} \sum_{\alpha=\pm} \sum_{p=1}^P \alpha l_m w_p^x \\
&\sum_{\beta=\pm} \frac{\beta l_n}{|S_{n^\beta}|} \int_{S_{n^\beta}} G_\nu(\mathbf{x}_{m^\alpha,p} - \mathbf{y}) dS_y, \\
K_{m,n}^\nu &\approx \sum_{\alpha=\pm} \sum_{p=1}^P \frac{\alpha l_m w_p^x}{2} (\mathbf{x}_{m^\alpha,p} - \mathbf{o}_{m^\alpha}) \\
&\cdot \sum_{\beta=\pm} \frac{\beta l_n}{2|S_{n^\beta}|} \int_{S_{n^\beta}} (\mathbf{y} - \mathbf{o}_{n^\beta}) \\
&\times \nabla_x G_\nu(\mathbf{x}_{m^\alpha,p} - \mathbf{y}) dS_y. \quad (14)
\end{aligned}$$

続いて、上式の \mathbf{y} に関する S_{n^β} 上の積分を、式 (13) と同じ要領で、 Q 次の Gauss-Legendre 積分公式を用いて

$$\begin{aligned}
&\int_{S_{n^\beta}} G_\nu(\mathbf{x} - \mathbf{y}) dS_y \\
&\approx \sum_{q=1}^Q w_q^y \frac{k_\nu}{4\pi} \Gamma(k_\nu |\mathbf{x} - \mathbf{y}_{n^\beta,q}|) |S_{n^\beta}| \quad (15)
\end{aligned}$$

のように評価する (式 (7) を用いた)。ここに、 w_q^y は q 番目の重み係数、 $\mathbf{y}_{n^\beta,q}$ は S_{n^β} 上に設けた q 番目の積分点を表す ($q = 1, \dots, Q$)。式 (14) と (15) より、式 (12) は次のように書ける ($m = 1, \dots, N_e$):

$$\begin{aligned}
\sum_{n=1}^{N_e} L_{m,n}^\nu \varphi_n &\approx -j\omega\mu_\nu \sum_{\alpha=\pm} \sum_{p=1}^P \frac{\alpha l_m w_p^x}{2} \\
&(\mathbf{x}_{m^\alpha,p} - \mathbf{o}_{m^\alpha}) \cdot \mathbf{A}^{\nu,\varphi}(\mathbf{x}_{m^\alpha,p}) \\
&- \frac{1}{j\omega\varepsilon_\nu} \sum_{\alpha=\pm} \sum_{p=1}^P \alpha l_m w_p^x \Phi^{\nu,\varphi}(\mathbf{x}_{m^\alpha,p}), \quad (16a)
\end{aligned}$$

$$\begin{aligned}
\sum_{n=1}^{N_e} K_{m,n}^\nu \varphi_n &\approx \sum_{\alpha=\pm} \sum_{p=1}^P \frac{\alpha l_m w_p^x}{2} \\
&(\mathbf{x}_{m^\alpha,p} - \mathbf{o}_{m^\alpha}) \cdot \mathbf{B}^{\nu,\varphi}(\mathbf{x}_{m^\alpha,p}), \quad (16b)
\end{aligned}$$

ここに、次のスカラー $\Phi^{\nu,\varphi}$ とベクトル $\mathbf{A}^{\nu,\varphi}$ および $\mathbf{B}^{\nu,\varphi}$ を定義する:

$$\begin{aligned}
\Phi^{\nu,\varphi}(\mathbf{x}) &:= \sum_{n=1}^{N_e} \varphi_n \sum_{\beta=\pm} \frac{\beta l_n}{|S_{n^\beta}|} \sum_{q=1}^Q w_q^y \frac{k_\nu}{4\pi} \\
&\Gamma(k_\nu |\mathbf{x} - \mathbf{y}_{n^\beta,q}|) |S_{n^\beta}|, \quad (17a)
\end{aligned}$$

$$\begin{aligned}
\mathbf{A}^{\nu,\varphi}(\mathbf{x}) &:= \sum_{n=1}^{N_e} \varphi_n \sum_{\beta=\pm} \frac{\beta l_n}{2|S_{n^\beta}|} \sum_{q=1}^Q w_q^y \frac{k_\nu}{4\pi} \\
&(\mathbf{y}_{n^\beta,q} - \mathbf{o}_{n^\beta}) \Gamma(k_\nu |\mathbf{x} - \mathbf{y}_{n^\beta,q}|) |S_{n^\beta}|, \quad (17b)
\end{aligned}$$

$$\begin{aligned}
\mathbf{B}^{\nu,\varphi}(\mathbf{x}) &:= \sum_{n=1}^{N_e} \varphi_n \sum_{\beta=\pm} \frac{\beta l_n}{2|S_{n^\beta}|} \sum_{q=1}^Q w_q^y \frac{k_\nu}{4\pi} \\
&(\mathbf{y}_{n^\beta,q} - \mathbf{o}_{n^\beta}) \times \nabla_x \Gamma(k_\nu |\mathbf{x} - \mathbf{y}_{n^\beta,q}|) |S_{n^\beta}|. \quad (17c)
\end{aligned}$$

なお、式 (11) の E_m^{inc} と H_m^{inc} は次のように書けることに注意する:

$$\begin{aligned}
\begin{Bmatrix} E_m^{\text{inc}} \\ H_m^{\text{inc}} \end{Bmatrix} &\approx \sum_{\alpha=\pm} \sum_{p=1}^P \\
&\frac{\alpha l_m w_p^x}{2} (\mathbf{x}_{m^\alpha,p} - \mathbf{o}_{m^\alpha}) \cdot \begin{Bmatrix} \mathbf{E}^{\text{inc}}(\mathbf{x}_{m^\alpha,p}) \\ \mathbf{H}^{\text{inc}}(\mathbf{x}_{m^\alpha,p}) \end{Bmatrix}.
\end{aligned}$$

式 (12) を式 (16) のように離散化したとき、計算コスト上ネックとなる計算は、 $\Phi^{\nu,\varphi}(\mathbf{x}_{m^\alpha,p})$ 、 $\mathbf{A}^{\nu,\varphi}(\mathbf{x}_{m^\alpha,p})$ および $\mathbf{B}^{\nu,\varphi}(\mathbf{x}_{m^\alpha,p})$ の計算である。実際、これらの計算は $O(N)$ 個の評価点 \mathbf{x}_i ごとに長さ $N_e (= 3N/2)$ の総和を実行するという $O(N^2)$ の行列・ベクトル積になっていることが定義式 (17) から分かる (より明確には、後述の式 (20) ~ (22) から分かる)。ゆえに、高速に計算すべきものは、 $\Phi^{\nu,\varphi}(\mathbf{x}_{m^\alpha,p})$ 、 $\mathbf{A}^{\nu,\varphi}(\mathbf{x}_{m^\alpha,p})$ および $\mathbf{B}^{\nu,\varphi}(\mathbf{x}_{m^\alpha,p})$ の 3 種類の行列・ベクトル積に整理される。

ここで、 $\Phi^{\nu,\varphi}$ 、 $\mathbf{A}^{\nu,\varphi}$ および $\mathbf{B}^{\nu,\varphi}$ の引数は $x_{b,p}$ ($b = 1, \dots, N$; $p = 1, \dots, P$) あるいは ξ_i ($i = 1, \dots, NP$) とすれば十分であることに注意する。なぜならば、 $m^\pm \in [1, N]$ ($m = 1, \dots, N_e$) だからである。ここで、 ξ_i は便宜上次のように定義する:

$$\xi_{P(b-1)+p} := x_{b,p} \quad b = 1, \dots, N; \quad p = 1, \dots, P.$$

3. MDGRAPE-2 の概要^{7),8)}

MDGRAPE-2 は、MD シミュレーションにおいて計算コスト上ネックとなる分子間相互作用 (クーロン力等) の計算を高速に実行するために開発された計算機である。MD シミュレーション²¹⁾ はもとより、天体シミュレーション²²⁾ やプラズマシミュレーション²³⁾ 等を含む多体問題用ハードウェアアクセラレータとして利用されてきた。以下、モーメント法への応用に際して必要となる MDGRAPE-2 の情報を述べる。

まず、MDGRAPE-2 の計算機能を述べる。それは、与えられた $\{\mathbf{x}_i\}_{i=1}^m \subset \mathbb{R}^3$ 、 $\{\mathbf{y}_j\}_{j=1}^n \subset \mathbb{R}^3$ 、 $\alpha \in \mathbb{R}$ 、 $\{b_j\}_{j=1}^n \subset \mathbb{R}$ (ここに、 $m, n \in \mathbb{N}$) と関数 $g: \mathbb{R} \rightarrow \mathbb{R}$ に対して、次の $\{\phi(\mathbf{x}_i)\}_{i=1}^m \subset \mathbb{R}$ あり

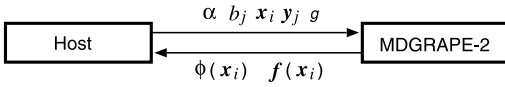


図 3 計算システムの構成
Fig. 3 Configuration of computing system.

るいは $\{f(x_i)\}_{i=1}^m \subset \mathbb{R}^3$ を計算することである：

$$\phi(x_i) := \sum_{j=1}^n b_j g(\alpha |x_i - y_j|^2), \quad (18a)$$

$$f(x_i) := \sum_{j=1}^n b_j g(\alpha |x_i - y_j|^2)(x_i - y_j). \quad (18b)$$

なお、MDGRAPE-2 を式 (18a) および (18b) の計算が実行できるように準備した状態を potential-mode および force-mode と各々いう。

さて、MDGRAPE-2 に式 (18) を実行させるためには、host 上で所定の手続きをプログラミングする必要がある。ここで、host とは MDGRAPE-2 と通信可能な汎用計算機である (図 3)。その手続きとは、データ ($\{x_i\}_{i=1}^m, \alpha, \{y_j\}_{j=1}^n, \{b_j\}_{j=1}^n, g$) の転送、計算開始の通知、計算結果 ($\{\phi(x_i)\}_{i=1}^m$ あるいは $\{f(x_i)\}_{i=1}^m$) の回収である。以後、これらの手続きを経て MDGRAPE-2 に式 (18a) (あるいは (18b)) の計算を実行させることを、potential-mode で (あるいは force-mode で) MDGRAPE-2 を drive するという。

次に、式 (18) の処理方式を簡単に述べる。MDGRAPE-2 は式 (18) の計算に特化したパイプラインを複数有している (図 4)。host から転送されたデータのうち、 $\alpha, \{y_j\}_{j=1}^n, \{b_j\}_{j=1}^n$ および g はすべてのパイプラインに共有される。一方、 x_i は各パイプラインに重複することなく均等に割り当てられる。このようなデータ配分の下ですべてのパイプラインが同時に作動する。これは、式 (18) を添字 j に関してベクトル処理 (すなわち、パイプライン処理) し、添字 i に関してパラレル処理 (すなわち、複数のパイプラインの並列作動) することを意味する。まとめると、MDGRAPE-2 は式 (18) に特化したベクトル並列計算機にほかならない。

続いて、式 (18) の計算の実行時間を概算する。その時間を t_{drive} と表すと、 $t_{drive} = t_{comm} + t_{calc}$ と書ける。ここに、 t_{comm} は、host から MDGRAPE-2 へのデータの転送に要する時間、 t_{calc} はパイプラインの作動時間である。なお、MDGRAPE-2 からのデータの回収はパイプラインの作動と並行して行われるので考えない。さて、式 (18) より $t_{calc} = O(mn)$ および $t_{comm} = O(n)$ であることが分かる。今、問題の

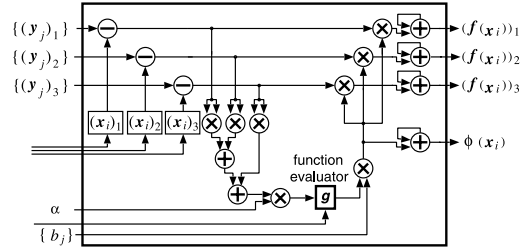


図 4 パイプラインのブロック線図
Fig. 4 Block diagram of a pipeline.

規模が大きい場合、すなわち、 m と n がともに大きい場合には t_{comm} は無視できて、次のように評価できる：

$$t_{drive} \sim t_{calc} \approx \frac{mn}{N_{pipe} \cdot f_{pipe}}, \quad (19)$$

ここに、 N_{pipe} は利用パイプライン数、 f_{pipe} はパイプラインの動作周波数である。

最後に、関数 g の評価について述べる (既報⁹⁾ も参照せよ)。関数 $g(x)$ は、MDGRAPE-2 ユーザが区間 $[x_{lower}, x_{upper}]$ (ここに、 $0 < x_{lower} < x_{upper}$) 上からとった 1,025 個のサンプルの区分 4 次多項式補間によって評価される。ここで、 $x \notin [x_{lower}, x_{upper}]$ のときは $g(x) = 0$ と打ち切られる。なお、ある g を MDGRAPE-2 に転送するとは、実際にはそのサンプルを転送することである。

4. MDGRAPE-2 によるベクトル並列化

モーメント法の実行において計算コスト上ネックとなる式 (16) の行列・ベクトル積、すなわち、 $\Phi^{\nu,\varphi}(\xi_i)$ 、 $A^{\nu,\varphi}(\xi_i)$ および $B^{\nu,\varphi}(\xi_i)$ ($i = 1, \dots, NP$) を、MDGRAPE-2 によってベクトル並列化するための定式化を示す。すなわち、それらが MDGRAPE-2 によって計算可能な形式 (18a) あるいは (18b) に帰着することを示す (4.1~4.3 節)。最後に、いくつかの注意点を述べる (4.4 節)。

以下、領域番号に関する添字 $\nu (= 1, 2)$ と表面密度に関する添字 $\varphi (= J, M)$ は固定して考える。

4.1 $\Phi^{\nu,\varphi}$ の場合

式 (17a) の右辺を見て分かるように、 $\Phi^{\nu,\varphi}$ は (グローバルな) 辺番号に関する添字 $n (= 1, \dots, N_e)$ の総和と、辺 n を共有する要素 $S_{n\beta}$ に関する添字 $\beta (= +, -)$ の総和とによって表されている。このとき、ある要素はその 3 つの辺から各々 1 回、合計 3 回参照される。これはその要素上の (数値) 積分を 3 回繰り返し実行することを意味する。これを 1 回で済ますためには、要素番号に関する添字 $b (= 1, \dots, N)$

の総和と、各要素のローカルな辺番号に関する添字 e ($= 1, 2, 3$) の総和とによって、 $\Phi^{\nu,\varphi}$ を次のように書き改めればよい:

$$\Phi^{\nu,\varphi}(\xi_i) = \sum_{b=1}^N \sum_{e=1}^3 \frac{\varphi_{(b,e)} \sigma(b,e) l_{(b,e)}}{|S_b|} \sum_{q=1}^Q w_q^y \frac{k_\nu}{4\pi} \Gamma(k_\nu |x - \mathbf{y}_{b,q}|) |S_b|,$$

ここに、 (b,e) は、要素 b のローカルな辺 e が指す辺に付されているグローバルな辺番号 ($1, \dots, N_e$) を表す. この写像 (\cdot, \cdot) は容易に構成できる. また、 $\sigma(b,e)$ は、 $(b,e)^\pm = b$ ならば ± 1 を、 $(b,e)^\pm \neq b$ ならば ∓ 1 を返す関数である (複合同順). この関数は式 (9) の右辺の符号を定めれば一意に定まる. 続いて、

$$\eta_{Q(b-1)+q} := \mathbf{y}_{b,q},$$

$$\phi_{Q(b-1)+q}^{\nu,\varphi} := \sum_{e=1}^3 \frac{\varphi_{(b,e)} \sigma(b,e) l_{(b,e)} w_q^y k_\nu}{4\pi}$$

と定義して、添字 b と q に関する 2 つの総和を 1 つにまとめて添字 j の総和とすると、 $\Phi^{\nu,\varphi}$ は結局次のように書ける ($i = 1, \dots, NP$):

$$\Phi^{\nu,\varphi}(\xi_i) = \sum_{j=1}^{NQ} \Gamma(k_\nu |\xi_i - \eta_j|) \phi_j^{\nu,\varphi}. \quad (20)$$

上式は、諸量が複素数であることを除けば、式 (18a) の形式となっていることが分かる.

式 (20) が実際に式 (18a) の形式となり、MDGRAPE-2 によって計算できることを示す. 今、式 (20) の右辺を、MDGRAPE-2 が実数しか取り扱えないことを考慮して、

$$\sum \operatorname{Re}[\Gamma] \operatorname{Re}[\phi] - \sum \operatorname{Im}[\Gamma] \operatorname{Im}[\phi]$$

$$+ j \sum \operatorname{Re}[\Gamma] \operatorname{Im}[\phi] + j \sum \operatorname{Im}[\Gamma] \operatorname{Re}[\phi]$$

と 4 つの総和で表す. このとき、各々の総和は式 (18a) の形式をしている. 実際、

$$m = NP, \quad n = NQ, \quad \alpha = k_\nu^2,$$

$$\{\mathbf{x}_i\}_{i=1}^m = \{\xi_i\}_{i=1}^{NP}, \quad \{\mathbf{y}_j\}_{j=1}^n = \{\eta_j\}_{j=1}^{NQ},$$

$$\{b_j\}_{j=1}^n = \{\operatorname{Re}[\phi_j^{\nu,\varphi}]\}_{j=1}^{NQ} \text{ or } \{\operatorname{Im}[\phi_j^{\nu,\varphi}]\}_{j=1}^{NQ},$$

$$g(x) = \operatorname{Re}[\Gamma(\sqrt{x})] \text{ or } \operatorname{Im}[\Gamma(\sqrt{x})]$$

を式 (18a) に代入したものはそれらの総和と一致する. したがって、MDGRAPE-2 を potential-mode で 4 回 drive すれば $\Phi^{\nu,\varphi}$ を計算できる.

4.2 $A^{\nu,\varphi}$ の場合

$\Phi^{\nu,\varphi}$ と同様にすれば、式 (17b) の $A^{\nu,\varphi}$ は次のように書ける ($i = 1, \dots, NP$):

$$A^{\nu,\varphi}(\xi_i) = \sum_{j=1}^{NQ} \Gamma(k_\nu |\xi_i - \eta_j|) a_j^{\nu,\varphi}, \quad (21)$$

ここに、

$$a_{Q(b-1)+q}^{\nu,\varphi} := \sum_{e=1}^3 \frac{\varphi_{(b,e)} \sigma(b,e) l_{(b,e)} w_q^y k_\nu}{8\pi} (\mathbf{y}_{b,q} - \mathbf{o}_{(b,e)\sigma(b,e)}).$$

続いて、式 (21) が MDGRAPE-2 によって計算できることを示す. 今、式 (21) の第 l ($= 1, 2, 3$) 成分 $(A^{\nu,\varphi})_l$ を 4 つの総和で表すと、各々の総和は式 (18a) の形式となる. 実際、

$$\{b_j\}_{j=1}^n = \{\operatorname{Re}[(a_j^{\nu,\varphi})_l]\}_{j=1}^{NQ} \text{ or } \{\operatorname{Im}[(a_j^{\nu,\varphi})_l]\}_{j=1}^{NQ},$$

$$g(x) = \operatorname{Re}[\Gamma(\sqrt{x})] \text{ or } \operatorname{Im}[\Gamma(\sqrt{x})]$$

を式 (18a) に代入すれば分かる (l は固定; 省略した量は $\Phi^{\nu,\varphi}$ の場合と同じ). したがって、MDGRAPE-2 を potential-mode で 4 回 drive すれば $(A^{\nu,\varphi})_l$ を計算できる. すべての l について繰り返せば、12 回の drive によって $A^{\nu,\varphi}$ を計算できる.

4.3 $B^{\nu,\varphi}$ の場合

同様に、式 (17c) の $B^{\nu,\varphi}$ は次のように書ける ($i = 1, \dots, NP$):

$$B^{\nu,\varphi}(\xi_i) = \sum_{j=1}^{NQ} b_j^{\nu,\varphi} \times \Psi(k_\nu |\xi_i - \eta_j|) (\xi_i - \eta_j), \quad (22)$$

ここに、 $\Psi(x) := (jx - 1)e^{jx}/x^3$,

$$b_{Q(b-1)+q}^{\nu,\varphi} := \sum_{e=1}^3 \frac{\varphi_{(b,e)} \sigma(b,e) l_{(b,e)} w_q^y k_\nu^3}{8\pi} (\mathbf{y}_{b,q} - \mathbf{o}_{(b,e)\sigma(b,e)}).$$

式 (22) が MDGRAPE-2 によって計算できることを示す. まず、次のような $C^{\nu,\varphi,l}$ ($l = 1, 2, 3$) を定義する:

$$C^{\nu,\varphi,l}(\xi_i) := \sum_{j=1}^{NQ} (b_j^{\nu,\varphi})_l \Psi(k_\nu |\xi_i - \eta_j|) (\xi_i - \eta_j).$$

さて、 $C^{\nu,\varphi,l}$ の右辺をこれまでと同様に 4 つの総和に分解する. このとき、各々の総和は

$$\{b_j\}_{j=1}^n = \{\operatorname{Re}[(b_j^{\nu,\varphi})_l]\}_{j=1}^{NQ} \text{ or } \{\operatorname{Im}[(b_j^{\nu,\varphi})_l]\}_{j=1}^{NQ},$$

$$g(x) = \operatorname{Re}[\Psi(\sqrt{x})] \text{ or } \operatorname{Im}[\Psi(\sqrt{x})]$$

とすれば式 (18b) に帰着する. したがって、固定した l について MDGRAPE-2 を force-mode で 4 回 drive すれば $C^{\nu,\varphi,l}$ を計算できる. すべての l について繰り返せば、計 12 回の drive で $C^{\nu,\varphi,1}$, $C^{\nu,\varphi,2}$ および $C^{\nu,\varphi,3}$ を計算できる. このとき、目的の $B^{\nu,\varphi}$ は次

式によって計算できる：

$$B^{\nu,\varphi} = \begin{pmatrix} (C^{\nu,\varphi,2})_3 - (C^{\nu,\varphi,3})_2 \\ (C^{\nu,\varphi,3})_1 - (C^{\nu,\varphi,1})_3 \\ (C^{\nu,\varphi,1})_2 - (C^{\nu,\varphi,2})_1 \end{pmatrix}.$$

4.4 注意事項

総 drive 回数について

式 (10) の左辺を計算するためには、 $\nu (= 1, 2)$ と $\varphi (= J, M)$ に関する 4 つの組合せをすべて考慮する必要がある。したがって、MDGRAPE-2 の総 drive 回数は 4.1~4.3 節で述べた drive 回数を合計して 4 倍したもので、つまり、112 である。ただし、式 (20)~(22) を MDGRAPE-2 に計算させる方法は 1 通りではないので、112 が最小の drive 回数であるとは限らない。今後、改良の可能性もある。

drive の順番について

4.1~4.3 節で示したように、MDGRAPE-2 に転送すべき 4 種のデータ (式 (18) の α , $\{y_j\}_{j=1}^n$, $\{b_j\}_{j=1}^n$ および g) は drive ごとにすべて異なるわけではない。MDGRAPE-2 は最後に転送されたデータを次のデータが転送されるまでデータ種ごとに保持できる (ただし、同じ mode で drive する場合に限る) ので、drive の順番を考慮し、不必要なデータ転送を避けるべきである。本論文では、図 5 に示す順序で式 (10) の左辺を計算する。図中で、「for ~ end」は繰返しを、「send X 」はある量 X を host から MDGRAPE-2 へ転送することを、そして「drive」は MDGRAPE-2 の drive (および結果の回収) を表す。データ転送が段階的に行われていることに注意する。

特異積分について

便宜上、式 (15) の積分が特異積分となる場合、すなわち、評価点 x が被積分要素 $S_{n\beta}$ 上にある場合の処理については述べなかった。実際には、特異積分は解析的に実行する必要がある。同時に、式 (15) の数値積分に基づいて誘導された式 (20)~(22) の右辺の総和から、その特異積分が寄与する項を差し引く必要がある。これらの処理は host によって実行する。この処理の計算コストは評価点の個数のオーダ、すなわち、 $O(NP)$ にすぎない。

関数評価の打ち切りに関する補正について

3 章の最後で述べたことにより、式 (20)~(22) のあるペア ξ_i と η_j がきわめて近い場合、関数 g の引数の下限 x_{lower} における打ち切りが発生しうる。打ち切られたペアの寄与は host で補う必要がある。このようなペアを host が効率良く探索できるように、 ξ_i ごとに適当な近傍境界 $\mathcal{N}(\xi_i)$ を設ける必要がある。本

Left-hand side of Eq. (10) routine:

for $\nu = 1$ or 2

for $\varphi = J$ or M

call $L^\nu\varphi$ and $K^\nu\varphi$ routines;

end

end

$L^\nu\varphi$ routine:

invoke MDGRAPE-2 in potential-mode;

for $g(x) = \text{Re}[\Gamma(\sqrt{x})]$ or $\text{Im}[\Gamma(\sqrt{x})]$

send $g(x)$, $\alpha = k_\nu^2$, and $\{y.\} = \{\eta.\}$;

for $\{b.\} = \{\text{Re}[\phi^{\nu,\varphi}]\}$ or $\{\text{Im}[\phi^{\nu,\varphi}]\}$

send $\{b.\}$;

drive to compute a part of $\Phi^{\nu,\varphi}$ in Eq. (20);

end

for $l = 1, 2$, or 3

for $\{b.\} = \{\text{Re}[(\mathbf{a}^{\nu,\varphi})_l]\}$ or $\{\text{Im}[(\mathbf{a}^{\nu,\varphi})_l]\}$

send $\{b.\}$;

drive to compute a part of $(\mathbf{A}^{\nu,\varphi})_l$

in Eq. (21);

end

end

end

$K^\nu\varphi$ routine:

invoke MDGRAPE-2 in force-mode;

for $g(x) = \text{Re}[\Psi(\sqrt{x})]$ or $\text{Im}[\Psi(\sqrt{x})]$

send $g(x)$, $\alpha = k_\nu^2$, and $\{y.\} = \{\eta.\}$;

for $l = 1, 2$, or 3

for $\{b.\} = \{\text{Re}[(\mathbf{b}^{\nu,\varphi})_l]\}$ or $\{\text{Im}[(\mathbf{b}^{\nu,\varphi})_l]\}$

send $\{b.\}$;

drive to compute a part of $\mathbf{C}^{\nu,\varphi,l}$,

thus $\mathbf{B}^{\nu,\varphi}$ in Eq. (22);

end

end

end

図 5 式 (10) の左辺を計算するためのデータ転送と drive の順序
Fig. 5 Order of data-transfers and drives to compute the left-hand side of Eq. (10).

論文では既報⁹⁾に従う。つまり、境界要素の 8 分木構造を用いて、 ξ_i を含む第 1 近傍 cell を $\mathcal{N}(\xi_i)$ と定義する。

一方、あるペア ξ_i と η_j がきわめて遠い場合にも、 g の引数の上限 x_{upper} における打ち切りが発生しうる。これを host で補正することはできる。しかし、打ち切られたペアを探索するためのコストは $O(N^2)$ となってしまう。この探索は解析全体のボトルネックとなりうるので受け入れられない。したがって、境界 S

の空間的スケールと取り扱う波数 $\max(k_1, k_2)$ とに応じて、十分大きな x_{upper} をあらかじめ選択する必要がある。

5. 解析コードの実装

前章を基にして、MDGRAPE-2 によってベクトル並列化されたモーメント法コードを開発した。以後、MD2 コードと呼ぶ。MD2 コードの元となるコードは、2 章の定式化に従い、反復解法を採用し、スカラ計算機上での実行を想定して並列化されていない。このコードを従来コードと呼ぶ。MD2 コードは、従来コードが含む式 (10) の左辺の行列・ベクトル積ルーチンを改変して得た。すなわち、そのルーチンを式 (20) ~ (22) の $\Phi^{\nu,\varphi}$ 、 $A^{\nu,\varphi}$ および $B^{\nu,\varphi}$ についていったん整理し、各々が 4.1 ~ 4.3 節の要領で MDGRAPE-2 の drive によって計算されるようにコーディングした (図 5 を参照)。その際、MDGRAPE-2 のユーザーアプリケーションインタフェースを提供する M2library²⁴⁾ を利用し、Fortran77 で記述した。

連立線形方程式 (10) の解法にはリスタート版 GMRES 法²⁵⁾ を用いた。前処理には point-Jacobi 法²⁶⁾ を用いた。この際、CERFACS が配布する GMRES 法コード^{5), 27)} を用いた。このコードは、式 (10) の左辺の行列・ベクトル積ルーチン、式 (10) の右辺のベクトル、前処理ルーチン、および、諸パラメータ (反復停止条件、リスタート回数等) が引数となるようにブラックボックス化されている。

式 (18) の関数 g のサンプリングにはユーティリティソフト gentable²⁸⁾ を用いた。サンプリングすべき関数 g は 4.1 ~ 4.3 節で示された $\text{Re}[\Gamma(\sqrt{x})]$ 、 $\text{Im}[\Gamma(\sqrt{x})]$ 、 $\text{Re}[\Psi(\sqrt{x})]$ および $\text{Im}[\Psi(\sqrt{x})]$ の 4 つである。これらは既報⁹⁾ において Helmholtz 方程式を扱った際に必要となったものと同じである。そこで、既報と同じく $x_{\text{lower}} = 2^{-18}$ および $x_{\text{upper}} = 2^{14}$ としてサンプリングを行った。

6. 性能評価

5 章で開発した MD2 コードの性能を、従来コードとの比較によって評価した。

評価にあたり、次の散乱問題を解いた：原点を中心とする半径 1 の球形の誘電体 (D_1) を考える。この誘電体に対して、 e_3 方向に進行する平面電磁波 ($E^{\text{inc}}(x) = e^{jk_2x_3} e_1$ 、 $H^{\text{inc}}(x) = e^{jk_2x_3} e_2$) を入射する。ここに、 e_l は第 l 成分が 1 で、その他の成分が 0 であるような単位ベクトルである。この問題の厳密解はよく知られている²⁾。

誘電体およびその周囲 (D_2) の誘電率と透磁率は、適当に無次元化して、 $\varepsilon_1 = 2$ 、 $\varepsilon_2 = \mu_1 = \mu_2 = 1$ とした。また、角周波数 ω を 1 とした。このとき、波数は $k_1 = 1/\sqrt{2}$ 、 $k_2 = 1$ となる。

誘電体表面 S の要素分割は 6 通りとした。要素数 N は小さいものから 80, 320, 1,280, 5,120, 20,480 および 81,920 である。このとき、最大の N における未知数の数 ($2N_e$) は 245,760 である。また、Gauss-Legendre 積分公式の次数 P と Q はともに 1 とした。なお、 $P = Q$ のとき、本解法は Galerkin 法にほかならない。

両コードを実行した計算機は、Xeon 2.66 GHz を CPU とする PC である。使用 OS は Linux 2.4.20-8smp であり、両コードは Intel Fortran Compiler version 9.0²⁹⁾ によってコンパイルした。最適化のためのオプションは -O3 -xN とした。

上記の PC には、MDGRAPE-2 の実装の 1 つである MDGRAPE-2s PCI ボード³⁰⁾ を 3 枚装着した。1 枚のボードは 16 本のパイプラインを持つ。各パイプラインの動作周波数 f_{pipe} は 100 MHz である。なお、ボード 1 枚あたりの消費電力は約 16 W であり、PCI バス以外による電力供給は必要としない。

GMRES 法の利用に際して、そのリスタート回数は 2 GB のメモリの範囲で最大となるようにした。このとき、最大の N については 493 であった。解の初期推定値は 0 とした。また、反復の停止条件は、初期残差に対する相対残差が 10^{-5} を超えるときとした。

以上の設定の下で、 N ごとに両コードを実行した。まず、図 6 に両コードの計算時間を N に対してプロットする (図中の記号 “MD2” は MD2 コードを、“conv” は従来コードを表す。以下の図表も同様)。ここで、MD2 コードにおける使用ボード数は 3 ($N_{\text{pipe}} = 48$) である。なお、従来コードは $N \leq 5,120$ の範囲で実行した。同図には、従来コードに対する MD2 コードのスピードアップ (前者の計算時間を後者のそれで除した値) をあわせて示した。 $N \gtrsim 200$ において MD2 コードは従来コードよりも高速であり、 $N = 5,120$ では 86 倍の高速化が達成された。 $N \gtrsim 5,000$ でも同程度の高速化が見込まれる。

図 7 に GMRES 法の反復あたりの計算時間を示す。ここで、反復回数は両コードともにほぼ同じであった (表 1)。同図には、式 (19) から予測される計算時間、すなわち、

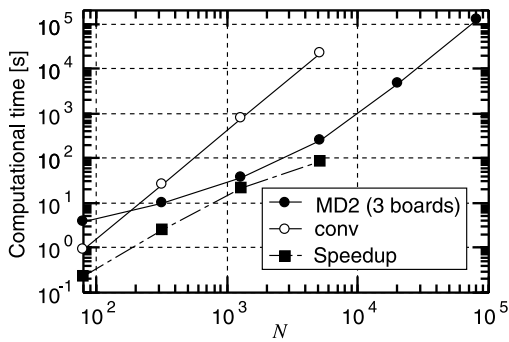


図 6 計算時間とスピードアップ
Fig.6 Computational time and speedup.

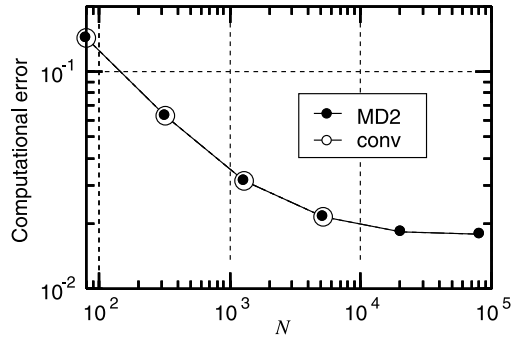


図 8 計算誤差
Fig.8 Computational error.

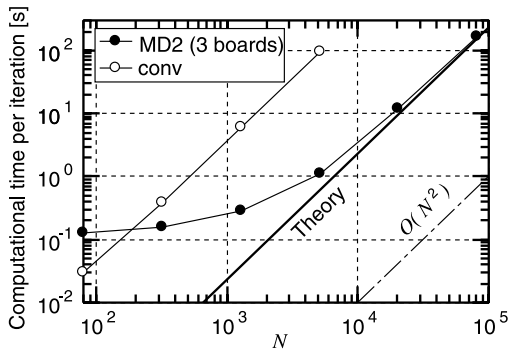


図 7 反復あたりの計算時間
Fig.7 Computational time per iteration.

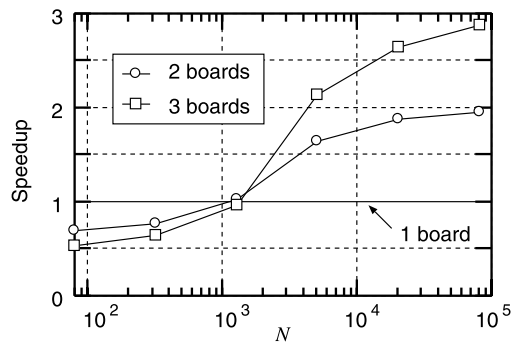


図 9 1 ボード使用時に対する 2 ボードおよび 3 ボード使用時のスピードアップ
Fig.9 Speedup for 2 and 3 boards against 1 board.

表 1 GMRES 法が収束するまでの反復回数

Table 1 Number of iterations to convergence of GMRES method.

N	80	320	1,280	5,120	20,480	81,920
MD2	31	65	130	226	393	736
conv	31	65	130	224	—	—

$$\begin{aligned}
 112 \times t_{\text{drive}} &= 112 \times \frac{NP \cdot NQ}{N_{\text{pipe}} \cdot f_{\text{pipe}}} \\
 &= 112 \times \frac{N \cdot 1 \cdot N \cdot 1}{48 \cdot 100 \times 10^6} \approx 2.3 \times 10^{-8} N^2 \text{ [s]}
 \end{aligned}$$

をあわせて示した(図中の“Theory”). N が大きい範囲で予測時間と実測時間はよく一致している.

次に、両コードの計算誤差を図 8 に示す. ここで、誤差とは、各要素の図心において計算される表面密度 J および M の厳密解と数値解(数値解は式(10)の解を式(8)に代入すれば得られる)の最大絶対誤差である. 両者はよく一致していることが分かる.

最後に、1 ボード使用時に対する 2 ボードおよび 3 ボード使用時のスピードアップを調べた(図 9). 2 ボード使用するならば $N \gtrsim 2 \times 10^4$ において、3 ボード使用するならば $N \gtrsim 8 \times 10^4$ においてパイプラインが理想値の約 9 割の効率で機能していることが分

かる.

他方、 $N \lesssim 10^3$ ではボード数が多いほどパフォーマンスは劣っている. これは式(18)の関数 g の通信に起因すると思われる. すなわち、式(18)の $\{b_j\}_{j=1}^n$ や $\{y_j\}_{j=1}^n$ が複数のボードに対してブロードキャストされるのとは異なり、 g はボードごとに逐次的に転送されるからである. $N \simeq 10^3$ が限界となった理由は、ボードあたりの g の転送量が 13 キロバイトであるのに対して、 $\{b_j\}_{j=1}^n$ および $\{y_j\}_{j=1}^n$ (今、 $n = N$ であることに注意)の転送量が $19N$ バイトであることからおおむね説明できる.

なお、MD2 コードの計算結果は使用ボード数によらないことに注意する.

7. 結 論

本論文では、分子動力学シミュレーションを加速するために開発された専用ハードウェア MDGRAPE-2 を用いて、誘電体による電磁波散乱問題に対するモーメント法を高速化する手法を提案した. モーメント法の定式化は標準のものに従った. それを反復解法によって解くことを前提とした. この場合に計算コスト

上ネックとなる境界積分の評価を，MDGRAPE-2の機能を利用してベクトル並列化するための定式化を示した．その定式化に基づいて実装したモーメント法コードを，1台の汎用PCと3枚のMDGRAPE-2sボードからなる計算機システム上で性能評価した．その結果，本コードは，同じPC上で実行した従来のコードと比べて境界要素数が約200以上の散乱問題において高速であり，境界要素数が約5,000以上では2桁程度高速であることを示した．

最後に今後の課題を述べる．まず，開発したコードを新しい専用計算機上に移植し，より大規模な解析をより速く実行することである．MDGRAPE-2の後継機として整備されつつあるMDGRAPE-3^{31),32)}上では，本コードはほぼそのまま実行可能と考えられる．一方，開発中の超高速計算機GRAPE-DR³³⁾ではプロセッサ内のデータベースをユーザが設定できる．そのため，より効率的なベクトル並列化手法を構成できると考えられる．その定式化は式(20)~(22)から出発すればよいと考えられる．

また，専用計算機の加速手法と高速多重極法(高速アルゴリズム)との併用手法の構成は中長期的な課題である．天体やMDシミュレーションの分野では併用手法による成果がすでに出ている^{22),34),35)}．これらのアナログとして，ポテンシャル問題の境界要素法における併用手法は構成できる¹⁶⁾．他方，より複雑な波動問題では，その高速多重極法と専用計算機の機能を解析的に結び付けることが困難である．そのため，個々の問題の解析的な性質に依存しない併用手法を構成することも今後必要であると考えている¹⁷⁾．いずれの併用手法を構成するとしても，MDGRAPE-2あるいは上記の新しい専用計算機を用いる限り，本論文で構成した専用計算機の加速手法は1つの道具になると考えられる．

謝辞 川井敦(埼玉工大)，薄田竜太郎(九州システム情報技研)，成見哲(理研)の諸氏に感謝する．本研究の一部は，理化学研究所基礎科学特別研究員制度(A03-61012)と文部科学省科学研究費補助金制度若手研究(B¥15760050)の助成による．記して謝意を表す．

参 考 文 献

- 1) 小林昭一(編): 波動解析と境界要素法, 京都大学学術出版会, 京都(2000).
- 2) 熊谷信昭, 森田長吉: 電磁波と境界要素法, 森北出版, 東京(1987).
- 3) Bonnet, M.: *Boundary Integral Equation*

Methods for Solids and Fluids, John Wiley & Sons, West Sussex (1995).

- 4) Brebbia, C., Tanaka, M., Aliabadi, M., Cheng, A. and Kamiya, N.(eds.): *Engineering Analysis with Boundary Elements*, Vol.18 (1996) (Special issue for Parallel BEM and Related Methods and Algorithms).
- 5) 櫻山和男, 西村直志, 牛島 省: 並列計算入門, 計算力学レクチャーシリーズ 3, 丸善, pp.93-109 (2003).
- 6) Cunha, M., Telles, J., Countinho, A. and J.Panetta: On the Parallelization of Boundary Element Codes using Standard and Portable Libraries, *Engineering Analysis with Boundary Elements*, Vol.28, pp.893-902 (2004).
- 7) Narumi, T.: *Special-Purpose Computer for Molecular Dynamics Simulations*, Ph.D. Thesis, University of Tokyo, Tokyo (1997).
- 8) Susukita, R., Ebisuzaki, T., Elmegreen, B., Furusawa, H., Kato, K., Kawai, A., Kobayashi, Y., Koishi, T., McNiven, G., Narumi, T. and Yasuoka, K.: Hardware Accelerator for Molecular Dynamics: MDGRAPE-2, *Computer Physics Communications*, Vol.155, pp.115-131 (2003).
- 9) Takahashi, T. and Ebisuzaki, T.: Accelerating Boundary Integral Equation Method using a Special-Purpose Computer, *International Journal for Numerical Methods in Engineering*, Vol.66, pp.529-548 (2006).
- 10) 高橋 徹, 戎崎俊一: 専用計算機による3次元静弾性問題に対する境界積分方程式法の高速度化, 日本機械学会論文集 A 編, Vol.71, No.712, pp.1620-1625 (2005).
- 11) 高橋 徹, 戎崎俊一: 専用計算機 MDGRAPE-2 による2次元スカラー波動方程式に対する境界積分方程式法の高速度計算, 第17回計算力学講演会講演論文集, Vol.17, pp.291-292, 仙台, 日本機械学会(2004).
- 12) Rokhlin, V.: Rapid Solution of Integral Equations of Classical Potential Theory, *Journal of Computational Physics*, Vol.60, pp.187-207 (1985).
- 13) Greengard, L. and Rokhlin, V.: A Fast Algorithm for Particle Simulations, *Journal of Computational Physics*, Vol.73, pp.325-348 (1987).
- 14) Nishimura, N.: Fast Multipole Accelerated Boundary Integral Equation Methods, *Appl. Mech. Rev.*, Vol.55, pp.299-324 (2002).
- 15) Chew, W., Song, J., Cui, T., Velamparambil, S., Hastriter, M. and Hu, B.: Review of Large Scale Computing in Electromagnetics with Fast Integral Equation Solvers, *Computer*

- Modelling in Engineering and Sciences*, Vol.5, No.4, pp.361–372 (2004).
- 16) 高橋 徹, 戎崎俊一: 専用ハードウェアと高速アルゴリズムによる境界積分方程式法の加速, 計算工学会講演会論文集, Vol.10, 東京, 日本計算工学会, pp.583–586 (2005).
- 17) 高橋 徹, 戎崎俊一: 専用計算機による高速多重極境界積分方程式法の加速, 第 18 回計算力学講演会講演論文集, Vol.18, 筑波, 日本機械学会, pp.539–540 (2005).
- 18) Rao, S., Wilton, D. and Glisson, A.: Electromagnetic Scattering by Surfaces of Arbitrary Shape, *IEEE Trans. Antennas and Propagation*, Vol.30, pp. 409–418 (1982).
- 19) Medgyesi-Mitschang, L., Puntnam, J. and Gedera, M.: Generalized Method of Moments for Three-Dimensional Penetrable Scatterers, *J. Opt. Soc. Am. A*, Vol.11, No.4, pp.1383–1398 (1994).
- 20) Abramowitz, M. and Stegun, I.(eds.): *Handbook of Mathematical Functions: with Formulas, Graphs, and Mathematical Tables*, Dover, New York, eighth dover printing edition (1972).
- 21) Narumi, T., Kawai, A. and Koishi, T.: An 8.61 T op/s Molecular Dynamics Simulation for NaCl with a Special-Purpose Computer: MDM, *Proc. Supercomputing 2001*, ACM (2001).
- 22) Kawai, A., Makino, J. and Ebisuzaki, T.: Performance Analysis of High-Accuracy Tree Code based on Pseudoparticle Multipole Method, *Astrophysical Journal Supplement*, Vol.151, pp.13–33 (2004).
- 23) Yatsuyanagi, Y., Ebisuzaki, T., Kiwamoto, Y., Hatori, T. and Kato, T.: Simulations of Diocotron Instability using a Special-Purpose Computer, MDGRAPE-2, *Phys. Plasmas*, Vol.10, pp.3188–3195 (2003).
- 24) Susukita, R.: M2library.
<http://atlas.riken.jp/~susukita>
- 25) Saad, Y. and Schultz, M.: GMRES; A Generalized Minimal Residual Algorithm for Solving Nonsymmetric Linear Systems, *SIAM J. Scientific and Statistical Comput.*, Vol.7, pp.856–869 (1986).
- 26) Barrett, R., Eijkhout, V., Romine, C. and Vorstt, H.: *Templates for the Solution of Linear Systems: Building blocks for iterative methods*, SIAM, Philadelphia (1994).
- 27) Frayssé, V., Giraud, L. and Gratton, S.: A set of GMRES Routines for Real and Complex Arithmetics, CERFACS Technical Report TR/PA/97/49, CERFACS, Toulouse (1997).
- 28) Narumi, T.: gentable.
<http://atlas.riken.jp/~narumi>
- 29) Intel Compiler 9.0 for Linux.
<http://www.intel.com/cd/software/products/asm-na/eng/219758.htm>
- 30) MDGRAPE-2s ボード. <http://www.riken.jp/r-world/info/release/pamphlet/annual/2002/pdf02/0954.pdf>
- 31) MDGRAPE-3. <http://mdgrape.gsc.riken.jp/index.php>
- 32) Taiji, M., Naurmi, T., Ohno, Y., Futatsugi, N., Suenaga, A., Takada, N. and Konagaya, A.: Protein Explorer: A Petaflops Special-Purpose Computer System for Molecular Dynamics Simulations, *Proc. Supercomputing 2003*, ACM (2003).
- 33) GRAPE-DR project.
<http://grape-dr.adm.s.u-tokyo.ac.jp>
- 34) Makino, J.: Yet Another Fast Multipole Method without Multipoles — Pseudoparticle Multipole Method, *Journal of Computational Physics*, Vol.151, pp.910–920 (1999).
- 35) Amisaki, T., Toyoda, S., Miyagawa, H. and Kitamura, K.: Development of Hardware Accelerator for Molecular Dynamics Simulations: A Computational Board that Calculates Non-bonded Interactions in Cooperations with Fast Multipole Method, *Journal of Computational Chemistry*, Vol.24, pp.582–592 (2003).

(平成 17 年 10 月 3 日受付)

(平成 18 年 1 月 23 日採録)



高橋 徹

1996 年東京工業大学工学部機械物理工学科卒業。2002 年京都大学大学院工学研究科博士課程修了。博士(工学)。現在、理化学研究所基礎科学特別研究員。計算力学、特に積分方程式法の研究に従事。日本機械学会、日本計算数理工学会、日本物理学会各会員。



戎崎 俊一 (正会員)

1981 年大阪大学理学部物理学科卒業。1986 年東京大学理学系大学院博士課程修了。博士(理学)。以後、1989 年東京大学教養学部助教等を経て、現在、理化学研究所戒崎計算宇宙物理学研究室主任研究員。専用計算機の開発とその計算科学への応用、観測衛星 EUSO の開発等に従事。日本天文学会、日本物理学会等各会員。