OpenMP を用いた FDTD 法の高速計算に関する検討

愛媛大学大学院　学生員　○徳永淳一，　愛媛大学大学院　正会員　中畑和之
東京工業大学大学院　正会員　木本和志，　東京工業大学大学院　正会員　廣瀬壮一

1. はじめに

FDTD 法は波動方程式を時間領域と空間領域で差分化し，その差分式を時間ステップごとに逐次計算することで対象領域内の波動場を数値的に求める方法[2)]である．時間域の解法であるため，波動伝搬シミュレーションに適した手法である．しかしながら，対象領域全体をメッシュに分割する必要があり，高周波数域や大領域を扱う場合には，メッシュの増加に伴って計算時間が増加する．そこで本研究では，OpenMP[3)]を FDTD 法に組み込むことによって，計算時間を短縮することを試みる．OpenMP とは，スレッド並列計算を行うための支援ツールであり，共有メモリ型計算機上で機能する．本研究では，SH 波が伝搬する 2 次元面外波動問題を考え，OpenMP による並列計算を実行することによって，大規模な波動伝搬シミュレーションを行う．ここでは，京都大学学術情報メディアセンターのスパコン HPC2500[1)] を使用し，スレッド数を変化させた場合の計算効率について述べる．

2. 面外波動問題の離散化と OpenMP の導入

x_1–x_2 平面において，x_3 軸方向に偏向しながら伝搬する SH 波を考える．このとき，x_3 軸方向の粒子速度を v_3，せん断応力を τ_{31} と τ_{32} とすれば，SH 波は次の波動方程式を満足する．

$$\rho\dot{v}_3(\boldsymbol{x},t) = \frac{\partial\tau_{31}(\boldsymbol{x},t)}{\partial x_1} + \frac{\partial\tau_{32}(\boldsymbol{x},t)}{\partial x_2} \tag{1}$$

また，構成式は次の 2 つの式で表される．

$$\dot{\tau}_{3\alpha}(\boldsymbol{x},t) = \mu\frac{\partial v_3}{\partial x_\alpha}, \quad (\alpha = 1,2) \tag{2}$$

ここで，ρ は密度，μ はせん断弾性係数である．上式の $\dot{(\ )}$ は時間微分 $\frac{d()}{dt}$ を表す．

式 (1)〜(2) を空間領域および時間領域に対して，中心差分で近似し，k+1 番目の時間ステップの v_3，および k+$\frac{1}{2}$ 番目の時間ステップの τ_{31},τ_{32} について整理すると，

$$(v_3)^{k+1}_{i+\frac{1}{2},j+\frac{1}{2}} = \Delta t\frac{(\tau_{31})^{k+\frac{1}{2}}_{i+1,j+\frac{1}{2}} - (\tau_{31})^{k+\frac{1}{2}}_{i,j+\frac{1}{2}}}{\rho h_1} + \Delta t\frac{(\tau_{32})^{k+\frac{1}{2}}_{i+\frac{1}{2},j+1} - (\tau_{32})^{k+\frac{1}{2}}_{i+\frac{1}{2},j}}{\rho h_2} + (v_3)^{k}_{i+\frac{1}{2},j+\frac{1}{2}} \tag{3}$$

$$(\tau_{31})^{k+\frac{1}{2}}_{i,j+\frac{1}{2}} = (\tau_{31})^{k-\frac{1}{2}}_{i,j+\frac{1}{2}} + \frac{\mu\Delta t}{h_1}\{(v_3)^{k}_{i+\frac{1}{2},j+\frac{1}{2}} - (v_3)^{k}_{i-\frac{1}{2},j+\frac{1}{2}}\} \tag{4}$$

$$(\tau_{32})^{k+\frac{1}{2}}_{i+\frac{1}{2},j} = (\tau_{32})^{k-\frac{1}{2}}_{i+\frac{1}{2},j} + \frac{\mu\Delta t}{h_2}\{(v_3)^{k}_{i+\frac{1}{2},j+\frac{1}{2}} - (v_3)^{k}_{i+\frac{1}{2},j-\frac{1}{2}}\} \tag{5}$$

となる．ここで，h_1 と h_2 は，それぞれ x_1 と x_2 方向におけるメッシュ(格子) の間隔を示し，下付き文字 i と j はそれぞれ x_1 と x_2 方向における基準位置からの節点番号を示している．また，Δt は時間間隔であり，解析を安定して実行するには，以下の Courant 条件を満足する必要がある．

$$\Delta t \leq \frac{1}{c_T\sqrt{(\frac{1}{h_1})^2 + (\frac{1}{h_2})^2}} \tag{6}$$

ここで，c_T は対象材料の横波音速である．式 (3)〜(5) より，ある整数次の時間ステップ k で求まった v_3 を用いて半整数次の時間ステップ $k+\frac{1}{2}$ における τ_{31} と τ_{32} が求まることになり，この作業を交互に実行することで順次に解が求まる．

ある時間ステップ k において，メッシュの数に応じた i と j に関する逐次計算を実行する必要がある．例えば Fortran では，式 (3)〜(5) の i と j の演算は DO ループ文で記述できる．OpenMP のループ構文[3)] を用いれば，これらの逐次計算は並列に処理できる．

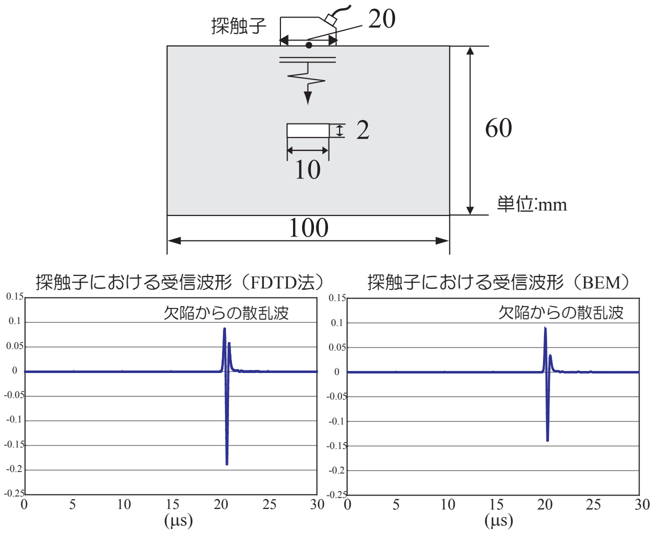
3. FDTD 法の精度および高速化の検証

HPC2500 は 1 ノードあたり 128 個の CPU を有している．従って，ノード内で計算を行う場合に，最高で 128 スレッドの並列処理が可能である．

(1) FDTD 法の精度検証

OpenMP により高速化した FDTD 法の精度を検証するために，同じ数値モデルを BEM で解析した結果と FDTD 法で解析した結果を比較する．BEM による解析では，Niwa らの論文[4)] を参考にしている．ここでは，図–1 に示すようなステンレス鋼 (c_T=3200m/s, ρ=7850kg/m^3) の中央に幅 10mm，高さ 2mm の矩形空洞が存在するモデルに対して，波動伝搬解析を行っ

キーワード：FDTD 法，並列計算，OpenMP，波動伝搬，シミュレーション
〒 790-8577　松山市文京町 3，　TEL: 089-927-9812, FAX: 089-927-9840，E-mail: nakahata@dpc.ehime-u.ac.jp



図– 1 探触子における受信波形

た．ステンレス鋼の上部の探触子から，中心周波数 2MHz のパルス波を発生させている．

探触子中央部の点において観測される波形を図–1 に示す．FDTD 法と BEM で得られた散乱波形は概ね一致しており，高速化した FDTD 法は精度良く計算できることが示された．

(2) 並列計算の性能評価

次に，並列計算の性能を検討するために，図–2 のような幅 Lmm，高さ Lmm の被検体 (c_T=3000m/s, ρ=7800kg/m^3) の中央に幅 0.2mm，高さ 0.2mm の矩形空洞欠陥が存在する被検体に対して，波動伝搬シミュレーションを行った．ここでは，メッシュの大きさを一定 (h_1=h_2=0.01mm) とし，L の大きさを変化させた場合の並列計算の効率について検討する．設定した L の大きさと，それに要する大凡のメッシュ数を表–1 に示す．

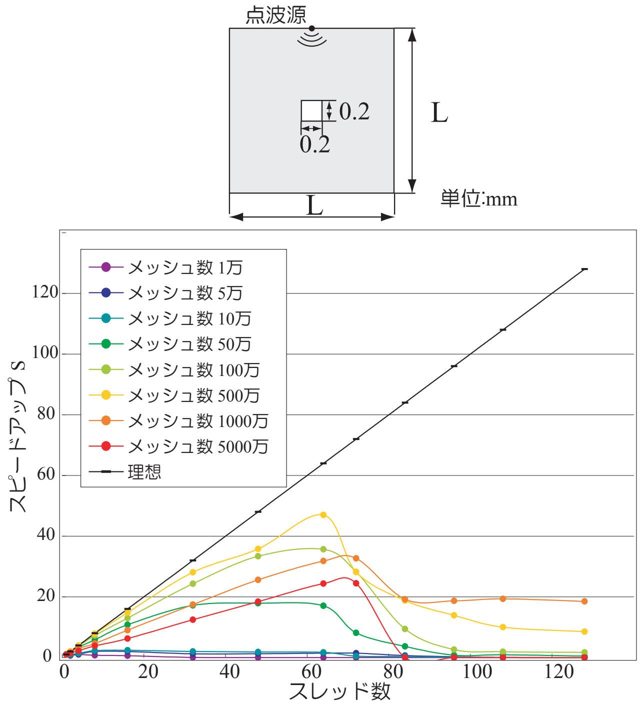
ここでは，スレッド数を変化させた場合の逐次計算と並列計算に要する時間を計測し，スピードアップ S を求める．計算効率を表す S は，次式で定義される．

$$S = \frac{\text{逐次計算に要する計算時間}}{\text{並列計算に要する計算時間}} \tag{7}$$

メッシュ数ごとに S をプロットしたものを図–2 に示す．図–2 において，理想値は S とスレッド数が一致する点であり，このとき並列計算の効率が最も高くなる．図–2 によれば，スレッド数の増加に従い S が大

表– 1 メッシュ数と被検体サイズの関係

メッシュ数	L (単位 mm)
1万	1.0
5万	2.23
10万	3.16
50万	7.07
100万	10.0
500万	22.36
1000万	31.62
5000万	70.71



図– 2 メッシュ数を変化させた場合のスピードアップ

きくなっており，逐次計算と比較して計算時間が大幅に短縮されていることが分かる．しかし，HPC2500 において並列計算を行う場合，スレッド数がおよそ 64 を超えると S が低下している．これはスレッド作成時のメモリ振り分け，およびスレッド終了時のメモリ同期にタイムロスを生じているのが原因の 1 つと考えられる．また，メッシュ数が少ない（およそ 10 万要素以下）場合，スレッド数が増加しても S に変化が見られない．メッシュ数が少ない場合には，並列計算による時間短縮よりも，スレッド作成・終了時のタイムロスの方が大きく，並列化の効果がほとんど表れないといえる．

4. 結論

本研究では，コンピュータ演算時にスレッド並列方式の計算を行う OpenMP を FDTD 法のプログラムに組み込むことによって，弾性波動問題を高速に解析することを試みた．シミュレーション結果から，並列計算の導入により FDTD 法の計算時間が大幅に短縮できることが示された．今後の課題として，2 次元面内問題や 3 次元問題への拡張，さらにはコンクリート等の非均質材料中の波動伝搬問題にも拡張したいと考えている．

参考文献

1) https://web.kudpc.kyoto-u.ac.jp/
2) 橋本修: 実践 FDTD 時間領域差分法, 森北出版, 2006.
3) 牛島省: OpenMP による並列プログラミングと数値計算法, 丸善, 2006.
4) Niwa, Y., Kitahara, M. and Ikeda, H.: The BIE approach to transient wave propagation problems around elastic inclusions. *Theor. appl. Mech.*, Vol.32, pp.183-198, 1984.