

JavaApplet を用いた浅水長波流れ解析システムの構築

中央大学	学生員	○ 東條	高宏
中央大学	学生員	岡田	岳
中央大学大学院	学生員	田中	聖三
中央大学	正会員	樫山	和男

1. はじめに

近年、インターネットの普及は著しいものがあり、それによって Web ブラウザ上でアプリケーションを実行可能な Java Applet に注目が集まり、工学問題に対して様々な Applet が構築されている。Applet を用いることの利点として、誰でも利用可能なシステムの構築が可能であること、またビジュアルコンピューティングにより計算結果を容易に把握できる事などが挙げられる。また Java はオブジェクト指向言語であるため、解析システムの構築用の言語としても有効である。

本報告では、Java Applet を用いて、Web ブラウザ上で解析を行うことのできる浅水長波流れ解析システムの構築について述べる。解析例を示すことにより本システムの妥当性と有効性を検討する。支配方程式としては、非線形浅水長波方程式、メッシュ分割には Courant 数が一定となる Delaunay 三角分割法、支配方程式に対する離散化手法としては SUPG(Streamline-Upwind/Petrov Galerkin) 法に基づく陽的安定化有限要素法を用いた。

2. 浅水長波流れ解析 Applet

本システムは以下の 3 つのプロセッシングから成り立っているが、それぞれの役割及び本システム内での働きについて述べていく。

- プリプロセッシング (Pre-Processing)
- メインプロセッシング (Main-Processing)
- ポストプロセッシング (Post-Processing)

図-1 に本システムによる計算の流れを示す。

(1) プリプロセッシング

プリプロセッシングとは、メッシュ生成や境界条件等、メインプロセッシングで必要なデータを作成する部分である。

本システムでは対象となる領域内に任意に設定された点を用いてメッシュ分割を行う手法として、Delaunay 三角分割法を用いる。Delaunay 三角分割法は任意の形状を持った領域を分割するときに有効であり、またメッシュの要素一つ一つが正三角形に近くなる特徴をもっている。これらの特徴は解析を高精度に行う上で有効である。今回用いた手法は波速、代表的要素幅及び微小時間増分量より求められる Courant 数が一定となるように節点発生を行いメッシュ分割を行っている。¹⁾

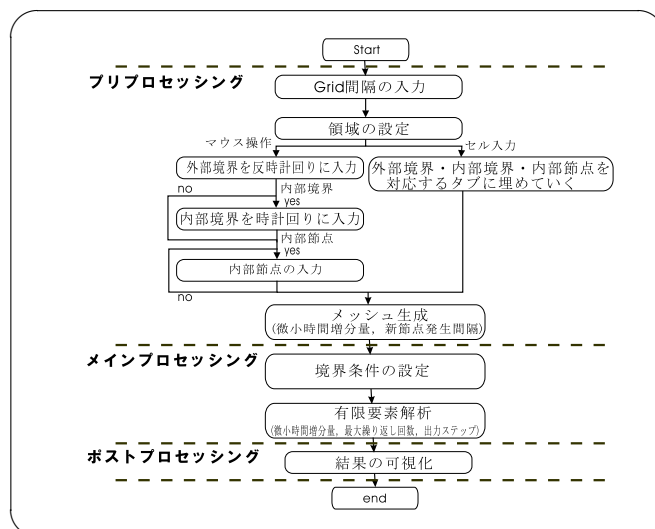


図-1 システムの流れ

図-2 にシステムのメインとなる画面を示す。外部境界 (反時計回り)、内部境界 (時計回り) 及び内部節点は入力補助として表示された格子部分 (10 単位で変更可能) をクリックしていくことによって設定することが可能である。クリックすることにより、同時に水深を入力するダイアログが出現し、水深を入力することにより右にあるテーブルのセルに数値が自動的に入るようになっている。またマウス操作が不要な場合は、右のセルに直接水深を入力してもメッシュの生成が可能である。

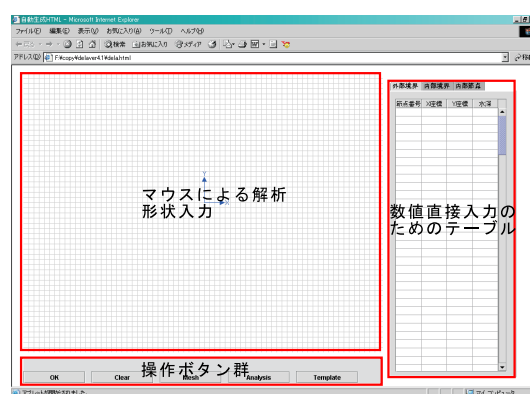


図-2 メイン画面

(2) メインプロセッシング及びポストプリセッシング

メインプロセッシングとは、プリプロセッシングにより入力されたデータをもとに解析を行う部分である。メインプロセッシングで用いる支配方程式には非線形浅水長波方

KeyWords : JavaApplet, 有限要素法, 浅水長波流れ解析, Delaunay 分割, 陽的安定化手法

連絡先 : 〒112-8551 東京都文京区春日 1-13-27 TEL 03-3817-1815 FAX 03-3817-1803

程式系を用いている。

$$\frac{\partial u_i}{\partial t} + u_j \frac{\partial u_i}{\partial x_j} + g \frac{\partial \zeta}{\partial x_i} - \frac{\partial}{\partial x_j} \left[A_h \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \right] = 0 \quad (1)$$

$$\frac{\partial \zeta}{\partial t} + \frac{\partial}{\partial x_i} [(h + \zeta)] = 0 \quad (2)$$

ここに、 u_i は断面平均流速、 ζ は水位変動量、 g は重力加速度、 A_h は水平方向の渦動粘性係数である。支配方程式に対する空間方向の離散化手法として、SUPG 法に基づく安定化有限要素法²⁾を適用する。一方、時間方向の離散化には前進差分を適用する、その結果得られる連立一次方程式の解法には、マルチパス (Multi-Pass) 法を用い、反復回数は 3 回とする。なお、式 (1)、(2) における M_τ は解法に集中行列を用いる陽解法を適用するため考慮しないものとする。

ポストプロセッシングとは、プリプロセッシングで作成されたデータ及びメインプロセッシングの解析結果の可視化を行う部分である。本システムでは、Delaunay 三角分割法により作成されたメッシュ図及びメインプロセッシングによる解析結果を任意の時間間隔で可視化させることが可能である。図-3 に Delaunay 三角分割法によるメッシュを可視化した例を示す。

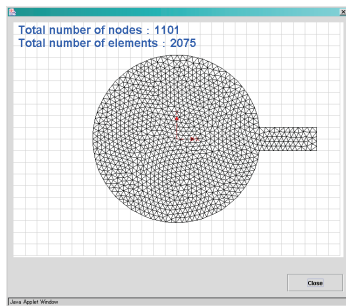


図-3 メッシュの可視化例

3. 解析例

解析例として、フラスコ型の湾内における浅水長波流れ問題を取り上げる。この問題では、システムの Template ボタンを押すことによって節点の座標と水深はすぐに右のテーブルのセルに収められるようになっており、水深は一律に 10.0[m] と仮定した。図-4 に解析領域の形状を示す。メッシュ図は図-3 に示したものである (総節点数:1101, 総要素数:2075)。

計算条件として湾口に振幅 0.5[m]、周期 30[s] の正弦波を与え、微小時間増分量 Δt は 0.05[s]、境界条件は *slip* 条件、初期条件は静水状態 (全節点で流速 $u, v = 0.0$ [m/s]、水位変動量 $\zeta = 0.0$ [m]) と仮定した。解析結果として図-5 に各時刻における水位コンター図及び流速ベクトル図を示す。この図より波の回折等の現象を捉えており本システムの妥当性を示すことが出来る。

次に本システムの計算効率を検討するために、Java 言語 (可視化の有無：計算結果を計算しながら表示するか否

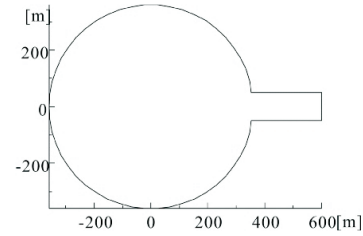


図-4 フラスコ型領域

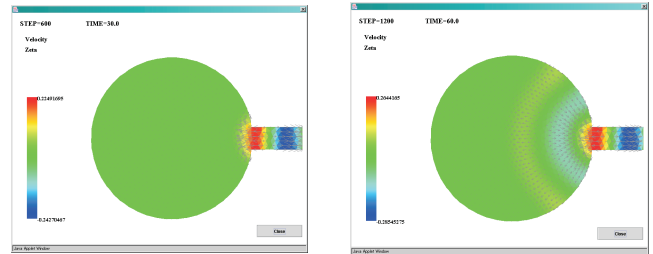


図-5 水位コンター図と流速ベクトル図

か) と、Fortran 言語による計算時間の比較を行った。各言語における要素数の違いによる計算時間の比較を図-6 に示す。なお、計算の全ステップ数を 3000 回とし画面への出力ステップ回数を 200 回毎とした。図-6 の結果より Java による計算時間は Fortran に比べ 5 割程度計算時間がかかることが明らかとなった。また可視化処理を計算と同時にに行わせると更に 5 割程度計算時間がかかることも確認できた。なお、計算機仕様は CPU:PentiumIII 600MHz, Memory:256MHz, OS:Windows2000 である。この結果は Java がインタラクティブ言語であることを考慮すると、計算時間的にも満足な結果といえる。

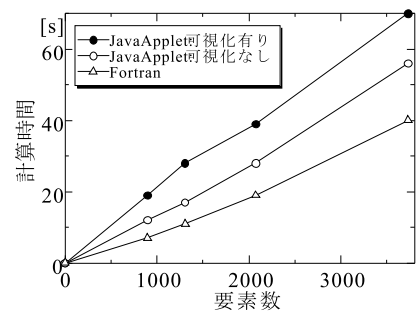


図-6 計算時間の比較

4. おわりに

本報告では JavaApplet を用いて浅水長波流れ解析システムの提案を行った。本手法の有効性を解析例を通じて示した。今後の課題として大規模計算可能なシステムへの移行、汎用性の向上などが挙げられる。

参考文献

- 1) Kazuo Kashiya, Tsuyoshi Okada, 'Automatic mesh generation method for shallow water flow analysis', International Journal for Numerical Methods in Fluids, Vol 15, 1037-1057(1992)
- 2) 日本数値流体学会有限要素法研究委員会編, '有限要素法による流れのシミュレーション', シュプリンガー・フェアラーク東京 (1998)