Development of naoeFOAM-os-SJTU Solver Based on Overset Grid Techniques for Self-Propulsion of Ship

Zhirong Shen and Decheng Wan

State Key Laboratory of Ocean Engineering, School of Naval Architecture, Ocean and Civil Engineering, Shanghai Jiao Tong University

International Research Exchange Meeting on Ship and Ocean Engineering, December 20-21, 2013, Osaka University, Japan
Contents

- Introduction
- Development of Solver Package: naoeFOAM-os-SJTU
- Numerical Examples
- Closing Remarks
Seakeeping, Self-propulsion and Maneuvering are still great challenges for computational ship hydrodynamics.

Limitations of traditional mesh methodologies

- Deforming and sliding grids

Advantages of overset grids

- Move grids without restriction
- Include hierarchy of objects, which allows appendages (moving rudder, rotating propeller) move independently with respect to the moving ship
Overset Grid

- A body fitted grid can be embedded into a Cartesian background mesh.
- Two grids are mutual independence.
- Body fitted grid can be moved without restriction.
- Two grids build the connectivity through the interpolation coefficients.
Dynamic overset grids
Dynamic overset grids
Development of Solver Package: naoeFOAM-os-SJTU
Object:

naoe-FOAM-SJTU + Overset

naoeFOAM-os-SJTU

To solve the problem of Self-Propulsion of Ship
Solver Package (naoe-FOAM-SJTU 1.0)
naoe-FOAM-SJTU Solver

naoe-FOAM-SJTU is a 3D Numerical Marine Basin based on OpenFOAM platform:

- take viscous effect into consideration, including violent flow (high Re flows, breaking waves)
- provide different types of waves (numerical wave generation and absorption)
- study wave (current, wind)-floating structures interaction easily (nonlinear, 6DOF, mooring)
Introduction to naeo-FOAM-SJTU

**Functions**

- **Numerical Wave Tank**
  - Provide wave conditions

- **6DoF Motion Module**
  - Model structures motion
  - provide mooring force,
  - keep body steady,
  - restrain structures motion

- **Mooringline system module**
focusing wave
Waves around Multi-Cylinders
Ship Large Motion in Waves

naoe-FOAM-SJTU
Dynamic overset grids

1st Refinement Grid

Background Grid
Dynamic overset grids
Dynamic overset grids

Donor & Interpolated points
Dynamic overset grids
Dynamic overset grids

Advantages of Overset Method
Implement Overset in naoe-FOAM-SJTU

- Read DCI from overset grid data.
- Computed interpolated values from donors.
- Solve N-S Equations.
- Solve VOF Equation.
- Solve Turbulence Equation.
- Parallelization.
- Validation.
liboverset: a library makes naoe-FOAM-SJTU capable of overset.
Implementation

How to implement overset capability in naoe-FOAM-SJTU solver by using liboverset?

Example:

- An incompressible laminar flow solver: icoFoam
- Step 1: Include two header files:

```cpp
#include "cellCenteredOverset.H"
#include "createOverset.H"
```
Implementation

Build Matrix:

\[
\frac{\partial \mathbf{U}}{\partial t} + \nabla \cdot (\mathbf{U} \mathbf{U}) - \nabla \cdot (\nu \nabla \mathbf{U}) = -\nabla p
\]

fvVectorMatrix UEqn

(  
    fvm::ddt(U)  
    + fvm::div(phi, U)  
    - fvm::laplacian(nu, U)  
    ==  
    - fvc::grad(p)  
);
Implementation

- **Step II: Modify Matrix and solve:**

  ```cpp
  overset.updateFvMatrix<vector>(UEqn);
  UEqn.solve();
  ```

- **Step III: Solve other equation (e.g. pressure):**

  ```cpp
  overset.updateFvMatrix<scalar>(pEqn);
  pEqn.solve();
  ```
**Flow Chart**

1. **Start**
   - **Setup Grid**
     - **Convert to OpenFOAM**
     - **Suggar**
       - **Initialize OpenFOAM**
       - **read DCI**
       - **Solve Equations**
       - **DCI**

2. **Build Matrix**
   - **Modify Matrix by DCI**
     - **Solve Equation**
     - **All Equations solved**
       - No
       - Yes
         - **Program Ends**
Numerical Experiments

A full CFD prediction of Self-propulsion characteristics

- Open water curves
- Towed condition
- Self-propulsion condition
Propeller Flows and Self-Propulsion of Ship Motion
Rotating Propeller in Open Water
Rotating Propeller in Open Water
Rotating Propeller in Open Water
- Single-point:
  - Fixed $V_a$ and RPS.
  - $J=0.6, 0.7, 0.8$
- Single-run:
  - $J = 0.05 \sim 1.05,$
  - Ramp time = 5 s
SELF-PROPULSION OF KCS
HSVA KCS Model
Towed and self-propulsion

Grids of self-propulsion
### Grid sizes

<table>
<thead>
<tr>
<th>Mesh size</th>
<th>Hull</th>
<th>Background</th>
<th>Propeller</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Towed</td>
<td>0.959 M</td>
<td>0.716 M</td>
<td>-</td>
<td>1.675 M</td>
</tr>
<tr>
<td>Self-propelled</td>
<td>1.129 M</td>
<td>0.716 M</td>
<td>1.368 M</td>
<td>3.213 M</td>
</tr>
</tbody>
</table>

- The grid used for the towed computations is the same grid but without the propeller and related refinement.
Towed condition (bare hull)

Wave elevation

Wave profile

Free-surface cut
Self-propulsion condition

- Fixed at even-keel condition.
- Performed at ship point.
- Different viscous force in model and ship scales.
- Skin friction correction:

\[
SFC = \left\{ (1 + k)(C_{F0M} - C_{F0S}) - \Delta C_F \right\} \times \frac{1}{2} \rho U_0^2 A_W
\]

- PI Controller to adjust RPS of propeller until final balance is reached:

\[
T = R_{T(SP)} - SFC
\]
- Semi-balanced horn rudder
- Propeller: SVP1193 (5-blade)
- Full 6DOF
- Rotating propeller
- Moving rudder.
Self-Propulsion of Ship Motion
Self-Propulsion of Ship Motion
Self-Propulsion of Ship Motion

Shanghai Jiao Tong University

naoe-FOAM-SJTU
Self-Propulsion of Ship Motion

Propeller speed

Ship speed
## Self-Propulsion of Ship Motion

<table>
<thead>
<tr>
<th></th>
<th>Experiment</th>
<th>Present Work</th>
<th>% error</th>
<th>CFDShip-Iowa (DES)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_T$</td>
<td>$3.942 \times 10^{-3}$</td>
<td>$3.840 \times 10^{-3}$</td>
<td>-2.586%</td>
<td>$4.011 \times 10^{-3}$</td>
</tr>
<tr>
<td>$K_T$</td>
<td>0.17</td>
<td>0.1682</td>
<td>-1.061%</td>
<td>0.1689</td>
</tr>
<tr>
<td>$K_Q$</td>
<td>0.0288</td>
<td>0.0290</td>
<td>0.863%</td>
<td>0.02961</td>
</tr>
<tr>
<td>$l-t$</td>
<td>0.853</td>
<td>0.8857</td>
<td>-3.838%</td>
<td>0.8725</td>
</tr>
<tr>
<td>$l-W_t$</td>
<td>0.792</td>
<td>0.7815</td>
<td>-1.326%</td>
<td>0.803</td>
</tr>
<tr>
<td>$\eta_o$</td>
<td>0.682</td>
<td>0.6785</td>
<td>-0.507%</td>
<td>0.683</td>
</tr>
<tr>
<td>$\eta_R$</td>
<td>1.011</td>
<td>0.9811</td>
<td>-2.955%</td>
<td>0.976</td>
</tr>
<tr>
<td>$J$</td>
<td>0.728</td>
<td>0.7363</td>
<td>1.142%</td>
<td>0.733</td>
</tr>
<tr>
<td>$n$</td>
<td>9.5</td>
<td>9.3231</td>
<td>-1.862%</td>
<td>9.62</td>
</tr>
<tr>
<td>$\eta$</td>
<td>0.74</td>
<td>0.7545</td>
<td>1.963%</td>
<td>0.724</td>
</tr>
</tbody>
</table>
SELF-PROPULSION FOR TWO PROPELLERS SHIP
Self-Propulsion of Ship Motion
Shanghai Jiao Tong University

Self-Propulsion of Ship Motion
Rudder and Yaw motion

Solid line – CFD; Dashed line -- EXP
Trajectory
Roll motion
Roll rate

Time (s)

Roll rate (deg/s)
Drift angle
Shanghai Jiao Tong University
Self-Propulsion of Ship Motion
• CFD: 360 deg
• EFD: 720 deg
Rudder and Yaw rate

Solid line – CFD; Dashed line -- EXP
Ship self-propulsion motion in waves
Motion histories

<table>
<thead>
<tr>
<th></th>
<th>TF3</th>
<th>TF5</th>
</tr>
</thead>
<tbody>
<tr>
<td>CFD</td>
<td>0.9785</td>
<td>0.7406</td>
</tr>
<tr>
<td>EFD</td>
<td>1.039</td>
<td>0.669</td>
</tr>
</tbody>
</table>
Closing Remarks
A solver package naoeFAOM-os-SJTU based on the implementation of the overset grid technique into naoe-FAOM-SJTU is presented.

A self-propulsion study of several ship models in both still water and waves was carried out. All self-propulsion factors were obtained through CFD computations. The results show good agreement between CFD and EFD.
Thank You!

naoe-FOAM-SJTU©