



| | |
|--------------|---|
| Title | 事例1 : 船体動揺による槽内の流体挙動と内壁に働く変動荷重の可視化 |
| Author(s) | 出口, 弘; 富田, 和雄; 西村, 直章 他 |
| Citation | 大阪大学大型計算機センターニュース. 1995, 95, p. 12-19 |
| Version Type | VoR |
| URL | https://hdl.handle.net/11094/66082 |
| rights | |
| Note | |

The University of Osaka Institutional Knowledge Archive : OUKA

<https://ir.library.osaka-u.ac.jp/>

The University of Osaka

船体動揺による槽内の流体挙動と 内壁に働く変動荷重の可視化

大阪大学大型計算機センター研究開発部

出口 弘

deguchi@center.osaka-u.ac.jp

大阪大学大学院工学研究科船舶海洋工学専攻

富田 和雄、西村 直章、鷺島 由佳

k_tomita@naoe.eng.osaka-u.ac.jp

1 はじめに

液体貨物を積んだ船舶が波浪中を航行する場合、タンク内の流体は船体運動にともなう動揺を受ける。海象と航行中の船の状態によっては内壁には流体から激しい力が働き、液体波面も複雑な形状を呈する。これはスロッシングと呼ばれる非線形な現象であるが、船体運動の小さな場合でもタンク壁の内外に繰り返し働く荷重により損傷へと至ることもある。タンク内の流体挙動は N-S 方程式を数値的解法により求める等の研究も行われているが、ここでは完全流体を仮定し線形理論の範囲内でタンク内の液体波面形状と内壁にかかる圧力分布の様子を調べることを目的としたものである。そのために、以下に示す理論計算をおこなった後、IRIS の Exploler を使って、ある瞬間におけるタンク内の波面形状と 2 つの側面における変動圧力分布を可視化した。そして、時間のパラメーターを変化させることにより、1 周期分 (60 コマ) のイメージファイルを作成し、それらの画像を VideoDesktop を用いてビデオを制御することにより動画像化した。

2 波面形状と変動圧力

船は静止空間座標系の X 軸と角度 χ をなす方向に一定速度で進む座標系の近傍で 6 自由度の微小な定常周期運動をしていると考える。船体固定座標系 $o-xyz$ と、等速移動空間固定座標系 $O_0-X_0Y_0Z_0$ の関係式は次式で表される。

$$\begin{aligned} X_0 &= \xi + x - y\psi + x\theta \\ Y_0 &= \eta + y + x\psi - z\varphi \\ Z_0 &= \zeta + z - x\theta + y\varphi \end{aligned} \tag{2.1}$$

座標系を次ページの図 1 に示し、運動は以下の表示を用いる。

ξ : 前後揺れ ζ : 上下揺れ θ : 縦揺れ
 η : 左右揺れ角 ψ : 船首揺れ角 φ : 横揺れ角

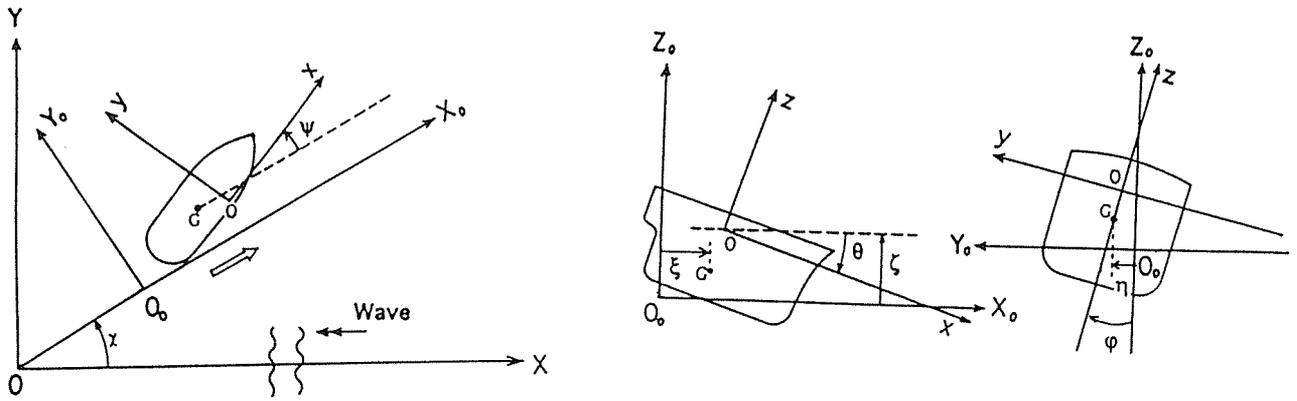


図1 座標系

タンク形状はここでは図2に示す矩形のものを考える。

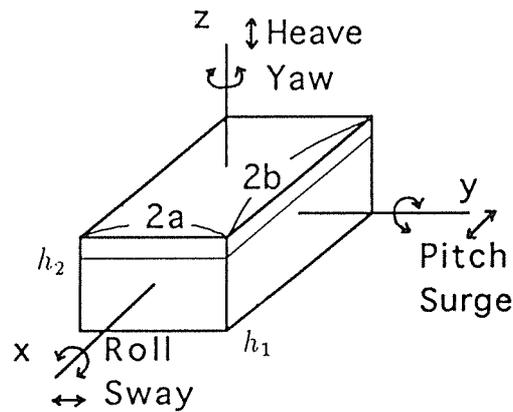


図2 タンクと運動の表示

速度ポテンシャル ϕ は支配方程式であるラプラスの式及び以下の境界条件を満足しなければならない。

$$\nabla^2 \phi = 0 \quad (2.2)$$

$$\frac{\partial^2 \phi}{\partial t^2} + g \frac{\partial \phi}{\partial z} = 0 \quad \text{at } z = h_2 \quad (2.3)$$

$$\frac{\partial \phi}{\partial z} = \dot{\zeta} - x\dot{\theta} + y\dot{\varphi} \quad \text{at } z = h_1 \quad (2.4)$$

$$\frac{\partial \phi}{\partial x} = \dot{\xi} - y\dot{\psi} + z\dot{\theta} \quad \text{at } x = b \quad (2.5)$$

$$\frac{\partial \phi}{\partial y} = \dot{\eta} + x\dot{\psi} - z\dot{\varphi} \quad \text{at } y = a \quad (2.6)$$

以上を満たすポテンシャルとして次式を考える。

$$\begin{aligned}
\phi = & \dot{\varphi} \left[-yz + \sum_{n=0}^{\infty} A_{\varphi(2n+1)} \sin\left\{\frac{(2n+1)\pi}{2a}y\right\} \cdot \cosh\left\{\frac{(2n+1)\pi}{2a}(z-h_1)\right\} \right. \\
& \left. + \sum_{n=0}^{\infty} B_{\varphi(2n+1)} \sin\left\{\frac{(2n+1)\pi}{2a}y\right\} \cdot \cosh\left\{\frac{(2n+1)\pi}{2a}(h_2-z)\right\} \right] \\
& + \dot{\theta} \left[-xz + \sum_{n=0}^{\infty} A_{\theta(2n+1)} \sin\left\{\frac{(2n+1)\pi}{2b}x\right\} \cdot \cosh\left\{\frac{(2n+1)\pi}{2b}(z-h_1)\right\} \right. \\
& \left. + \sum_{n=0}^{\infty} B_{\theta(2n+1)} \sin\left\{\frac{(2n+1)\pi}{2b}x\right\} \cdot \cosh\left\{\frac{(2n+1)\pi}{2b}(h_2-z)\right\} \right] \\
& + \dot{\varphi} \left[\sum_{n=0}^{\infty} A_n \sin\left\{\frac{(2n+1)\pi}{2a}x\right\} \cdot \sinh\left\{\frac{(2n+1)\pi}{2a}y\right\} \right. \\
& + \sum_{m=0}^{\infty} B_m \sin\left\{\frac{(2m+1)\pi}{2b}y\right\} \cdot \sinh\left\{\frac{(2m+1)\pi}{2b}x\right\} \\
& + \sum_{n=0}^{\infty} \sum_{m=0}^{\infty} C_{mn} \sin\left\{\frac{(2n+1)\pi}{2a}x\right\} \cdot \sin\left\{\frac{(2m+1)\pi}{2b}y\right\} \cdot \cosh \\
& \quad \left. \cdot \sqrt{\frac{(2n+1)\pi^2}{2a} + \frac{(2m+1)\pi^2}{2b}}(z-b) \right] \\
& + \dot{\eta} \left[y + \sum_{n=0}^{\infty} A_{\eta(2n+1)} \sin\left\{\frac{(2n+1)\pi}{2a}y\right\} \cdot \cosh\left\{\frac{(2n+1)\pi}{2a}(z-h_1)\right\} \right] \\
& + \dot{\xi} \left[x + \sum_{n=0}^{\infty} A_{\xi(2n+1)} \sin\left\{\frac{(2n+1)\pi}{2b}x\right\} \cdot \cosh\left\{\frac{(2n+1)\pi}{2b}(z-h_1)\right\} \right] \\
& + \dot{\zeta} \left\{ (z-h_2) + \frac{g}{\omega^2} \right\} \tag{2.7}
\end{aligned}$$

ここで、A、B、Cの係数は、境界条件を考慮することによって決定される。

上記のポテンシャルを使って、波面形状 η 、静水圧を除いた変動圧力 P_d は次式で表すことができる。

$$\eta = -\frac{1}{g} \frac{\partial \phi}{\partial t} \Big|_{z=h_2} \tag{2.8}$$

$$P_d = -\rho_{in} \frac{\partial \phi}{\partial t} \tag{2.9}$$

これらの式より、求めた値を可視化した図を次ページに示す。

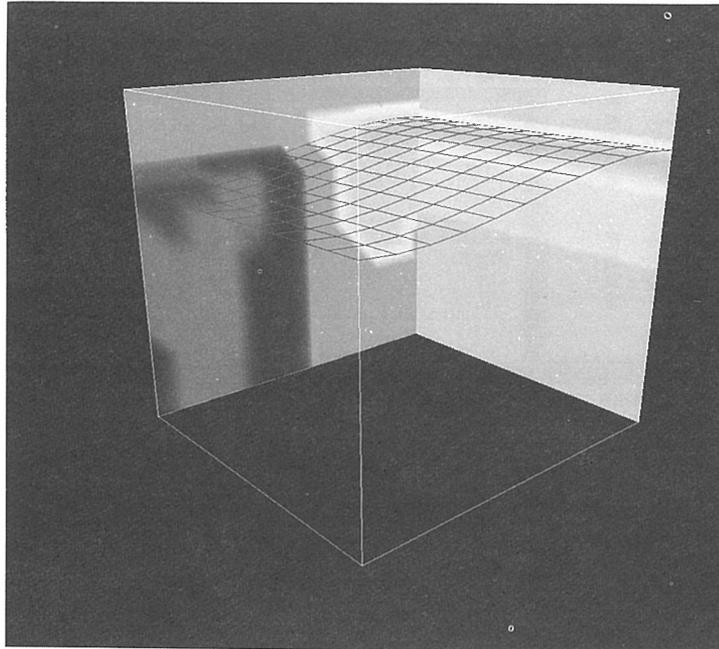


図3 タンク内壁に働く変動荷重の可視化

3 入力モジュールの説明

可視化に用いたデータは、以下のフォーマットでアスキーコードの文字列として表現されている。

```

12                x 軸方向のデータの個数
11                y 軸方向のデータの個数
11                z 軸方向のデータの個数
8.271022  0       波面のデータと内壁の変動圧力データ
8.060827  0
7.874774  0
7.280679  -0.3568306
(以下略)

```

データは、立方体を $12 \times 11 \times 11$ のセルに等分割して表現しているので、Explorer では 3次元定型格子データとして扱うことができる。データの並びは、最初に各軸方向のデータの個数、その続きに、対応する格子点の波面のデータと内壁の変動圧力データが並んでいる。波面のデータは、格子点での波面からの距離を表し、また変動圧力データは各格子

点での変動圧力の値である。圧力分布を表示したい壁面以外では、0 が入力されている。下図はデータを読み込み、Explorer 型データタイプとして出力する read3dn2 モジュールを Deta Scribe で作成したところである。



図4 入力モジュール read3dn2

3.1 入力テンプレート

Template メニューで、入力テンプレートを作成したのち、以下のように編集する。

- (1) まず、データの個数を読み込むために、Vector をドラッグする。そして、名前を res に変更し、そのインデックスをデフォルトの 1-N から 1-3 にする。
- (2) 各格子点のデータを読み込むために、パレットから 3D Array をドラッグする。そして、名前を data と変更し、インデックスを res[1]、res[2]、res[3] にする。また、その要素は、波面と圧力であるから、Vector をドラッグし、インデックスを 1-2 とする。Vector の要素は Float とする。
- (3) 波面みのデータを取り出すように入力テンプレートを設定する。3D Array の中の Vector の右丸をクリックしてインデックスを開く。そして、Array Component Dialog Button(右下の緑色のボタン) をマウスの右ボタンでドラッグして < New > を選択し、NAME を Wave data として 1 番目の要素を取り出すように設定する。(図4 参照) 同様に NAME を Pressure data として 2 番目の要素を取り出すように設定する。

- (4) 次に、3D Array の Array Component Dialog Button を右ボタンでドラッグして<New>を選択し、NAME を All wavein とする。そしてその Section element として Wave data を選択する。また同様に NAME を All Pressure としたものを作成し、その Section element として Pressure data を選択する。

3.2 出力テンプレート

出力テンプレートは波形用と圧力用の2つを用意する。それぞれに 3D Unif をドラッグし、名前を Wave out、Pressure out とする。また、両方の nCoodVar の値に 3 を入力する。

3.3 接続と保存

入力テンプレートの res と出力テンプレートの dims1、dims2 を接続する。同様に入力テンプレートの All Wavein と Template2 の data1 を、All Pressure と Template3 の data2 を接続する。そして、できたモジュールを read3dn2 として保存する。

4 マップの説明

まず、Data Scribe で作成した read3dn2 モジュールを呼び出す。そして、次のモジュールをモジュールライブラリからマップエディタ上にドラッグする。

BoundingBox, Render, Generate Colormap×2, Contour, OrthoSlice×2
LatToGeom ×2, WriteImg, For

また、/usr/explorer/unsupported/modules にある AnimFile を 2 つ呼び出す。

これらの入力及び出力ポートパッドをマウスの右ボタンを使って以下のように接続する。

| モジュール名/出力ポートパッド | モジュール名/入力ポートパッド |
|---|--|
| <i>AnimFile/out_file</i> | <i>read3dn2/Template1</i> |
| <i>read3dn2/Wave out-Lattice</i> | <i>BoundingBox/Input-Lattice</i> |
| | <i>Contour/Input-Lattice</i> |
| <i>read3dn2/Pressure out-Lattice</i> | <i>Orthoslice/Input-Lattice</i> |
| | <i>Orthoslice<2>/Input-Lattice</i> |
| <i>GenerateColormap/Colormap-Lattice</i> | <i>LatToGeom/Colormap-Lattice</i> |
| <i>GenerateColormap/Colormap-Lattice</i> | <i>LatToGeom<2>/Colormap-Lattice</i> |
| <i>GenerateColormap<2>/Colormap-Lattice</i> | <i>Contour/Colormap-Lattice</i> |
| <i>Orthoslice/Slice-Lattice</i> | <i>LatToGeom/Input-Lattice</i> |
| <i>Orthoslice<2>/Slice-Lattice</i> | <i>LatToGeom<2>/Input-Lattice</i> |
| <i>BoundingBox/Output-Geometry</i> | <i>Render/Input-Geometry</i> |
| <i>LatToGeom/Output-Geometry</i> | <i>Render/Input-Geometry</i> |

LatToGeom<2>/Output-Geometry
Contour/Contours-Geometry
Render/Sync-parameter
Render/Snapshot-Lattice
AnimFile<2>/out_file
WriteImg/filename
For/current value

Render/Input-Geometry
Render/Input-Geometry
Render/snap
WriteImg/Input-Lattice
WriteImg/File name
For/Loopin Value 0
AnimFile/value

WriteImg モジュールとは、Render モジュールの画像をファイル化するモジュールである。Render モジュールの入力ポートの snap に信号が入力された瞬間の画像をイメージファイルとして出力する。

今回は、作成したイメージファイルからアニメーションを作成するので、イメージファイルのフォーマットは SGI 形式が適している。そこで、WriteImg モジュールのファイルフォーマットを SGI にする。

AnimFile モジュールとは、in_filename と ext の間に Value の数字をはめ込んだファイル名を出力するモジュールである。今回のマップでは入力するデータファイルのファイル名 (HDPRE_) を AnimFile モジュールの in_filename に、出力するイメージファイルのファイル名 (HDPRE_) を AnimFile<2>モジュールの in_filename に入力する。また、VideoDesktop 上でファイルフォーマットを自動認識させるために、AnimFile<2>モジュールの ext には .rgb を入力する。

in_filename と ext は VideoDesktop を用いてアニメーションを作成する際に用いる Sequence コマンドの prefix と suffix にそれぞれ対応している。

For モジュールは、アニメーションのコマ番号を生成するために用いており、この数字を利用して、入力するデータファイルのファイル名と出力するイメージファイルのファイル名を対応させている。入力ポートの LoopinValue に信号が入力されると、Current Value の値が Step の数だけ増える。Start Value、End Value、Step Value にファイル名のスタート番号、エンド番号、ステップ数として 1、60、1 をそれぞれ入力する。そして、For の Refire をクリックすることにより、イメージファイルを自動作成することが出来る。

5 アニメーションの作成

作成した一連のイメージファイル (HDPRE1.rgb...HDPRE60.rgb) を、ビデオ制御ユーティリティ VideoDesktop を使うことによってコマ撮りし、アニメーションを作成する。

●詳細はセンターニュース本号の”ONYX によるアニメーションの作成方法について”を参照して下さい。

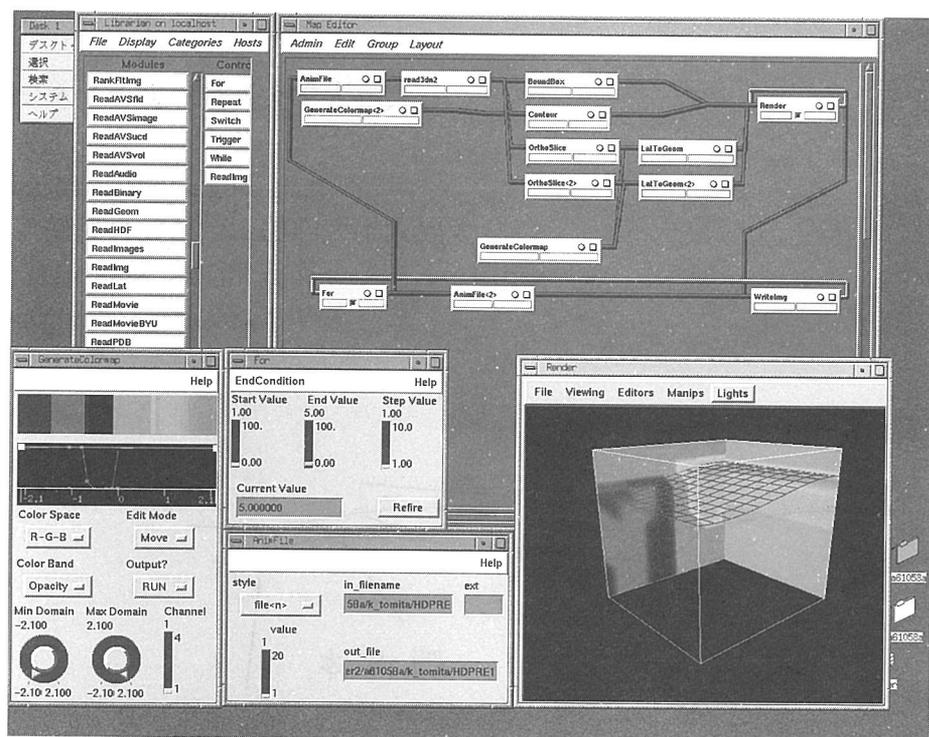


図 5 可視化マップ tank.map

謝辞

本文の内容に関して懇切なる御指導を頂いた本学内藤林助教授と画像の作成の際に適切な助言を下された本学博士後期課程 1 年の木原一氏、清水保宏技官に感謝致します。

参考文献

1. 出口 弘：汎用可視化ツール *Explorer* の使い方、大阪大学大型計算機センターニュース、22-2(1992)、p66-80
2. 出口 弘、富田 和雄：ONYX によるアニメーションの作成方法について、大阪大学大型計算機センターニュース、24-4(1995)