

可視化対話型物理シミュレータの試作と応答性の評価

Visual / Interactive Simulator for Physical Phenomena and the Evaluation of Response Time

吹原教弘¹⁾, 梅谷征雄²⁾

Takahiro FUKIHARA and Yukio UMETANI

1)静岡大学 理工学研究科計算機工学専攻 (〒432-8011 浜松市城北3-5-1, s5092@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 visual / interactive simulator for physical phenomena is designed, developed, and evaluated. The purpose of the simulator is to facilitate rapid construction and the parameter calibration of the target model. For that purpose, an object-oriented design with visual interface is adopted within VC++ framework. Based on the measurement of response time, a rule to control the presentation interval in relation to mesh size is determined and tested.

Key Words : simulator for physical phenomena, object-oriented design, visual interface, VC++

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物理シミュレータが必要と考える[1]。

本研究では物理モデルとして平板の曲げ振動を取り上げ、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の影響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評価した。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

2.1 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、μ: 板の単位面積あたりの質量

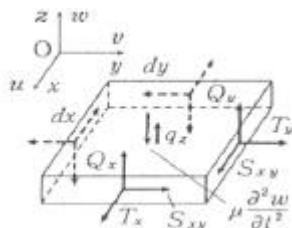


図1 板の曲げ振動のモデル図

2.2 離散化

計算機上への実現のための数値解法として、本研究では有限差分法を用いた。

$$\begin{aligned} \frac{\partial^2 w}{\partial x^2} &= \frac{w(i+1, j, n) - 2w(i, j, n) + w(i-1, j, n))}{\Delta x^2} \\ \frac{\partial^4 w}{\partial x^4} &= \frac{w(i+2, j, n) - 4w(i+1, j, n) + 6w(i, j, n) - 4w(i-1, j, n) + w(i-2, j, n))}{\Delta x^4} \\ \frac{\partial^2 w}{\partial t^2} &= \frac{w(i, j, n+1) - 2w(i, j, n) + w(i, j, n-1))}{\Delta t^2} \end{aligned}$$

有限差分法を用いて振動方程式を離散化しとめると、次の計算式を得る。

$$\begin{aligned} w(i, j, n+1) &= a_0 w(i, j, n) \\ &+ a_1 [w(i, j+1, n) + w(i, j-1, n)] \\ &+ a_2 [w(i, j+2, n) + w(i, j-2, n)] \\ &+ a_3 [w(i+1, j+1, n) + w(i+1, j-1, n) \\ &\quad + w(i-1, j+1, n) + w(i-1, j-1, n)] \\ &+ a_4 [w(i+1, j, n) + w(i-1, j, n)] \\ &+ a_5 [w(i+2, j, n) + w(i-2, j, n)] \\ &- w(i, j, n-1) + \frac{\Delta t^2}{m} q_z(i, j, n) \end{aligned}$$

ここで、係数 $a_0 \sim a_5$ は以下のようなになる。

$$a_0 = \frac{-2D\Delta t^2}{m} \left\{ \frac{3}{\Delta x^4} + \frac{4}{\Delta x^2\Delta y^2} + \frac{3}{\Delta y^4} \right\} + 2$$

$$a_1 = \frac{4D\Delta t^2}{m} \left\{ \frac{1}{\Delta y^4} + \frac{1}{\Delta x^2\Delta y^2} \right\}$$

$$a_2 = \frac{-D\Delta t^2}{m\Delta y^4}$$

$$a_3 = \frac{-2D\Delta t^2}{m\Delta x^2\Delta y^2}$$

$$a_4 = \frac{4D\Delta t^2}{n} \left\{ \frac{1}{\Delta x^4} + \frac{1}{\Delta x^2\Delta y^2} \right\}$$

$$a_5 = \frac{-D\Delta t^2}{m\Delta x^4}$$

3. シミュレータの試作

シミュレータの作成にあたって、以下の手順で進めた。

(1) 機能設定

シミュレータに必要な機能を設定した。

表1 機能設定

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
境界条件ボタン	境界条件を指定
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

(2) 画面構成

機能設定に基づき画面を構成した。図2は作成したシミュレータの画面である。

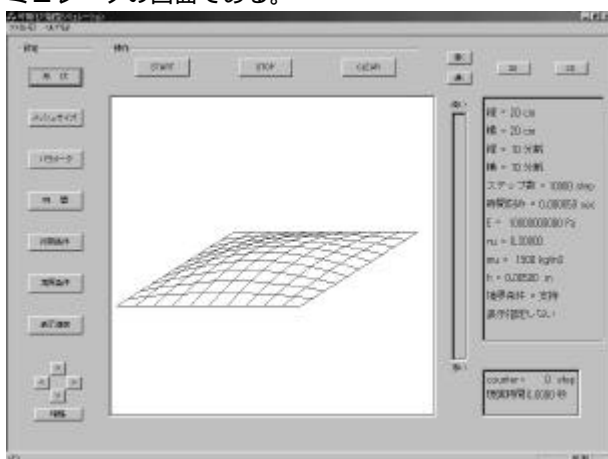


図2 画面構成

(3) クラス作成

設定した機能と画面を実現するためのクラス構成の設計とコード化を行った。図3は本研究で試作したシミュレータのクラス階層図である。

クラス「CDialog」から派生しているクラス「keijou」「mesh」...は、画面上のそれぞれのボタンを押したときに現れるダイアログを定義している。

クラス「CsmpView」は、ユーザがこのシミュレータとやりとりする時に使うウィンドウを実現するためのクラスである。クラス「keijou」「mesh」などで定義したダイアログの呼び出しはこのクラスで行う。クラス「keijou」「mesh」などで設定する各値に対応した変数がこのクラスに用意してあり、各値が変更されればそれに対応した変数に反映されるようになっている。またシミュレータ本体もここに記述してある。

4. 操作方法

ここでこのシミュレータの操作方法を簡単に説明する。

(1) 板の形状、パラメータを設定する。

ここでは板のパラメータを設定する。「形状」ボタンを押すと図3(a)に示されるダイアログが表示されるので、そこに値を入力する。「パラメータ」ボタンについても同様であるが、ヤング率などのパラメータを材質(木、コンクリートなど)で選べるようになっている。

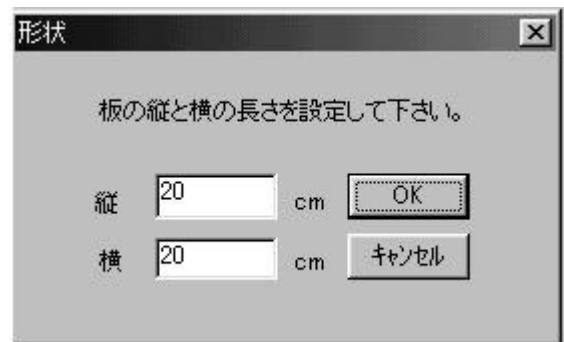


図4(a) 「形状」ダイアログ

(2) 境界条件、初期条件を設定する。

ここではそれぞれの条件を設定する。「境界条件」ボタンを押すと図3(b)に示されるダイアログが表示されるので、そこから希望の条件を選択する。初期条件についても同様である。



図4(b) 「境界条件」ダイアログ

(3) メッシュサイズ、時間刻み幅を設定する。

ここではシミュレーションするための板のメッシュサイズ、時間刻み幅、シミュレート時間ステップ数を設定する。「メッシュサイズ」ボタンを押すと図3(c)に示されるダイアログが表示されるので、そこに値を入力する。時間刻み幅、時間ステップ数についても同様である。

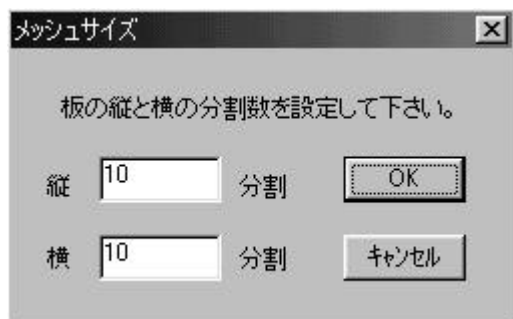


図4(c) 「メッシュサイズ」ダイアログ

(4) シミュレーションを始める。

「start」ボタンを押すとシミュレーションが始まり、板の振動の様子を画面中央の表示ウィンドウに随時表示する。止めたいときは「stop」ボタンを押す。

5. 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一つである。その応答性を向上させるために、ここでシミュレーションに要する時間の解析を行う。

5.1 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 100 cm	横 100 cm
ヤング率	10×10^9 Pa	
ポアソン比	0.3	
密度	1500 kg/m^3	
板の厚さ	5 mm	
時間刻み幅	5×10^{-5} sec	
時間ステップ数	1000 step	
境界条件	四辺単純支持	

とした[3]。

5.2 実験結果

実験結果は図5のようになった。グラフの縦軸は時間 (10^{-3} sec)、横軸はメッシュ(分割)をそれぞれあらわしている。

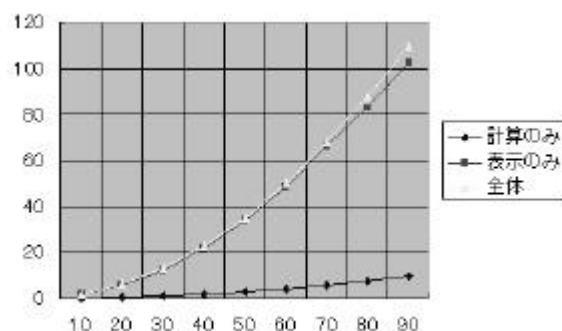


図5 実験結果

5.3 応答性の評価と対策

実験結果より、メッシュサイズを細かくした時、計算時間に比べて表示時間のほうがはるかに時間がかかっていることがわかった。従って応答性向上のための対策として、次のようなものが挙げられる。

- ・表示アルゴリズムを改善し、表示時間を短縮する
- ・表示回数を減らし、全体にかかる時間を短縮する

5.4 応答性の工夫

本研究では5.3節の表示回数を減らす事に関して、メッシュサイズにより表示回数を調整する工夫を行った。できるだけ実時間に比例した表示を行わせるために、表示が遅い場合は適当な間引きを入れ、逆に早過ぎる場合は表示の間に待ちを入れるように試みた。本研究では待ち操作にループを使用した。実測に基づいて定めた間引き数と待ちのループ回数は表2の通りである。表2の上段は間引き数、下段はループ回数 ($\times 10^4$) である。この結果、実時間に比例した表示ができるようになった。

表2 メッシュサイズと間引き数,ループ回数の関係

メッシュ 〔分割〕	指 1 0	定 2 0	ス 3 0	テ 4 0	ッ 5 0	ブ 6 0	数 7 0
1 0	0 610	0 297	0 193	0 141	0 110	0 86	0 72
2 0	0 586	0 270	0 165	0 113	0 79	0 59	0 44
3 0	0 540	0 225	0 120	0 67	0 36	0 16	0 2
4 0	0 477	0 162	0 58	0 5	1 0	2 0	2 0
5 0	0 398	0 88	1 0	2 0	2 0	3 0	3 0
6 0	0 299	1 0	2 0	2 0	3 0	4 0	5 0
7 0	0 200	2 0	2 0	4 0	5 0	7 0	8 0
8 0	0 52	2 0	3 0	5 0	7 0	11 0	15 0
9 0	1 0	3 0	5 0	8 0	12 0	18 0	33 0

6. おわりに

本研究では物理モデルの調整を効率よく行うための可視化対話型物理シミュレータを試作し、その応答性について評価した。またその応答性向上の対策を実装した。今後はシミュレーションにおけるもう一つの重要なパラメータである対話性について形状やメッシュの指定をより直接的に行えるようにし、またメッシュが細かい時におこる画質の劣化を改善するための表示方法の工夫もやりたい。

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

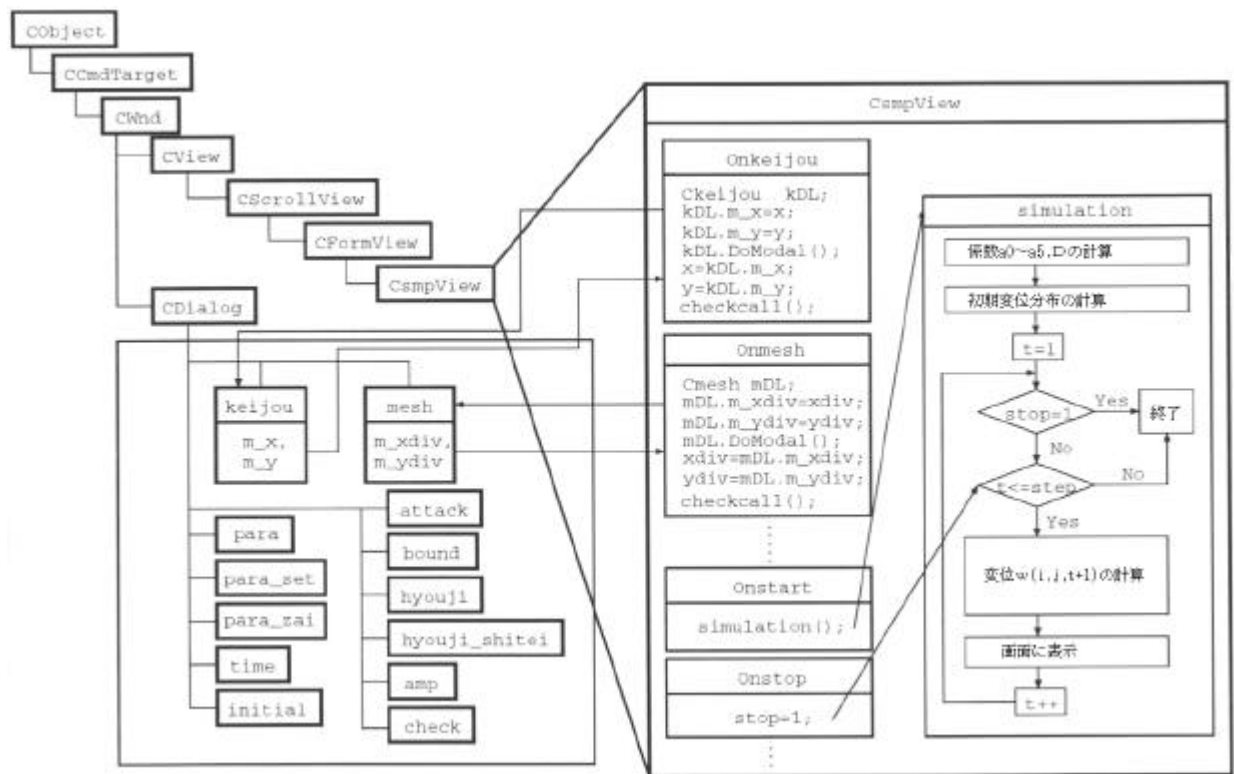


図3 クラス階層図