Numerical Simulation of Ice Ridge Breaking

Aleksei ALEKSEEV, EMSHIP student

Robert BRONSART, Prof. University of Rostock
Quentin HISTETTE, Hamburg Ship Model Basin

February 2\textsuperscript{nd} 2016
Rostock
Goal

To develop a **numerical solver** capable of simulating **ship breaking** through an **ice ridge**

Solution Steps

• Ice ridges
• Discrete Element Method
• Software development
• Validation & Results
• Conclusions & Proposals
Ice ridges

Source: https://www.youtube.com/watch?v=9vJ3QkRCuPs
Ice ridges

• Dimensions

Source: Ship Breaking Through Ice Ridges by D.Ehle

• Configuration of ice ridge

Air
Sail blocks
Consolidated layer
Rafted level ice
Water
Keel
**Discrete Element Method**

**DEM** – numerical method for calculation of motion of large number of particles

**Application in:**
- Soil mechanics
- Rock engineering
- Geophysics
- Mineral processing
- Powder metallurgy

Source: [http://www.metariver.kr/](http://www.metariver.kr/)
**Discrete Element Method**

**DEM** – numerical method for calculation of motion of large number of particles

**Application in:**
- Soil mechanics
- Rock engineering
- Geophysics
- Mineral processing
- Powder metallurgy
- Ice-related simulations?
Discrete Element Method

Program start

Initialization of elements

Graphical output (initialized simulation domain)

Update elements

Graphical output, velocity and acceleration output

Update bounding boxes

Update neighborhood list

Compute forces and torques

New position

Program End

Contact

\[ t = 0 \]

\[ t = 0 + dt \]
Discrete Element Method

- **Ice ridge** as an assembly of discrete elements

- **Ship** as a discrete element with special features
Introducing ship hull into simulation

Hull Surface (NURBS, etc.)

Hull Mesh (triangular)

OBJ. file

DEM data structures

VERT_COORD
vertex index 1 → 4

FACE_EQUATION
face index 1 → 4

FACE_VERTEX_TABLE
face index 1 → 4

VERT_FACE_TABLE
vertex index 1 → 4
**Quaternions & Spatial Orientation**

\[ q = w + xI + yJ + zK \]

\[ I \cdot I = -1 \quad J \cdot J = -1 \quad K \cdot K = -1 \]

**Definition of quaternion components:**

\[
w = \cos \frac{\varphi}{2} \cos \frac{\theta}{2} \cos \frac{\psi}{2} + \sin \frac{\varphi}{2} \sin \frac{\theta}{2} \sin \frac{\psi}{2}
\]

\[
x = \sin \frac{\varphi}{2} \cos \frac{\theta}{2} \cos \frac{\psi}{2} - \cos \frac{\varphi}{2} \sin \frac{\theta}{2} \sin \frac{\psi}{2}
\]

\[
y = \cos \frac{\varphi}{2} \sin \frac{\theta}{2} \cos \frac{\psi}{2} + \sin \frac{\varphi}{2} \cos \frac{\theta}{2} \sin \frac{\psi}{2}
\]

\[
z = \cos \frac{\varphi}{2} \cos \frac{\theta}{2} \sin \frac{\psi}{2} - \sin \frac{\varphi}{2} \sin \frac{\theta}{2} \cos \frac{\psi}{2}
\]

4464 vertices

3908 faces

\[ \tilde{r} = q \, r \, q^* \text{conjugation} \]
Ship Buoyancy and Propulsion

1.1

ΔT

1.2

ΔT

2.1

Δφ

2.2

Δφ

3.1

Δθ

3.2

Δθ
Ship Buoyancy and Propulsion
Gift wrapping algorithm

First point
Second point
Third point

\[ A_i = \frac{V_1V_2 \times V_2V_3}{2} \]

Cross sectional area

\[ A = \sum_{i=1}^{N_A} A_i \]

Simpson's First Rule integrator

\[ V = \frac{h}{3} (1 \cdot \text{Area}_1 + 4 \cdot \text{Area}_2 + 1 \cdot \text{Area}_3) \]

Roll

Displacement

Draft

Pitch

Buoyancy restoring moment

\[ \vec{M} = \vec{F_b} \times \vec{GB} \]
Ship Buoyancy and Propulsion

![Graph showing KT, J1, KT1, J2, KT2, and efficiency curves.](image)

- **KT**: Indicates the performance metric for the system.
- **10KQ**: Represents another performance metric.
- **Efficiency**: Shows the overall efficiency levels across the range of J values.

Key points:
- **J1, KT1**: Indicates a peak in performance at a specific J value.
- **J2, KT2**: Shows another peak in performance at a different J value.

**Software**:
- Software 1
- Software 2
- Software 3

**Sections**:
- Ice ridges
- DEM
- Results
- Conclusions
Equations of Motions

• Rectilinear degrees of freedom

\[ \mathbf{F}_i = m \cdot \mathbf{v}_i \]

• Rotational degrees of freedom

\[ \frac{d^2 \mathbf{q}}{dt^2} = \frac{1}{2} \left( \mathbf{\ddot{q}} \mathbf{q} + \dot{\mathbf{q}} \mathbf{\dot{\omega}} \right) \]
1. Predictor step

- Rectilinear degrees of freedom
  \[ r = r + \dot{r}dt + \ddot{r} \frac{dt^2}{2} \]
  \[ \dot{r} = \dot{r} + \ddot{r}dt \]

- Rotational degrees of freedom
  \[ q = q + \dot{q}dt + \ddot{q} \frac{dt^2}{2} \]
  \[ \dot{q} = \dot{q} + \ddot{q}dt \]

2. Corrector step

- Rectilinear degrees of freedom
  \[ r = r + c_0 \Delta \ddot{r} \]
  \[ \dot{r} = \dot{r} + c_1 \Delta \dddot{r} \]

- Rotational degrees of freedom
  \[ q = q + c_0 \Delta \dot{q} \]
  \[ \dot{q} = \dot{q} + c_1 \Delta \ddot{q} \]
  \[ \ddot{q} = \ddot{q} + c_2 \Delta \dddot{q} \]
Forces Calculation

Element 1

Element 2

Overlap polyhedron

Elastic
Friction
Buoyancy

Cohesion
Drag
Gravity

Ice ridges  DEM  Software 1  Software 2  Software 3  Results  Conclusions
Hanging non-convex ship hulls

• Translation

• Rotation
Lindqvist ice resistance theory

\[ R_{ice} = R_c + R_b + R_s \]

Source: Simulation of Ice Management Operations by Q. Hisette
Visualization
Simulation

**Ice ridges**

**DEM**

**Software 1**

**Software 2**

**Software 3**

**Results**

**Conclusions**
Validation: ridge 1

Ship Velocity During Ridge Breaking

- test 1
- test 3
- test 4
- test 5
- test 6
- test 7
- Experiment

Velocity [m/s]

Time [s]
Conclusions

• Flexible software for ship breaking through an ice ridge
• DEM is suitable to model ice/hull interaction
• Calibration of forces models and validation is required

Proposals

• Computational speed
• Level ice resistance
• Development towards brash ice, ice floes, etc.