

# 有限要素法可視化対話シミュレータの試作

The Visual/Interactive Numerical Simulation System Using Finite Element Method

竹下明良<sup>1)</sup>, 塩沢徳人<sup>2)</sup>, 梅谷征雄<sup>3)</sup>

Akira TAKESHITA and Yoshihito SHIOZAWA and Yukio UMETANI

1)静岡大学 情報学研究科情報学専攻 (〒432-8011 浜松市城北3-5-1, cs6055@cs.inf.shizuoka.ac.jp)

2)現在 日本タイムシェア株式会社

3)工博 静岡大学教授 情報学部情報科学科 (〒432-8011 浜松市城北3-5-1, umetani@cs.inf.shizuoka.ac.jp)

Quick response and interactivity is crucial for the simulator of the physical phenomena to facilitate the rapid construction and the parameter calibration of the target model. We developed a visual/interactive simulator using the FEM(Finite Element Method). To keep the interactivity great efforts are need to balance the computation and visualization time, because the computation cost for FEM is greater than that for FDM(Finite difference Method).

We developed several methods to reduce the computation cost of iterative and direct linear solvers to improve the interactivity of the simulator

**Key Words** : simulator , interactive , FEM , liner solver

## 1 はじめに

物理シミュレーションにおいてそのモデルの忠実度を上げるためにはパラメータの感度解析や実測値との対比が重要な役割を占める。このような操作を容易にするためには、応答性と対話性に優れたシミュレーション環境が必要と考える。[1]

本研究は物理モデルとして平板の板振動を取り上げ、そのモデルの空間形状、分割刻み数、パラメータ値などの変更を即座に計算して表示することにより、そのモデルの調整を効率よく行うための物理シミュレータを試作し、試作したシミュレータの応答性についての評価を行う。

また既存の差分法シミュレータに対し境界条件の自由度を増すために有限要素法による解法機能を付加した。有限要素法においては剛性マトリックスを使った線形計算を解く必要があるため計算負荷が大きい。したがって差分法シミュレータと比べ可視化と計算のバランスに注意を払う必要がある。本研究は線形方程式の解法に反復法と直説法を取り入れ、計算効率の向上を行い有限要素法可視化対話シミュレータの応答性を確保し実時間比例可視化処理の検討を行う。試作のためのプログラミング言語としてVisual C++を用いた。

## 2 物理モデル

### (1) 板の曲げ振動の物理モデル

2次元の板の曲げ振動の方程式は次のようになる[2].

$$D \left( \frac{\partial^4 w}{\partial x^4} + 2 \frac{\partial^4 w}{\partial x^2 \partial y^2} + \frac{\partial^4 w}{\partial y^4} \right) + m \frac{\partial^2 w}{\partial t^2} - q_z(x, y, z) = 0$$

D: 板の曲げ剛性, w: 板の変位,  $\mu$ : 板の単位面積あたりの質量

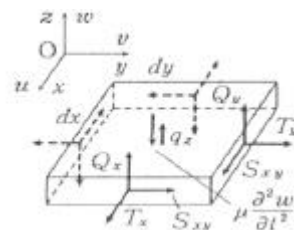


図 1 板の曲げ振動モデル

## 3 シミュレーション環境

### (1) シミュレーション画面

有限要素法可視化対話シミュレータのシミュレーション画面を図に示す。

### (2) 機能

表 1 にシミュレーション機能を示す。

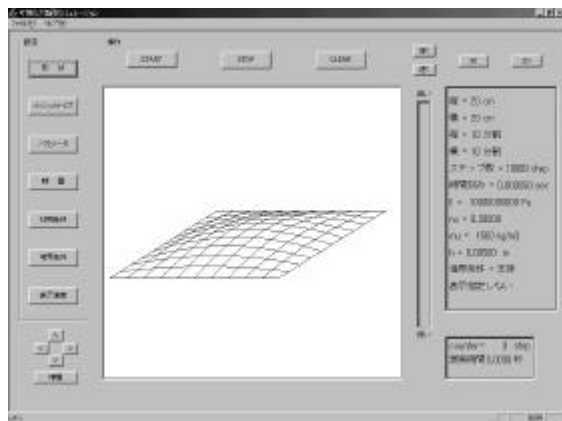


図 2 シミュレーション画面

表 1 シミュレーション機能

|          |               |
|----------|---------------|
| 形状ボタン    | 板の縦と横の長さを設定   |
| メッシュボタン  | 板の縦と横の分割数を設定  |
| パラメータボタン | 各種パラメータの設定    |
| 時間ボタン    | 時間刻みとステップ数を設定 |
| 初期条件ボタン  | 初期条件を指定       |
| 表示速度ボタン  | 表示速度を指定       |
| Startボタン | シミュレーションを始める  |
| Stopボタン  | シミュレーションを止める  |
| Clearボタン | 画面をクリア        |
| 速くボタン    | 表示速度を速くする     |
| 遅くボタン    | 表示速度を遅くする     |
| 増幅ボタン    | 振幅の増幅率を設定     |
| 表示画面1    | 板の振動の様子を表示    |
| 表示画面2    | 設定値を表示        |
| カウンター画面  | 現在のステップ数を表示   |

#### 4 応答性の評価

物理シミュレーションにおいて、応答性は重要なパラメータのひとつである。その応答性をより良いものにするために、シミュレーションに必要な時間の解析を行う。そして有限差分法シミュレータと比較し有限要素法シミュレータの応答性を評価する。なお線形計算は直接法のガウス消去法で行った。

##### (1) 実験方法

一定の条件の下でメッシュサイズを変化させたときの

- ・計算のみにかかる時間
- ・表示のみにかかる時間
- ・全体にかかる時間

をそれぞれ独立に測定する。メッシュサイズは縦、横とも同じとする。

今回の実験では一定条件を

|       |                             |
|-------|-----------------------------|
| 板の形状  | 縦 100cm 横 100cm             |
| ヤング率  | $10 \times 10^9 \text{ Pa}$ |
| ポアソン比 | 0.3                         |

|         |                                |
|---------|--------------------------------|
| 密度      | $1500 \text{ kg/m}^3$          |
| 板の厚さ    | 5mm                            |
| 時間刻み幅   | $5 \times 10^{-5} \text{ sec}$ |
| 時間ステップ数 | 1000step                       |
| 境界条件    | 四辺単純支持                         |

とした[3]。

##### (2) 実験結果

有限差分法の測定結果を表2に、有限要素法の測定結果を表3に示す。この結果の時間は1時間ステップにかかる時間を表している。

表 2 有限差分法の計算・表示・全体時間(msec)

| メッシュ | 計算時間 | 表示時間  | 全体時間  |
|------|------|-------|-------|
| 10   | 0.09 | 8.27  | 8.23  |
| 20   | 0.16 | 12.83 | 13.00 |
| 30   | 0.36 | 19.90 | 20.33 |
| 40   | 0.61 | 30.50 | 30.97 |

表 3 有限要素法の計算・表示・全体時間(msec)

| メッシュ | 計算時間      | 表示時間  | 全体時間      |
|------|-----------|-------|-----------|
| 10   | 152.80    | 3.20  | 157.60    |
| 20   | 9335.10   | 13.50 | 9354.40   |
| 30   | 64274.00  | 30.10 | 64304.10  |
| 40   | 355866.50 | 43.20 | 355909.70 |

##### (3) 評価

結果より有限差分法と比較した場合、有限要素法は計算時間がかかっていることが確認できる。つまり線形計算がシミュレーション性能を大きく低下させている。有限要素法シミュレータで十分なメッシュサイズでの応答性を確保するには計算時間を短縮し表示と計算時間のバランスを改善する必要がある。

#### 5 応答性の改善

有限要素法は剛性行列を使った線形計算を解くため、計算負荷が有限差分法と比較して大きいことが確認された。したがって有限差分法シミュレータ以上に可視化と計算のバランスに注意を払う必要がある。反復法と直接法の両者について計算効率の向上に努め、表示と計算にかかる時間のバランスを改善し有限要素法シミュレータの応答性を確保した。

##### (1) 直接法の改善

###### 1) 帯行列を用いる

係数行列の外側にある零成分を取り除いたガウス消去法

###### 2) LU分解

クラウトのLU分解、ガウスのLU分解、コレスキーのLU分解をあらかじめ行って、その後前進、後退代入操作の

み行う。

(2) 反復法の改善

1) 出発値の改善

反復計算の出発値に前ステップの値を代入する

2) 不要計算の排除

非零成分のみを計算対象とする

3) 出発値の改善と不要計算の排除

4) マトリックス生成時間の短縮

線形計算で使用するマトリックス生成の際非零成分のみを対象とする

5) コンパイラの最適化

(3) 結果

以上の改善を行ったときの測定結果を表4に示す。

表 4 メッシュ数20の計算・表示・全体時間

|                  | 計算時間    | 表示時間  | 全体時間    |
|------------------|---------|-------|---------|
| 差分法              | 0.36    | 19.90 | 20.33   |
| 直接法              | 9335.10 | 13.50 | 9354.40 |
| 帯行列              | 111.90  | 11.34 | 123.24  |
| クラウド             | 132.70  | 11.30 | 144.01  |
| ガウス              | 155.36  | 10.72 | 166.08  |
| コレスキー            | 217.50  | 12.68 | 230.18  |
| 反復法              | 837.70  | 9.80  | 847.50  |
| 出発値改善            | 297.00  | 13.50 | 306.80  |
| 不要計算排除           | 27.10   | 11.20 | 38.40   |
| 出発値改善+<br>不要計算排除 | 11.50   | 10.40 | 22.00   |
| 最適化              | 4.70    | 10.70 | 15.20   |

(4) 考察

直接法で帯行列を用いた場合は計算量が減り、LU分解を用いた場合は演算回数が減るので計算時間が短縮された。しかし一番の計算時間短縮ができた帯行列を扱った方法でも、20×20 分割の時1秒間に7～8時間ステップしか進めることができない。また、板の大きさが同じであっても、分割数が増えると計算時間が大きくなってしまふ。これは、実時間可視化という点から見ると、この物理シミュレータの板振動を自然現象と捉えるためには、まだまだ充分でないとと言える。

これに対し反復法のガウスザイデル法を取り入れたことで直接法に比べ計算時間が短縮された。反復法は直接法に比べ1ループでの計算量は減少するが収束までループを繰り返す必要がある。しかし直接法より計算時間が短縮されたということは、収束までループを有限回行っても全体の計算量は反復法の方が少なかったことになり、今回の実験では反復法の方が直接法より良い結果が得られた。次に出発値の改善は、1ステップの平均ループ回数を約50%減少させ計算時間も短縮された。さらに、時間ステップ毎のループ回数のばらつきがなくなり処理時間

が安定する効果もあった。最後に不要計算の排除はマトリックスの特性上、線形計算およびマトリックス生成時間の短縮において大きな効果が得られた。

ガウスザイデル法で出発値の改善と不要計算を排除しコンパイラの最適化まで行った場合、計算時間が短縮され表示と計算にかかる時間がほぼ等しくなった。今回メッシュ分割数40までの計算時間と表示時間のバランスの改善が行われた。

6 実時間比例処理

実時間比例処理ではメッシュ分割数に影響せずに、ユーザが指定した1ステップ時間で処理を行う必要がある。つまり、実時間比例処理を行うには1ステップの時間を長くしたり短くしたりする必要がある。ユーザに1ステップの時間を指定させ、指定時間が実際の処理時間より長い場合は待ち時間を設け時間調整を行う。逆に指定時間が実際の処理時間より短い場合は、表示回数を間引くことで処理時間の短縮を行う。

1ステップ時間の指定を行ったときの待ち時間を表5上段に表示間引き率を下段に示す。間引き率とは表示間隔をあらわす。間引き率0ときはすべての表示が行われる。逆に間引き率100とは1度も表示が行われなかったことになる。値が入っていない部分は、指定された1ステップ時間が計算時間より短く、指定の時間で処理を行うことができない部分である。

表 5 メッシュ20の1ステップ指定時間と間引きの関係  
(上段:待ち時間 下段:間引き率)

| 1ステップ指定時間 (msec) |     |     |     |     |      |      |      |      |      |
|------------------|-----|-----|-----|-----|------|------|------|------|------|
| 1                | 2   | 3   | 4   | 5   | 6    | 7    | 8    | 9    | 10   |
|                  |     |     |     | 0.0 | 0.1  | 0.2  | 0.6  | 0.8  | 1.8  |
|                  |     |     |     | 97  | 89   | 80   | 75   | 67   | 67   |
| 1ステップ指定時間(msec)  |     |     |     |     |      |      |      |      |      |
| 11               | 12  | 13  | 14  | 15  | 16   | 17   | 18   | 19   | 20   |
| 2.8              | 2.0 | 3.0 | 4.0 | 5.0 | 0.6  | 1.6  | 2.6  | 3.6  | 4.6  |
| 67               | 50  | 50  | 50  | 50  | 0    | 0    | 0    | 0    | 0    |
| 1ステップ指定時間(msec)  |     |     |     |     |      |      |      |      |      |
| 21               | 22  | 23  | 24  | 25  | 26   | 27   | 28   | 29   | 30   |
| 5.6              | 6.6 | 7.6 | 8.6 | 9.6 | 10.6 | 11.6 | 12.6 | 13.6 | 14.6 |
| 0                | 0   | 0   | 0   | 0   | 0    | 0    | 0    | 0    | 0    |

(1) 評価基準

実時間比例処理の応答性が確保されているかを見るために2つの上限を設けた。

1) 間引き率の上限

間引き率の上限とは各ステップ数で応答性を確保できる間引き率の最高値。間引き率の上限を超えた場合は、表示間の時間が長くなりすぎ振動の様子を表示する画面

の画質が悪くなる。つまり応答性を確保した実時間比例処理が行えない。今回は表示間隔が25msecになるものを間引き率の上限とした。メッシュ分割数20では間引き率の上限は図3より約80になる。

表 6 間引き率と表示間隔の関係 (メッシュ20)

|      |     |      |      |      |      |
|------|-----|------|------|------|------|
| 間引き率 | 0   | 10   | 20   | 30   | 40   |
| 表示間隔 | 4.7 | 5.2  | 5.8  | 6.7  | 7.8  |
| 間引き率 | 50  | 60   | 70   | 80   | 90   |
| 表示間隔 | 9.4 | 11.6 | 15.7 | 23.5 | 47.0 |

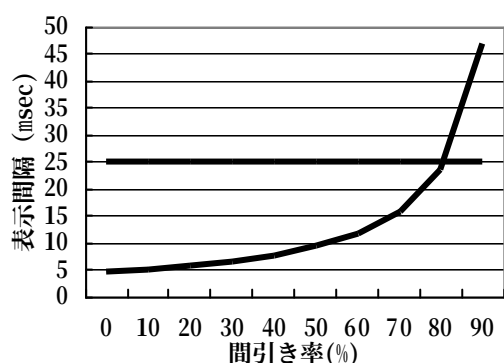


図 3 間引き率と表示間隔の関係 (メッシュ20)

## 2) 1ステップ指定時間の上限

1ステップ指定時間の上限とは応答性を確保できる1ステップ指定時間の最高値。1ステップ指定時間が長すぎると、シミュレーション速度そのものが遅くなり応答性を確保した実時間比例処理が行えない。今回は1ステップ指定時間が25msecになるものを1ステップ指定時間の上限とした。

### (2) 評価結果

実時間比例処理の応答性が確保できる1ステップ指定時間は表5中の1ステップ指定時間が5から25の範囲になる。しかし、間引き率の上限は80なので、1ステップ指定時間が5から6の範囲は、指定時間での処理は可能であるが応答性が確保されない。つまりメッシュ分割数20では1ステップ指定が7から25の範囲で実時間比例処理が可能である。この状況を図4に示す。同様にメッシュ分割数10では指定時間が1~25、メッシュ分割数30では23~25、メッシュ分割数40では実時間比例処理が可能な指定時間が存在しなかった。

### (3) 結論

1ステップ時間の指定が可能になり、メッシュ分割数による処理時間のばらつきは是正された。しかし、間引き率の上限を超えた場合は、振動を表示する画面の画質が悪くなってしまいスムーズなシミュレーションが行われない。つまり表示を間引いて処理時間を短くし実時間比例処理を行う方法は、メッシュ分割数が大きくなったと

きに応答性が確保されない。また表示回数を減らす方法では計算時間より小さい1ステップ時間を指定され場合は対処できない。つまり表示を間引く方法は、メッシュ分割数が大きくなる程に実時間比例処理が可能な1ステップ指定時間の範囲が狭くなる。

計算時間が小さい有限差分法において表示を間引く方法は効果的であったが、有限要素法ではやはり計算時間が問題になった。大きなメッシュ分割数でもスムーズなシミュレーションを行うためには計算時間をさらに短縮するか表示時間を短縮し間引き率を低くする必要がある。

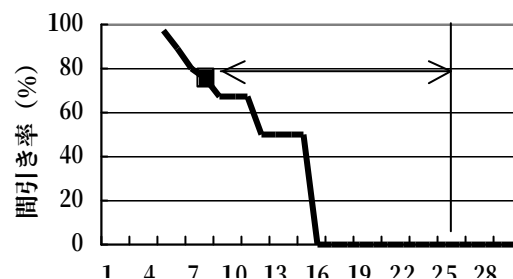


図 4 間引き率と1ステップ指定時間 (メッシュ20)

## 7 まとめ

本研究では有限要素法可視化対話シミュレータの計算時間を短縮し応答性の確保を行った。計算アルゴリズムの改善によって、計算と表示時間のバランスを改善することができた。また実時間比例処理を行うために、表示アルゴリズムを改善しメッシュ分割数の違いによる1ステップの処理時間のばらつきを是正した。しかし、表示アルゴリズムを改善するだけでは、メッシュ分割数が大きくなったときはスムーズなシミュレーションを行うことはできなかった。つまり、有限要素法シミュレータで実時間比例処理を行うには、計算時間がまだ問題になっている。

### 謝辞：

本研究を行うに当たり、直接懇切丁寧なる御指導、御教授頂いた、静岡大学情報学部情報科学科 梅谷征雄教授に深く感謝致します。また、日頃親切に御協力頂いた梅谷研究室の皆様に厚く御礼を申し上げます

### 参考文献

- 1)梅谷征雄：PSE可視化インターフェース，第一回問題解決ワークショップ論文集，pp25-30，1998
- 2)小林繁夫：振動学，丸善株式会社，1994
- 3)柳沢猛，野田直剛，他共著：パソコン対応基礎材料力学，日新出版株式会社，1995
- 4)菊地文雄：有限要素法概説，株式会社 サイエンス社，1999