Dynamic Positioning Simulator

Professional Training Tool

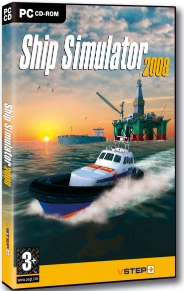
Jalitha Wills

TU Delft / VSTEP

8 juni 2007

About VSTEP

- VSTEP: Virtual Safety Training & Education Platform;
- Locations: Rotterdam (headquarters), Oxford;
  Development teams in India, China and Eastern Europe;
- 3 subjects: Scenario Training, Procedure Training and Training Simulator;
- Simulator game: Ship Simulator.



Outline

Definitions

A means of holding a vessel in relatively fixed position with respect to the ocean floor, without using anchors accomplished by two or more propulsive devices controlled by inputs from sonic instruments on the sea bottom and on the vessel, by gyrocompass, by satellite navigation or by other means.

What is Dynamic Positioning?

Different operational modes:

- manual/joystick mode
- auto-heading mode
- auto-position mode
- auto area position mode
- autopilot mode

Why Dynamic Positioning?

Advantages Dynamic Positioning:

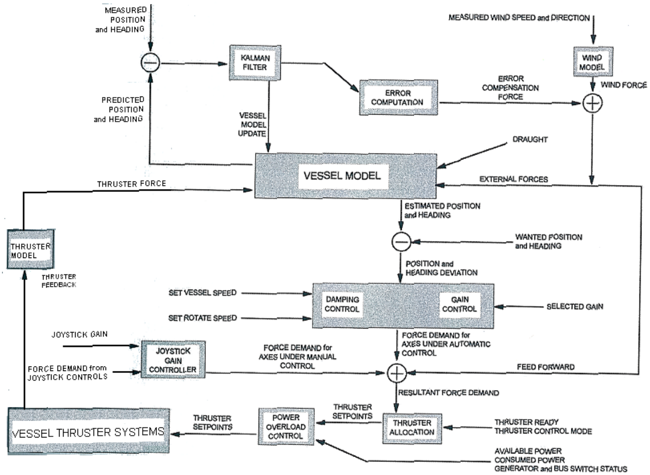
- No tugboats needed;
- Offshore set-up is quick;
- Power saving;
- Precision situations more easily.

Dynamic Positioning Training

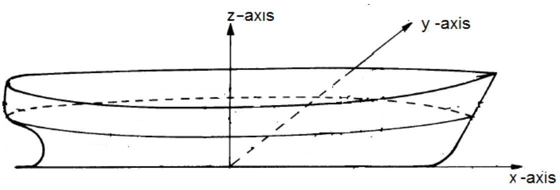
Test cases:

- Learn the system, joystick steering;
- Failure of the system;
- Erroneous input from sensors;
- Extreme weather situations.

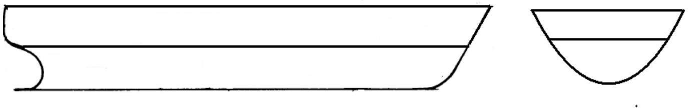
Mathematical Model behind Dynamic Positioning



Coordinate System



Forces on Ship: Current Force



$$F_c = \begin{pmatrix} \frac{1}{2}\rho V_c^2 C_{X_c}(\alpha_c) A_{TS} \\ \frac{1}{2}\rho V_c^2 C_{Y_c}(\alpha_c) A_{LS} \end{pmatrix}$$

$$M_c = \frac{1}{2}\rho V_c^2 C_{M_c}(\alpha_c) A_{LS} L$$

With: ρ density of water, V_c current velocity, α_c current direction, A_{TS} submerged transverse projected area, A_{LS} submerged longitudinal projected area, L length of ship, $C_{*c}(\alpha_c)$ current coefficient.

Forces on Ship: Wind Force

$$F_w = \begin{pmatrix} \frac{1}{2}\rho_{air} V_{rw}^2 C_{X_w}(\alpha_{rw}) A_T \\ \frac{1}{2}\rho_{air} V_{rw}^2 C_{Y_w}(\alpha_{rw}) A_L \end{pmatrix}$$

$$M_w = \frac{1}{2}\rho_{air} V_{rw}^2 C_{M_w}(\alpha_{rw}) A_L L$$

With: ρ_{air} density of air, V_{rw} relative wind velocity, α_{rw} relative wind direction, A_T transverse projected wind area, A_L longitudinal projected wind area, L length of ship, $C_{*w}(\alpha_{rw})$ wind coefficient.

$$V_w(z) = V_w(z = 10m) \cdot \left(\frac{z}{10}\right)^{\frac{1}{8}}$$

With: $V_w(z = 10m)$ velocity at 10m.

Forces on Ship: Wave Force

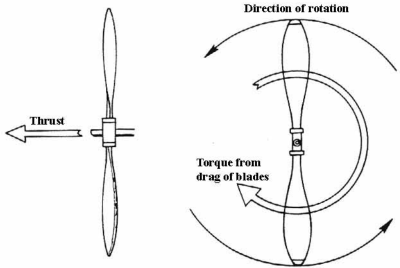
$$F_{wd} = \begin{pmatrix} C_{X_{wd}}(\alpha_{wd}, f_{wd})\frac{1}{8}\rho g H_{1/3}^2 L \\ C_{Y_{wd}}(\alpha_{wd}, f_{wd})\frac{1}{8}\rho g H_{1/3}^2 L \end{pmatrix}$$

$$M_{wd} = C_{M_{wd}}(\alpha_{wd}, f_{wd})\frac{1}{8}\rho g H_{1/3}^2 L^2$$

With: ρ density of water, α_{wd} wave direction, f_{wd} regular wave frequency, $C_{*wd}(\alpha_{wd}, f_{wd})$ wave drift coefficient, g gravity coefficient, $H_{1/3}$ significant wave height and L length of ship.

Thrusters

Thruster Force



$$
\begin{aligned}
\text{Thrust}: T &= C_T \rho n^2 D^4 \\
\text{Torque}: Q &= C_Q \rho n^2 D^5
\end{aligned}
$$

With: ρ density of water, n rpm, D diameter of propeller, C_T thrust coefficient and C_Q torque coefficient.

A First Model

Limitations:

- Thruster force defined as x-force and y-force;
- No max thruster force;
- No moment for external force;
- No penalty for reverse thrust;
- Ship as pointmass;
- Only force calculation, no sailing yet;
- Inertia thrusters and ship.

Solution

Algorithm - (1/3)

1. Calculate necessary thruster forces:
   Force x-direction: $F_x = F_{x_{demand}} - F_{x_{wind}} - F_{x_{current}} - F_{x_{wave}}$
   Force y-direction: $F_y = F_{y_{demand}} - F_{y_{wind}} - F_{y_{current}} - F_{y_{wave}}$
   Moment: $M = M_{demand} - M_{wind} - M_{current} - M_{wave}$

Solution

Algorithm - (1/3)

1. Calculate necessary thruster forces:
   Force x-direction: $F_x = F_{x_{demand}} - F_{x_{wind}} - F_{x_{current}} - F_{x_{wave}}$
   Force y-direction: $F_y = F_{y_{demand}} - F_{y_{wind}} - F_{y_{current}} - F_{y_{wave}}$
   Moment: $M = M_{demand} - M_{wind} - M_{current} - M_{wave}$
2. Now it must hold:
   $F_x = \sum_{i=1}^{n} (F_x)_i$
   $F_y = \sum_{i=1}^{n} (F_y)_i$
   $M = \sum_{i=1}^{n} (-y_i \cdot (F_x)_i + x_i \cdot (F_y)_i)$

Solution

Algorithm - (2/3)

3. Express F_{x_n} and F_{y_n} in other variables:
   $(F_x)_n = F_x - \sum_{i=1}^{n-1} (F_x)_i$
   $(F_y)_n = F_y - \sum_{i=1}^{n-1} (F_y)_i$

Solution

Algorithm - (2/3)

3. Express F_{x_n} and F_{y_n} in other variables:
   $(F_x)_n = F_x - \sum_{i=1}^{n-1} (F_x)_i$
   $(F_y)_n = F_y - \sum_{i=1}^{n-1} (F_y)_i$
4. With this and moment equation:
   $(F_y)_{n-1} = \frac{M + y_n \cdot F_x - x_n \cdot F_y}{x_{n-1} - x_n} + \sum_{i=1}^{n-1} \left(\frac{y_i - y_n}{x_{n-1} - x_n} \cdot (F_x)_i \right) + \sum_{i=1}^{n-2} \left(\frac{x_n - x_i}{x_{n-1} - x_n} \cdot (F_y)_i \right)$

VSTEP TUDelft

Solution

Algorithm - (3/3)

5. Total power:
   $g((F_x)_1, \ldots, (F_x)_{n-1}, (F_y)_1, \ldots, (F_y)_{n-2}) = \sum_{i=1}^{n} \sqrt{(F_x)_i^2 + (F_y)_i^2}$
6. Minimize this total power with Method of Steepest Descent with Linesearch.

Steepest Descent with Linesearch

Searching for $\omega = \omega_{opt}$ s.t. $f(x^{old} - \omega \nabla f)$ is minimal.

Algorithm

1. Start with $\omega_{min} = 0$ and $\omega_{max} = 1$.
   Calculate $f_i = f(x^{old} - \omega_i \nabla f(x^{old}))$ with $i = \{min, max\}$.
    - IF $f_{max} > f_{min}$: STOP.
    - ELSE: $\omega_{min} = \omega_{max}$; $\omega_{max} = 2 \cdot \omega_{max}$.
      $f_{min} = f_{max}$; $f_{max} = f(x^{old} - \omega_{max} \nabla f(x^{old}))$.
      REPEAT.

   Now: $\omega_{min} \leq \omega_{opt} \leq \omega_{max}$.

TUDelft

Steepest Descent with Linesearch

Searching for $\omega = \omega_{opt}$ s.t. $f(x^{old} - \omega \nabla f)$ is minimal.

Algorithm

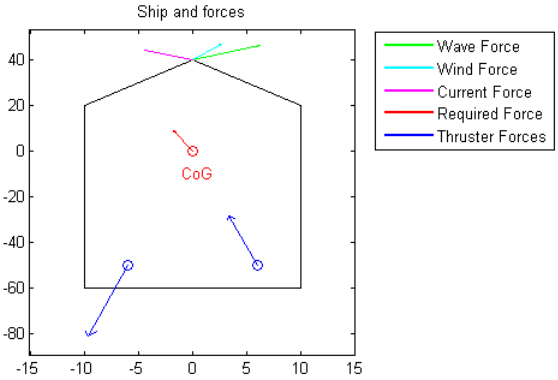
1. Start with $\omega_{min} = 0$ and $\omega_{max} = 1$.
   Calculate $f_i = f(x^{old} - \omega_i \nabla f(x^{old}))$ with $i = \{min, max\}$.
   - IF $f_{max} > f_{min}$: STOP.
   - ELSE: $\omega_{min} = \omega_{max}$; $\omega_{max} = 2 \cdot \omega_{max}$.
     $f_{min} = f_{max}$; $f_{max} = f(x^{old} - \omega_{max} \nabla f(x^{old}))$.
     REPEAT.

   Now: $\omega_{min} \leq \omega_{opt} \leq \omega_{max}$.
2. Find ω_{opt} in the interval:
   - IF $||\omega_{min} - \omega_{max}|| \leq \epsilon$: STOP.
   - ELSE: Split interval in two s.t. ω_{opt} remains between ω_{min} and ω_{max}.
     REPEAT.

TU Delft

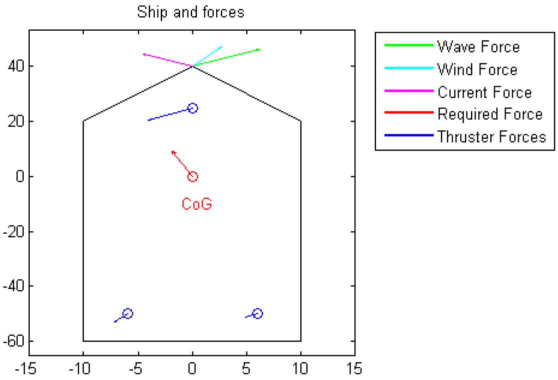
Results

Minimal power is: 58.7518; Positions: (-6,-50); (6,-50)
Forces: (-4.1,-34.167); (-2.9,24.167)



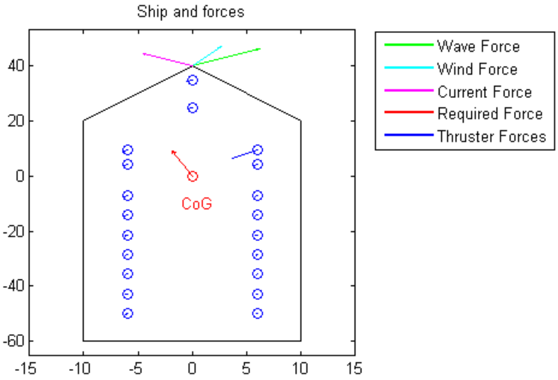
Results

Minimal power is: 12.3296; Positions: (-6,-50); (6,-50); (0,25)
Forces: (-1,339,-3.325); (-1.16,-1.377); (-4.511,5.298)



Results

Minimal power is: 12.2234



Future Goals

1. More realism;
2. Transition Matlab $\Leftrightarrow$ C++;
3. Sailing the ship;
4. Test cases.

Questions?