

可視化対話型 3 次元ピアノ響板モデルの開発

Development of 3 Dimensional Visual/Interactive Piano Soundboard Simulator

能崎友博¹⁾、梅谷征雄²⁾

Tomohiro NOZAKI¹⁾, Yukio UMETANI²⁾

1) 静岡大学大学院情報学研究科情報学専攻 (〒432-8011 浜松市城北 3-5-1, cs9068@cs.inf.shizuoka.ac.jp)

2) 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)

A component oriented visual / interactive simulator is a PSE which can perform the simulation of physical phenomena based on the connection graph of physical components. We implemented a new 3 Dimensional piano soundboard component using FEM (Finite Element Method) by converting an existing soundboard simulator to a ATL COM (Component Object Model) component. It becomes possible to perform a more detailed simulation by creating a 3 Dimensional model. The process of conversion is described here.

Key Words: Piano soundboard, simulator, FEM, ATL COM, 3 Dimensional

1. はじめに

現在、コンピュータによる物理現象・工学現象のシミュレーションは、実験・理論に次ぐ第 3 の手法として活用されている。しかし、新規にシミュレータを作成する場合には、シミュレーションに必要な作業は非常に煩雑で、作業負荷が大きい。そこで、それぞれの物理現象をそれぞれ 1 つのコンポーネントとして用意し、それらのコンポーネントを組み合わせることによってシミュレータを作成するシミュレーション環境が提案されている。この方法により、シミュレータ作成過程における作業負荷の軽減が模索されている。

本研究の目的は、コンポーネントを使った既存のシミュレータ「コンポーネント指向可視化対話型シミュレータ」(図 1)用の COM コンポーネント「3 次元ピアノ響板コンポーネント」を Microsoft Visual C++搭載の ATL を使って開発し、響板振動の可視化を実現しながらリアルタイムにシミュレーションを行うことである。

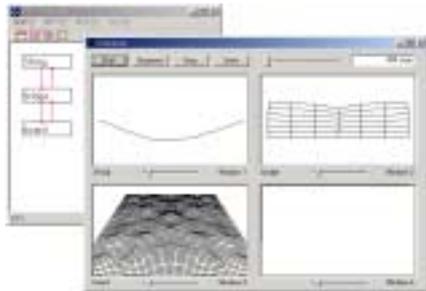


図 1 コンポーネント指向可視化対話型シミュレータ

2. ANSYS とその問題点

市販ソフトウェア ANSYS は、高精度で多機能な汎用有限要素法解析ソフトウェアである。しかし、ピアノの物理モデルのシミュレーションを行う場合、可視化対話を実現しつつ ANSYS を使うことは出来ない。その理由を以下に述べる。

ANSYS は、自身が生成したデータファイルをやり取りすることでしか他のプログラムと連動させることが出来ない仕様になっている。また、実行中の ANSYS データベース形式は非公開であるためアクセスすることが出来ない。

ANSYS の解析(30000 節点響板)では 1 ステップの実行に約 1 分かかり、可視化対話を実現する速度とは到底いえない。精度が高く、安定したソルバを使っているためか解析速度が極めて遅い。

また、1 ステップごとに膨大な量のシミュレーション結果をデータファイルに出力するため、ユーザはシミュレーションが終わるまでパラメータを変更したり、結果を表示したりすることが出来ない。

以上の理由から対話性を持たせるにあたって ANSYS を使うことは出来ないと考える。

3. 2 次元モデルから 3 次元モデルへ

以前の研究でコンポーネント指向可視化対話型シミュレータに有限要素法による 2 次元ピアノ響板コンポーネントを実装した。本研究で開発する 3 次元モデルはその

拡張版である。しかし、一言に拡張といっても3次元モデルは2次元モデルと比べて相違点が多くある。

第1に、3次元化によって計算式が複雑化したことが挙げられる。計算式が複雑化したことで、現状の計算部分のコードを流用することが出来なくなった。よって新規にコーディングする必要がある。

第2に、節点数の増加により計算量が増大したことが挙げられる。プログラムで一番計算の負荷が高い部分は、後述する連立一次方程式のソルバのコード部分である。

そこではn行n列(nは節点数)の2次元配列とn行のベクトルが引数として使われる。よって、nの値が増加するほどメモリ使用量と計算量が増えることになる。

第3に、3方向(X軸,Y軸,Z軸)のシミュレーション結果を表示する際に従来の表示方法では表示が困難になったことが挙げられる。従来の2次元解析結果の表示方法ではX方向、Y方向の変位を表示することが出来ない。

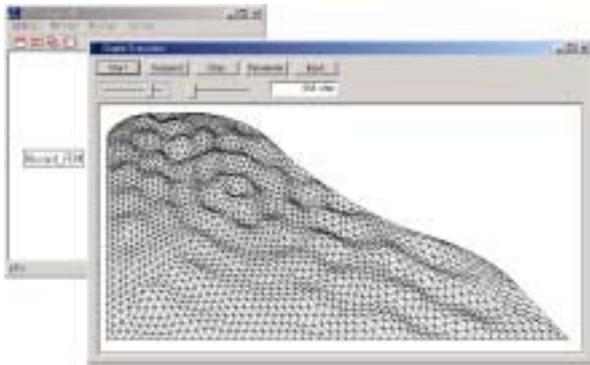


図2 2次元響板シミュレーション画面

4. 数理モデルと離散化

3次元響板の振動方程式は3次元弾性体の微分方程式から導出した。つりあい条件式、ひずみと変位の関係式、弾性法則(ひずみと応力の関係式)を展開してX,Y,Z方向それぞれについて組み立てる。(1)

$$\left\{ \begin{aligned} \rho \frac{\partial^2 u}{\partial t^2} + R \frac{\partial u}{\partial t} - \bar{X} &= \frac{\partial}{\partial x} \left\{ \frac{E_x}{h} \left(1 - \nu_{xz}^2 \frac{E_y}{E_z} \right) \frac{\partial u}{\partial x} + \frac{E_x}{h} \left(\nu_{yz} + \nu_{xz} \nu_{xy} \frac{E_z}{E_y} \right) \frac{\partial v}{\partial y} + \frac{E_x}{h} \left(\nu_{xz} + \nu_{xy} \nu_{yz} \frac{E_z}{E_y} \right) \frac{\partial w}{\partial z} \right\} \\ &\quad + \frac{\partial}{\partial z} \left\{ G_{xz} \left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x} \right) \right\} + \frac{\partial}{\partial y} \left\{ G_{xy} \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \right\} \\ \rho \frac{\partial^2 v}{\partial t^2} + R \frac{\partial v}{\partial t} - \bar{Y} &= \frac{\partial}{\partial y} \left\{ \frac{E_y}{h} \left(\nu_{yz} + \nu_{xz} \nu_{xy} \frac{E_z}{E_x} \right) \frac{\partial u}{\partial x} + \frac{E_y}{h} \left(1 - \nu_{yz}^2 \frac{E_x}{E_z} \right) \frac{\partial v}{\partial y} + \frac{E_y}{h} \left(\nu_{yz} + \nu_{xz} \nu_{xy} \frac{E_x}{E_z} \right) \frac{\partial w}{\partial z} \right\} \\ &\quad + \frac{\partial}{\partial z} \left\{ G_{yz} \left(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} \right) \right\} + \frac{\partial}{\partial x} \left\{ G_{xy} \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \right\} \\ \rho \frac{\partial^2 w}{\partial t^2} + R \frac{\partial w}{\partial t} - \bar{Z} &= \frac{\partial}{\partial z} \left\{ \frac{E_z}{h} \left(\nu_{xz} + \nu_{xy} \nu_{yz} \frac{E_x}{E_y} \right) \frac{\partial u}{\partial x} + \frac{E_z}{h} \left(\nu_{yz} + \nu_{xz} \nu_{xy} \frac{E_x}{E_y} \right) \frac{\partial v}{\partial y} + \frac{E_z}{h} \left(1 - \nu_{yz}^2 \frac{E_x}{E_y} \right) \frac{\partial w}{\partial z} \right\} \\ &\quad + \frac{\partial}{\partial y} \left\{ G_{yz} \left(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} \right) \right\} + \frac{\partial}{\partial x} \left\{ G_{xz} \left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x} \right) \right\} \end{aligned} \right. \quad (1)$$

この方程式をGreenの公式を使って変形した後に、離散化を行う。響板の振動が偏微分方程式で記述される現象であること、響板の形状が複雑であらかじめ部材とし

て自然なものが存在しないこと、シミュレーションにはコンピュータを使う、という点を考慮した結果、離散化手法として有限要素法を用いることにした。

導いたX方向の方程式に対して、Galerkin式有限要素法による離散化を行う。(2)

$$\left(\begin{array}{c} \left(\frac{\rho}{(\Delta t)^2} + \frac{R}{2\Delta t} \right) \int_{\Omega} L_j L_i dx \\ \vdots \\ \left(\frac{2\rho}{(\Delta t)^2} \int_{\Omega} L_j L_i dx \right) \\ \vdots \\ \int_{\Omega} L_j L_i dx \end{array} \right) \begin{pmatrix} u_1^{t+1} \\ u_2^{t+1} \\ u_3^{t+1} \\ \vdots \\ u_n^{t+1} \end{pmatrix} = \left(\begin{array}{c} \left(\frac{R}{2\Delta t} - \frac{\rho}{(\Delta t)^2} \right) \int_{\Omega} L_j L_i dx \\ \vdots \\ \left(\frac{2\rho}{(\Delta t)^2} \int_{\Omega} L_j L_i dx \right) \\ \vdots \\ \int_{\Omega} L_j L_i dx \end{array} \right) \begin{pmatrix} u_1^{t-1} \\ u_2^{t-1} \\ u_3^{t-1} \\ \vdots \\ u_n^{t-1} \end{pmatrix} + \left(\begin{array}{c} \int_{\Omega} L_j L_i dx \\ \vdots \\ \int_{\Omega} L_j L_i dx \\ \vdots \\ \int_{\Omega} L_j L_i dx \end{array} \right) \begin{pmatrix} f_{i_1}^t \\ f_{i_2}^t \\ f_{i_3}^t \\ \vdots \\ f_{i_n}^t \end{pmatrix} + \left(\begin{array}{c} \int_{\Omega} g_{i_1} L_j d\gamma \\ \vdots \\ \int_{\Omega} g_{i_n} L_j d\gamma \end{array} \right) \begin{pmatrix} \int_{\Omega} g_{i_1} L_j d\gamma \\ \vdots \\ \int_{\Omega} g_{i_n} L_j d\gamma \end{pmatrix} - \left(\begin{array}{c} \int_{\Omega} A_{i_1} grad(L_j) grad(L_j) dx \\ \vdots \\ \int_{\Omega} A_{i_n} grad(L_j) grad(L_j) dx \end{array} \right) \begin{pmatrix} u_1^t \\ u_2^t \\ u_3^t \\ \vdots \\ u_n^t \end{pmatrix} - \left(\begin{array}{c} \int_{\Omega} A_{i_1} grad(L_j) grad(L_j) dx \\ \vdots \\ \int_{\Omega} A_{i_n} grad(L_j) grad(L_j) dx \end{array} \right) \begin{pmatrix} w_1^t \\ w_2^t \\ w_3^t \\ \vdots \\ w_n^t \end{pmatrix} \quad (2)$$

5. 設計

前述した問題点や計算式を踏まえて開発すべきシミュレータの仕様を考える。リアルタイムシミュレーションを実現するために、ANSYSのシミュレーション精度と描画精度には若干劣るが、高精度で高画質表示のシミュレーションを行えるようにする。また、シミュレーション結果をファイル出力しないことでHDDにアクセスする時間を削る。更に、COMコンポーネント化を行うことで、ANSYSでは不可能であった弦シミュレータとの連結を行えるようにする。

響板のモデル作成と要素分割はコンポーネントで行わず、ANSYSで作成してそれをコンポーネントで読み込む形をとる。

連立一次方程式を解くにあたって計算機を用いることから線形反復法を適用するのがよい。今回は線形反復法で最もよく使われるGauss-Seidel法を用いた。Gauss-Seidel法では式(3)を利用して反復式を構成する。

$$u_i = \left(f_i - \sum_{\substack{j=1 \\ j \neq i}}^m a_{ij} u_j \right) / a_{ii} \quad (3)$$

u_j として近似値が得られたならば、それを式(3)の右辺に代入すれば、 u_i の新しい近似値が得られ、それは初め

の近似値よりも改善されたものであろうと期待するのである。

連立一次方程式で使用する左辺の全体係数行列は(節点数)²の大きさを持つ DOUBLE 型の 2 次元配列である。予定とする響板モデルの最大節点数は約 30000 個である。しかし、(30000)²個の DOUBLE 型のメモリ容量は約 6.7GB となり、それだけの容量を物理メモリ上に確保するのは現実的ではない。そこで係数行列の中身に注目すると、対称で、零成分が多く含まれる疎な行列であった。このことから、係数行列の非零成分のインデックスと値とを保存し、零成分を削除して配列に格納することにする。格納した配列からインデックスを参照して値を読み込む仕様にする事で、メモリ使用量を約 1/10 に削減できる。また、行列の要素数が減ったことで、結果的に要素の検索にかかる時間を短縮させることができる。

3次元響板シミュレーションの実行結果表示は、変位の表示方向を 1 方向のみに限定し、表示方向を選択して実行する仕様にする。結果はカラーで表示し、青が最小、赤が最大、緑が中間の変位値をさすようにする。これは ANSYS の結果表示方法を参考にしている。

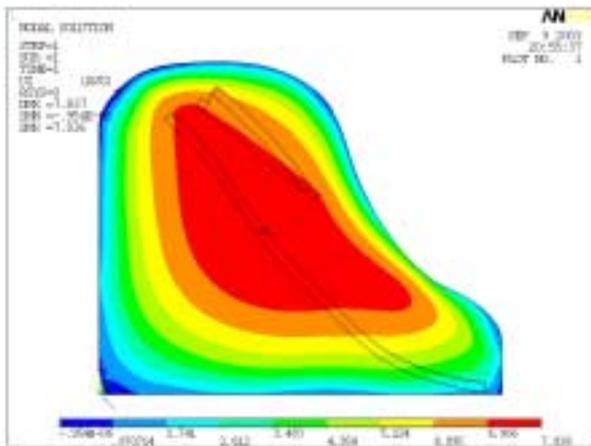


図3 ANSYS の結果表示画面

6. 開発

6.1 モデル生成

3次元響板モデルを四面体の有限要素によって分割しモデル化する。形状データを作成するために ANSYS を使ってモデル形状を生成・分割し、要素番号・節点番号対応リスト、節点番号・節点座標値対応リスト、境界上節点番号リストの 3 つに分けてそれぞれをファイルに書き出す。それらをコンポーネント内でユーザに指定させて読み込む関数を実装した。

6.2 パラメータ値変更

響板にユーザが与えなければならないパラメータには、響板材料特性値、弦コンポーネントと響板が接する座標、

表示する変位値の方向がある。弦コンポーネントと接する座標は鍵盤のキー番号によって指定することにした。

表1 ユーザが与えるパラメータ

パラメータ名	変数名
ヤング率	Ex, Ey, Ez
メジャーポアソン比	xy, yz, zx
せん断弾性係数	Gxy, Gyz, Gzx
体積密度	
減衰率	R
入出力キー番号	Key_Num
表示方向	View_Direct

6.3 係数行列生成

2次元係数行列の生成を行う。導いた計算式に従って、それぞれの要素マトリックスを計算する。すると、1要素につき 4×4 の行列が得られるので、その値を全体行列のしかるべき位置に割り当てる。この計算を全要素の個数の分だけ繰り返して全体係数行列を作成する。次に、作成した係数行列に境界条件を与える。本モデルでは響板の側面部を固定する。つまり要素 a_{ii} の値を 1 に、 a_{ij} の値を 0 にする。

メモリ使用量削減のために、作成した係数行列から非零要素を検索し、インデックスと値を保存して零要素を削除する。

6.4 連立一次方程式ソルバ

Gauss-Seidel 法による連立一次方程式ソルバをコーディングする。係数行列の非零要素を持つインデックスと値とが配列内に保存されている。行列内の値を読み込むには、読み込もうとする要素のインデックスを配列から検索し、検索したインデックスが指す場所にある値を取り出す。収束判定条件に使用する許容誤差境界の値は 10^{-15} に設定した。

6.5 結果表示

最後に、解析結果表示部を作成する。X, Y, Z 方向それぞれから表示できるようにするために探索を行い、それぞれの視点から表面に見える節点をリスト化する。変位値は最小値から最大値までを 21 個に分割し、青色を最小値、緑色を中間値、赤色を最大値に割り当てて描画する。

7. 実形状響板シミュレーション

7.1 実行結果

実形状 3次元響板モデル(節点数 4997、要素数 17334) の C4 弦の位置に響板上面からサイン波(1000Hz)を入れて、作成した響板コンポーネント単体でのシミュレーション

ョンを行い、Z方向の変位値を響板の上から見た図を表示した。(図4)

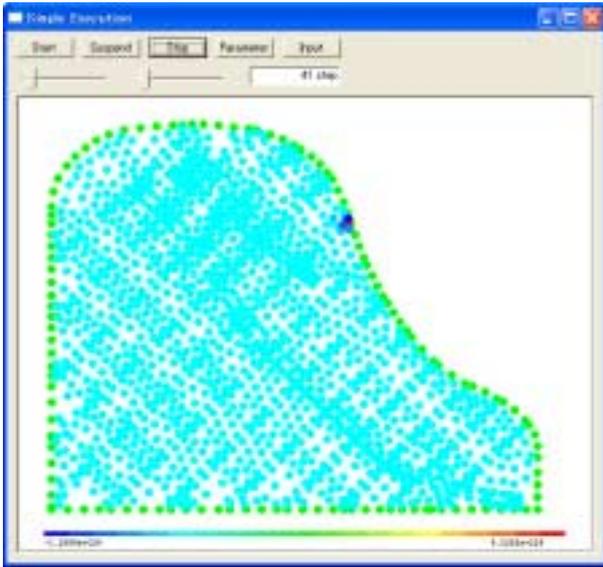


図4 響板上面部表示 (発散)

現段階では節点の座標値が指す位置に色をつけた丸を表示させているだけである。そのため ANSYS の結果表示画面(図3)に比べると画像が粗くなっている。

シミュレーション結果は、全体の節点の変位値が一部分を除いて最小値と最大値を繰り返す発散状態になってしまった。参考までに、このときの実行速度は約 0.5step/sec であった。

7.2 考察

発散が起こった以上、シミュレーションが正しく行われておらず、プログラムが計算式のどこかに問題があると考えられる。

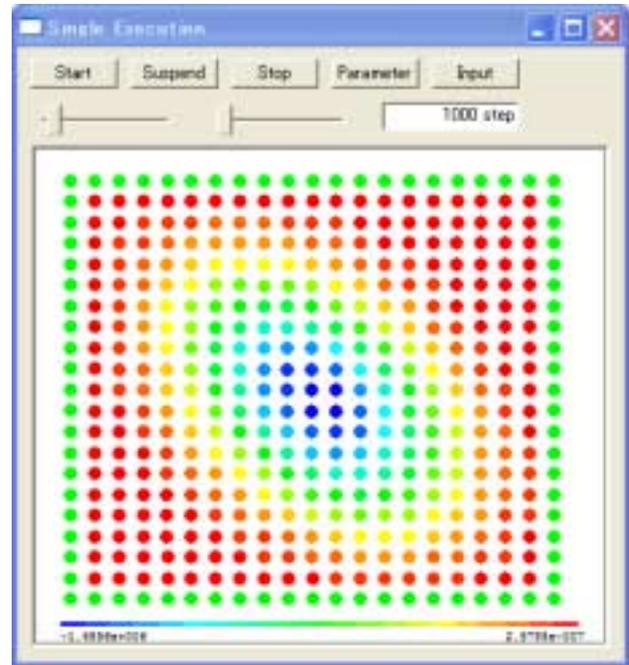
Gauss-Seidel 法の検証を他の連立一次方程式に適用した結果、正しく動作していることが確認できた。よって計算負荷がこれより大幅に高くなることはないと考えられるため、実行速度については信頼できる値であると判断する。

プログラムを修正するために検証を行う必要があるが、シミュレーションしたモデルが大きすぎて検証がしづらい。簡素な形状で、比較的節点数の少ないモデルを生成する必要がある。また、検証にあたって比較するプログラムが必要である。ANSYS はブラックボックスなプログラムであるので、ここでは偏微分方程式数値シミュレーションツール PSILAB を利用してプログラムの問題点を探ることにする。

8 正方形響板シミュレーション

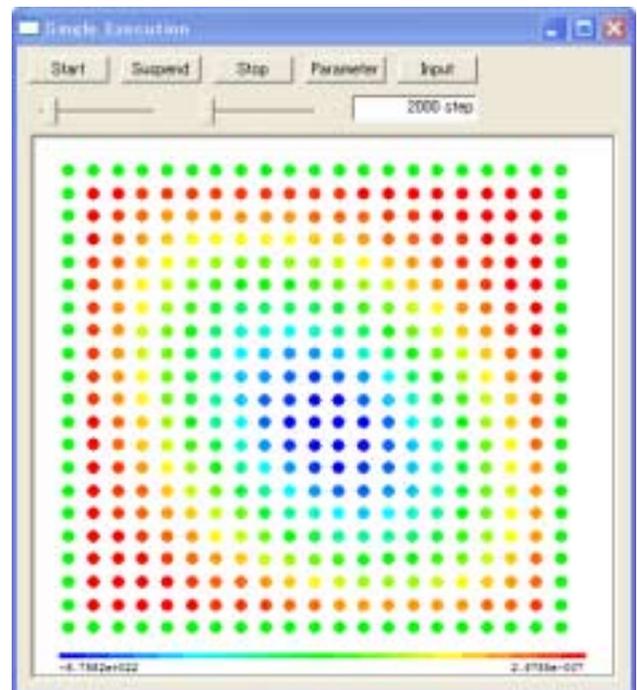
8.1 実行結果

テストモデルとして2次元平板モデル(節点数 882、要素数 2400)に板中央部上面からサイン波(1000Hz)を入れて、響板コンポーネントと PSILAB のそれぞれでシミュレーションを行った。



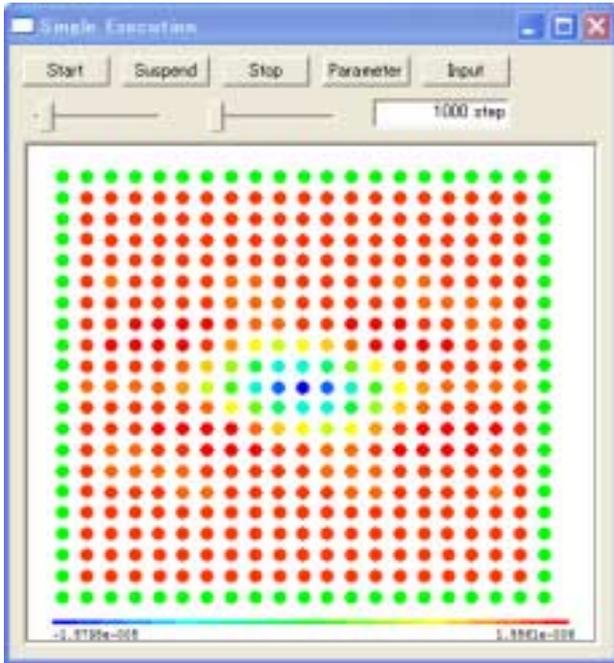
変位値 $-1.4596e+6 \sim 2.9755e-7$ (mm)

図5 3次元響板コンポーネント実行結果(1000step)

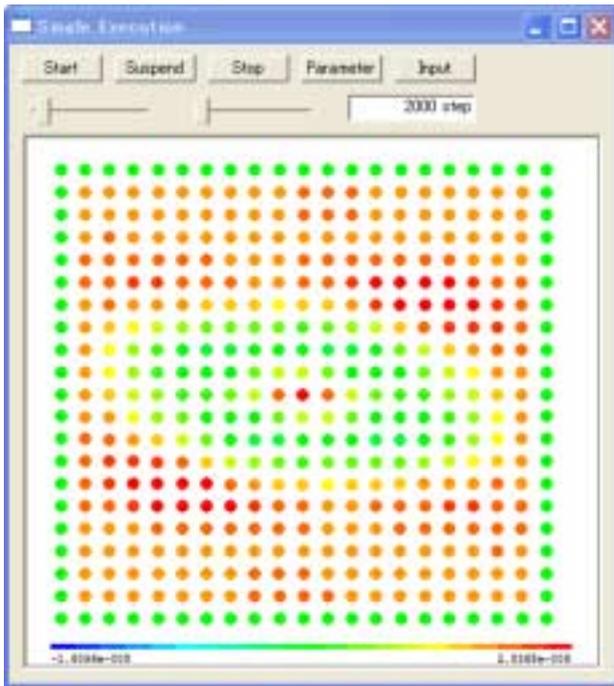


変位値 $-6.7582e+22 \sim 2.9755e-7$ (mm)

図6 3次元響板コンポーネント実行結果(2000step)



変位値 $-1.5798e-5 \sim 1.5561e-6$ (mm)
 図7 3次元響板 PSILAB 実行結果(1000step)



変位値 $-1.6096e-5 \sim 2.8368e-6$ (mm)
 図8 3次元響板 PSILAB 実行結果(2000step)

図5と図7を比較すると、波の伝わり方にかなり違いがあることがわかる。図5は変位値がかなり大きく、板中央部を中心に大きく窪んでいる。一方、図7は同じく中央が窪んでいるものの、変位値は図5ほど大きくなく、板全体に小さな波が出来ているのが特徴である。

図5から1000step後の図7では窪みが更に大きくなり、最小変位値も更に大きくなっているが、最大変位値に変化はない。それに比べて図8では逆に中央部が盛り上がり、波が板全体に広がって、響板全体が波打った状態になっている。変位値にはそれほど大きな変化はない。

7.2 考察

図5、6の動きと図7、8の動きの変化を比較すると、作成した響板コンポーネントシミュレータはサイン波を入れているにも関わらず、一向に波打っていないことがわかる。変位値も1000stepで 10^6 、2000stepで 10^{22} と非常に大きくなっている。ここまで値が大きくなってしまうと、サイン波が変位値に影響を与えることが出来ず、板上面にかかる力がないような状態になる。したがって、板が常に窪んでいくという問題があることがわかる。

7.3 検証

まずコンポーネントとPSILABのそれぞれで生成された式(2)左辺の全体係数行列を比較したところ、無視できるほどの微小な差であった。よって、左辺の係数行列は同じであるとみなす。また、左辺の係数行列と右辺第1、2、3項の係数行列は、まず $L_i L_j$ だけで行列を作成し、そこに違う係数をかけただけの項である。よって、右辺のそれらの項についても問題はないとみなす。

次に右辺4、5、6項についてだが、これらは外力に関係する項である。PSILABとの外力項部分に関しても同じであることが確認された。よって、これらの項についても問題はないとみなす。

最後に右辺7、8、9項についてである。これらの項をPSILABの項と比較すると、係数行列のオーダーが $10^1 \sim 10^2$ ほど違っているようであった。よって今後はこの部分の正確な値を導出しなければならない。

8 課題

コンポーネントのシミュレーションで板の中央部が窪み続ける現象の原因が式(2)右辺7、8、9項の係数行列にあると判明した。早急にPSILABの係数行列の生成コードを理解してコンポーネントのプログラムを修正し、正しくシミュレーションが行える環境を整えなければならない。

節点数4997の響板での実行速度が現時点では0.5step/secであった。しかし、将来的に30000節点のモデルを使ってシミュレーションを行いたい。また可視化対話を実現するためには最低10step/secの速度を確保したい。速度を上げるためには、Gauss-Seidel法に加速係数を加えて収束を早めるSOR法を組み込むという方法がある。また、係数行列の対称性を生かして、メモリ上

に保持したインデックスと値を約半分にして検索速度を上げるといった方法もある。将来的には両方とも実装する予定であるが、いずれも大幅にシミュレーション速度を向上させられる可能性は低い。だが、連立一次方程式のソルバ部をスレッド化して、複数の PC に計算負荷を分散する方法は有効であると考えられる。

また、現状の結果表示画面は節点をただ描画しただけであるため、ANSYS のように色が同じ部分を滑らかに繋げて表示を行えるように改善しなければならない。

参考文献

- 1) 吹原教弘「コンポーネント指向可視化対話型シミュレータの研究」平成 12 年度修士論文
- 2) Richard Grimes, Alex Stockton, George Reilly, Julian Templeman 著 田中正造 監訳「ATL COM プログラミング」翔泳社 1999 年

- 3) 竹下明良「可視化対話型シミュレーションワークベンチの研究」平成 13 年度修士論文
- 4) 菊地文雄 著 「有限要素法概説[新訂版] -理工学における基礎と応用-」サイエンス社 1999
- 5) 大地羊三 著 「マトリックス構造解析」コロナ社 1977
- 6) 「PSILAB 偏微分方程式数値シミュレーションツール マニュアル」日立超 LSI エンジニアリング株式会社 1995
- 7) Jeanne C. Adams, Walter S. Brainerd, Jeanne T. Martin, Brian T. Smith, Jerrold L. Wagener 「Fortran 95 Handbook Complete ISO/ANSI Reference」 The MIT Press 1997