OASIS
Offshore Advanced Simulation Software
—
Theory Manual

Álvaro Rodríguez Luis

April 2026

Offshore Engineering and Ocean Energy Group
Environmental Hydraulics Institute of Cantabria

Contents

1 Introduction **3**

2 Coordinate Systems and Rigid Body Dynamics **4**

2.1 Reference frames . . . 4

2.2 Euler angles and rotation matrices . . . 4

2.3 Newton–Euler equations for the rigid body . . . 4

3 Waves **5**

3.1 Regular waves . . . 5

3.2 Irregular waves (JONSWAP) . . . 5

3.3 Directional spreading . . . 6

4 Hydrostatics **6**

4.1 Linear hydrostatic restoring . . . 6

4.2 Non-linear Froude–Krylov . . . 6

5 Hydrodynamics **7**

5.1 Cummins equation . . . 7

5.2 Radiation forces . . . 7

5.3 First-order excitation forces . . . 7

5.4 Second-order excitation forces (QTFs) . . . 8

5.5 Viscous corrections . . . 8

5.6 Morison wind and current forces . . . 8

6 Mooring and Towing Lines **8**

6.1 Governing equation . . . 8

6.2 External forces . . . 9

6.3 Internal forces and tension models . . . 9

6.3.1 Rayleigh damping model . . . 9

6.3.2 Viscoelastic (Banks) model . . . 10

6.3.3 Tabulated stress–strain curves . . . 10

6.4 Weak formulation . . . 10

6.5 SEM spatial discretisation 10
6.6 Derivative operator 11
6.7 Added mass iteration 11
6.8 Boundary conditions 11
6.8.1 Anchor, fairlead or actuator 11
6.8.2 Multi-line joint 12
6.8.3 Winch 12
6.8.4 Elastic anchor 12
6.9 Quasi-static initial conditions 12

7 Seafloor Interaction **13**
7.1 Flat seafloor 13
7.2 Inclined plane 13
7.3 Complex bathymetry 13
7.4 Ground normal and friction forces 13

8 Springs and Connectors **14**
8.1 Deformation measurement 14
8.2 Spring forces 14

9 Winches **15**
9.1 Mechanical model 15
9.2 Controllers 15

10 Oscillating Water Columns **15**
10.1 Pneumatic chamber model 15
10.2 Turbine model 16

11 Wind Turbines **16**

12 Sinking **16**

13 Time Integration **17**
13.1 Adaptive BDF2 17

13.2 ESDIRK 17

14 Coupling **18**

14.1 Global state vector 18

14.2 RHS evaluation order 18

14.3 Multi-rate coupling 19

1 Introduction

OASIS (Offshore Advanced Simulation Software) is a time-domain numerical simulation tool for offshore floating structures developed in C++ at IHCantabria. It couples potential-flow hydrodynamics with structural and mooring dynamics to predict the motion response of floating platforms under waves, wind and current.

The software is structured as a collection of coupled modules, each solving a specific physical subsystem. The modules are:

- **Rigid body dynamics** — six degree-of-freedom (6-DOF) motion of floating bodies.
- **Waves** — generation of regular and irregular sea states.
- **Hydrostatics** — linear restoring forces and non-linear Froude–Krylov pressure integration.
- **Hydrodynamics** — potential-flow radiation/diffraction framework including first- and second-order wave loads, Morison drag for wind and current.
- **Mooring and towing lines** — spectral element method (SEM) formulation for dynamic cable simulation.
- **Seafloor interaction** — seabed projection algorithms for flat, inclined and complex bathymetries.
- **Springs and connectors** — 6-DOF spring/damper/friction model for multi-body coupling.
- **Winches** — mechanical winch model with controllers.
- **Oscillating water columns (OWC)** — pneumatic chamber and turbine model.
- **Wind turbines** — coupling interface to OpenFAST.
- **Sinking** — progressive flooding model with time-varying hydrostatic and hydrodynamic properties.

All subsystems are assembled into a single initial value problem (IVP) and integrated in time using implicit ODE solvers (BDF2 or ESDIRK) with adaptive step-size control. The coupling among subsystems is described in Section 14.

This document describes the mathematical formulation behind each module and how they are coupled together. References to the corresponding source code files are given throughout.

2 Coordinate Systems and Rigid Body Dynamics

2.1 Reference frames

Two types of reference frames are used:

1. **Global inertial frame**: a fixed reference frame with its origin at the mean free surface. The first two basis vectors ($\mathbf{e}_1$, $\mathbf{e}_2$) are tangent to the undisturbed water surface, and $\mathbf{e}_3$ points upward.
2. **Local body frame**: a non-inertial frame with its origin at the centre of gravity (COG) of each body. When the body is at rest, the local axes coincide with the global axes. The frame moves with the body.

2.2 Euler angles and rotation matrices

Body rotations are described using Tait–Bryan (nautical) angles: roll (α), pitch (β) and yaw (γ). These define three successive rotations around the global axes X, Y and Z:

$$\mathbf{R}_1(\alpha) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{pmatrix}, \quad \mathbf{R}_2(\beta) = \begin{pmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{pmatrix}, \quad \mathbf{R}_3(\gamma) = \begin{pmatrix} \cos\gamma & -\sin\gamma & 0 \\ \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{1}$$

The combined rotation matrix is $\mathbf{R}(\alpha, \beta, \gamma) = \mathbf{R}_3(\gamma) \cdot \mathbf{R}_2(\beta) \cdot \mathbf{R}_1(\alpha)$.

The position of the body is defined by:

$$\mathbf{r} = \begin{pmatrix} x \\ y \\ z \end{pmatrix}, \quad \mathbf{\Phi} = \begin{pmatrix} \alpha \\ \beta \\ \gamma \end{pmatrix} \tag{2}$$

where $\mathbf{r}$ contains the COG coordinates in the global frame and $\mathbf{\Phi}$ the Euler angles. A point with local coordinates $\mathbf{p}_L$ has global coordinates:

$$\mathbf{p}_G = \mathbf{r} + \mathbf{R}(\mathbf{\Phi}) \cdot \mathbf{p}_L \tag{3}$$

Its velocity and acceleration in the global frame are:

$$\dot{\mathbf{p}}_G = \dot{\mathbf{r}} + \dot{\mathbf{R}}(\mathbf{\Phi}, \dot{\mathbf{\Phi}}) \cdot \mathbf{p}_L, \qquad \ddot{\mathbf{p}}_G = \ddot{\mathbf{r}} + \ddot{\mathbf{R}}(\mathbf{\Phi}, \dot{\mathbf{\Phi}}, \ddot{\mathbf{\Phi}}) \cdot \mathbf{p}_L \tag{4}$$

2.3 Newton–Euler equations for the rigid body

Forces and moments can be computed at the COG or at arbitrary body points. A force $\mathbf{F}$ and moment $\mathbf{M}$ applied at a body point with local coordinates $\mathbf{p}_L$ produce equivalent quantities at the COG:

$$\mathbf{F}_{\text{cm}} = \mathbf{F}, \qquad \mathbf{M}_{\text{cm}} = \mathbf{M} + \mathbf{p}_L \times \left(\mathbf{R}^T(\boldsymbol{\Phi}) \cdot \mathbf{F}\right) \tag{5}$$

The net forces $\sum \mathbf{F}_{\text{cm},i}$ and moments $\sum \mathbf{M}_{\text{cm},i}$ at the COG are assembled from all acting subsystems (hydrodynamics, mooring lines, springs, wind turbines, OWC, etc.). The body accelerations are obtained from:

$$\left[\begin{pmatrix} m \cdot \mathbf{I}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{I}_b \end{pmatrix} + \mathbf{A}\right] \cdot \begin{pmatrix} \ddot{\mathbf{r}} \\ \ddot{\boldsymbol{\Phi}} \end{pmatrix} = \begin{pmatrix} \sum \mathbf{F}_{\text{cm},i} \\ \sum \mathbf{M}_{\text{cm},i} \end{pmatrix} \tag{6}$$

where m is the structural mass, $\mathbf{I}_b$ is the inertia matrix in the principal body axes, and $\mathbf{A}$ is the added mass matrix (at infinite frequency) plus any viscous added mass corrections:

$$\mathbf{A} = \mathbf{A}_\infty + \mathbf{A}_{\text{visc}} \tag{7}$$

For multi-body systems, the matrices are assembled in block-diagonal form, with off-diagonal coupling through the hydrodynamic added mass.

3 Waves

3.1 Regular waves

A regular wave is described by a single component with height H, period T and heading φ. The free surface elevation at position (x, y) and time t is:

$$\eta(x, y, t) = \frac{H}{2} \cos\left(k_x x + k_y y - \omega t + \phi_0\right) \cdot r(t) \tag{8}$$

where $\omega = 2\pi/T$ is the angular frequency, $k_x = k\cos\varphi$, $k_y = k\sin\varphi$, ϕ_0 is the initial phase, and $r(t)$ is a ramp function for smooth startup. The wavenumber k is obtained from the linear dispersion relation:

$$\omega^2 = g\,k \tanh(k\,h) \tag{9}$$

solved iteratively using Newton–Raphson's method, where g is gravitational acceleration and h is water depth.

3.2 Irregular waves (JONSWAP)

Irregular sea states are represented as a superposition of n_C wave components:

$$\eta(x, y, t) = \sum_{i=1}^{n_C} \sum_{j=1}^{n_H} a_{ij} \cos\left(k_{x,i} \cos\varphi_j\, x + k_{y,i} \sin\varphi_j\, y - \omega_i t + \phi_{ij}\right) \cdot r(t) \tag{10}$$

where $a_{ij} = \sqrt{2\, S(\omega_i)\, D(\varphi_j)\, \Delta\omega\, \Delta\varphi}$ is the amplitude from the directional spectrum.

The JONSWAP spectrum is defined as:

$$S(\omega) = \frac{\alpha g^2}{\omega^5} \exp\left[-\frac{5}{4}\left(\frac{\omega_p}{\omega}\right)^4\right] \gamma^{\exp\left[-\frac{(\omega-\omega_p)^2}{2\sigma^2\omega_p^2}\right]} \tag{11}$$

where $\omega_p = 2\pi/T_p$ is the peak frequency, γ is the peak enhancement factor, $\sigma = 0.07$ for $\omega \leq \omega_p$ and $\sigma = 0.09$ for $\omega > \omega_p$, and α is chosen so that the zeroth spectral moment matches $H_s^2/16$.

3.3 Directional spreading

Directional spreading is modelled using the cosine-power function:

$$D(\varphi) = C_s \cos^{2s}\left(\frac{\varphi - \varphi_0}{2}\right) \tag{12}$$

where s is the spreading parameter and C_s is a normalisation constant such that $\int D(\varphi)\, d\varphi = 1$.

The dynamic pressure at a point (x, y, z) due to the wave field is computed using linear wave theory:

$$p(x, y, z, t) = \rho_w g \sum_{i=1}^{n_C} a_i \frac{\cosh[k_i(z + h)]}{\cosh(k_i h)} \cos(k_{x,i}x + k_{y,i}y - \omega_i t + \phi_i) \tag{13}$$

4 Hydrostatics

4.1 Linear hydrostatic restoring

For small displacements from equilibrium, the hydrostatic restoring force is:

$$\mathbf{F}_{\mathrm{hs}} = -\mathbf{G} \cdot \boldsymbol{\nu} \tag{14}$$

where $\mathbf{G}$ is the 6×6 hydrostatic stiffness matrix (computed from the equilibrium waterplane area, COG and centre of buoyancy) and $\boldsymbol{\nu} = (\mathbf{r}^T, \boldsymbol{\Phi}^T)^T$ is the 6-DOF displacement vector.

4.2 Non-linear Froude–Krylov

For large motions, the hydrostatic forces are computed by pressure integration over the instantaneous wetted surface of the body. The body mesh is transformed to the current position using the rotation matrix $\mathbf{R}$, clipped at the free surface elevation $\eta(x, y, t)$, and the hydrostatic and Froude–Krylov pressure is integrated over the resulting triangulated surface:

$$\mathbf{F}_{\mathrm{FK}} = -\oint_{S_w} p\, \mathbf{n}\, dS \tag{15}$$

where $p = -\rho_w g z + p_{\mathrm{dynamic}}$ is the total pressure (hydrostatic plus wave dynamic contribution),

$\mathbf{n}$ is the outward surface normal and S_w is the wetted surface.

5 Hydrodynamics

The hydrodynamic model is based on linear potential flow theory. The BEM-computed frequency-domain coefficients (from NEMOH, ANSYS AQWA, SESAM, etc.) are transformed to the time domain using the Cummins equation [5].

5.1 Cummins equation

The equation of motion for the floating body in the time domain is:

$$(\mathbf{M} + \mathbf{A}_\infty + \mathbf{A}_{\text{visc}}) \cdot \ddot{\boldsymbol{\nu}}(t) = \mathbf{F}_{\text{rad}}(t) + \mathbf{F}_{\text{hs}}(t) + \mathbf{F}_{\text{exc}}(t) + \mathbf{F}_{\text{visc}}(t) + \mathbf{F}_{\text{mor}}(t) + \mathbf{F}_{\text{ext}}(t) \tag{16}$$

where $\mathbf{M}$ is the structural mass matrix, $\mathbf{A}_\infty = \lim_{\omega\to\infty} \mathbf{A}(\omega)$ is the infinite-frequency added mass, $\mathbf{A}_{\text{visc}}$ is an empirical added mass correction, and the force terms are defined below.

5.2 Radiation forces

The radiation force accounts for the memory effects of wave radiation:

$$\mathbf{F}_{\text{rad}}(t) = -\int_{t-t_0}^{t} \mathbf{K}(t-\tau) \cdot \dot{\boldsymbol{\nu}}(\tau)\, d\tau \tag{17}$$

where $\mathbf{K}(t)$ is the impulse response function (IRF), computed from the radiation damping matrix $\mathbf{B}(\omega)$:

$$\mathbf{K}(t) = \frac{2}{\pi} \int_0^\infty [\mathbf{B}(\omega) - \mathbf{B}_\infty] \cos(\omega t)\, d\omega \tag{18}$$

The integral is truncated at a finite time t_0 where the IRF has decayed to negligible values.

5.3 First-order excitation forces

The first-order wave excitation force is computed from the diffraction database:

$$\mathbf{F}_{\text{1st}}(t) = \frac{1}{2} \sum_{i=1}^{n_C} |\mathbf{F}_{\text{exc}}(\omega_i, \varphi_i + \gamma)| \cdot H_i \cdot \cos\left(-\omega_i t + \phi_i + \hat{\mathbf{F}}_{\text{exc}}(\omega_i, \varphi_i + \gamma) + k_{x,i} x + k_{y,i} y\right) \tag{19}$$

where $|\mathbf{F}_{\text{exc}}|$ and $\hat{\mathbf{F}}_{\text{exc}}$ are the magnitude and phase of the excitation force transfer function, H_i is the wave height of component i, and (x, y) is the instantaneous body position (allowing for position-dependent updates).

5.4 Second-order excitation forces (QTFs)

Second-order wave loads are computed using quadratic transfer functions (QTFs). For difference-frequency (slow-drift) forces:

$$\mathbf{F}_{2\mathrm{nd}}(t) = \frac{1}{4}\sum_{i=1}^{n_C}\sum_{j=1}^{n_C} H_i H_j \left|\mathbf{QTF}_{\mathrm{dif}}(\omega_i,\omega_j)\right| \cos\left[-(\omega_i-\omega_j)t + (\phi_i-\phi_j) + \widehat{\mathbf{QTF}}_{\mathrm{dif}}(\omega_i,\omega_j) + \Delta\mathbf{k}\cdot\mathbf{x}\right] \tag{20}$$

where $\Delta\mathbf{k}\cdot\mathbf{x} = (k_{x,i}-k_{x,j})x + (k_{y,i}-k_{y,j})y$ and the QTF matrices are interpolated to the actual wave heading.

5.5 Viscous corrections

To compensate for the omission of viscous effects in potential flow theory, empirical damping terms are added:

$$\mathbf{F}_{\mathrm{visc}}(\dot{\boldsymbol{\nu}}) = -\mathbf{k}_l \circ \dot{\boldsymbol{\nu}} - \mathbf{k}_{nl} \circ \dot{\boldsymbol{\nu}} \circ |\dot{\boldsymbol{\nu}}| \tag{21}$$

where $\circ$ denotes element-wise multiplication, $\mathbf{k}_l$ contains linear damping coefficients and $\mathbf{k}_{nl}$ quadratic damping coefficients.

5.6 Morison wind and current forces

Wind and current forces are modelled using Morison's equation. For each flow type (wind/current) and incidence direction θ, drag coefficients $c_d(\theta)$ and effective areas $A(\theta)$ are pre-computed. The force is:

$$\mathbf{F}_{\mathrm{mor}}(\theta) = \mathbf{C}_D(\theta)\cdot u^2 \tag{22}$$

where u is the flow velocity and $\mathbf{C}_D(\theta)$ is a 6×1 coefficient vector containing both forces and moments (the latter computed from the lever arm between the COG and the centre of the projected area).

6 Mooring and Towing Lines

6.1 Governing equation

Considering that no external angular momentum is applied, Newton's equation expressed per unit of line length is used to model mooring and towing lines [1, 12]:

$$\rho_0 \frac{\partial^2 \mathbf{r}(t,s)}{\partial t^2} = \frac{\partial \mathbf{F}(t,s)}{\partial s} + \mathbf{f}(t,s)\left|\frac{\partial \mathbf{r}}{\partial s}\right| \tag{23}$$

where ρ_0 is the cable mass per unit of length, $s \in [0, L]$ is the arc-length parameter, L is the unstretched cable length, $\mathbf{r} : [t_0, \infty) \times [0, L] \to \mathbb{R}^3$ is the position, $\mathbf{F}$ is the internal forces vector and $\mathbf{f}$ is the external forces per unit of length vector.

The strain is defined as:

$$e(t, s) = \left| \frac{\partial \mathbf{r}}{\partial s} \right| - 1 \tag{24}$$

6.2 External forces

The external forces per unit of length are [1, 6]:

$$\mathbf{f} = \mathbf{f}_{hg} + \mathbf{f}_{dt} + \mathbf{f}_{dn} + \mathbf{f}_{mt} + \mathbf{f}_{mn} + \mathbf{f}_{gn} + \mathbf{f}_{gd} + \mathbf{f}_{fr} \tag{25}$$

where each term is defined as follows:

Force	Expression	Description
$\mathbf{f}_{hg}$	$\dfrac{\rho_0 - \rho_w A}{1 + e}\,\mathbf{g}$	Gravity and buoyancy
$\mathbf{f}_{dt}$	$-\frac{1}{2} C_{DT}\, d\, \rho_w \lvert\mathbf{v}_t\rvert \mathbf{v}_t$	Tangential drag
$\mathbf{f}_{dn}$	$-\frac{1}{2} C_{DN}\, d\, \rho_w \lvert\mathbf{v}_n\rvert \mathbf{v}_n$	Normal drag
$\mathbf{f}_{mt}$	$-C_{MT} \frac{\pi d^2}{4} \rho_w \mathbf{a}_t$	Tangential added mass
$\mathbf{f}_{mn}$	$-C_{MN} \frac{\pi d^2}{4} \rho_w \mathbf{a}_n$	Normal added mass
$\mathbf{f}_{gn}$	$\mathbf{z}\, \lvert\mathbf{f}_{hg}\rvert\, e^{-G_K d (r_z - z_f)/\lvert\mathbf{f}_{hg}\rvert}$	Ground normal (exponential spring)
$\mathbf{f}_{gd}$	$\lvert\mathbf{f}_{hg}\rvert\, G_D\, d\, \min(v_z, 0)^2\, \mathbf{z}$	Ground damping
$\mathbf{f}_{fr}$	See Section 7	Seabed friction

The tangent velocity is $\mathbf{v}_t = (\mathbf{v} \cdot \mathbf{t})\mathbf{t}$ where $\mathbf{t} = \mathbf{r}'/|\mathbf{r}'|$ is the unit tangent vector, and $\mathbf{v}_n = \mathbf{v} - \mathbf{v}_t$. The same decomposition applies to accelerations.

6.3 Internal forces and tension models

Neglecting bending and torsion, the internal force is:

$$\mathbf{F} = T \cdot \mathbf{t} \tag{26}$$

Three tension models are available:

6.3.1 Rayleigh damping model

$$T = EA\,(e + \beta \dot{e}) \tag{27}$$

where E is Young's modulus, A is the cross-sectional area, β is the internal damping coefficient, $e = |\mathbf{r}'| - 1$ is the strain and $\dot{e} = \mathbf{t} \cdot \dot{\mathbf{r}}'$ is its time derivative. A tension-only variant sets $T = \max\{0, T\}$.

6.3.2 Viscoelastic (Banks) model

For materials with hysteresis, a Boltzmann superposition constitutive law is used [2, 8]:

$$\frac{T(t)}{A} = \sigma(t) = g_e(\varepsilon(t)) + \int_0^t \dot{Y}(t-s)\, g_v(\varepsilon(s), \dot{\varepsilon}(s))\, ds + Y(0)\, g_v(\varepsilon(t), \dot{\varepsilon}(t)) - Y(t)\, g_{vi}(\varepsilon(0)) + \sum_{k=0}^{M} Y(t-t_k)(-1)^k\, [g_{vi}(\varepsilon \tag{28}$$

where $g_e(\varepsilon) = \sum_{i=1}^{3} E_i \varepsilon^i$ is the elastic response, g_{vi} and g_{vd} are third-degree polynomial functions for loading and unloading respectively, $Y(t) = e^{-Ct}$ is the exponential relaxation kernel, and $\{t_k\}$ are the turning points where $\dot{\varepsilon} = 0$.

6.3.3 Tabulated stress–strain curves

For arbitrary materials, stress–strain data can be provided in tabular form and interpolated during the simulation.

6.4 Weak formulation

The boundary conditions are Dirichlet (prescribed positions at both line ends). Let $V = \{w \in H^1(0, L) \mid w(0) = w(L) = 0\}$ be the test function space. Multiplying (23) by $w \in V$ and integrating by parts:

$$\int_0^L \rho_0 \ddot{\mathbf{r}}\, w\, ds = -\int_0^L \mathbf{F}\, w'\, ds + \int_0^L \mathbf{f} \left| \frac{\partial \mathbf{r}}{\partial s} \right| w\, ds \tag{29}$$

6.5 SEM spatial discretisation

The domain $[0, L]$ is divided into $n-1$ elements of equal length $h = L/(n-1)$. Within each element, $p+1$ Gauss–Lobatto–Lagrange (GLL) nodes are placed, yielding $N = p(n-1)+1$ global nodes. The GLL nodes $\{\xi_i\}_{i=0}^{p}$ in $[-1, 1]$ include the endpoints $\xi_0 = -1$ and $\xi_p = 1$, ensuring inter-element continuity.

The basis functions $\{\phi_i\}_{i=0}^{N-1}$ are the Lagrange interpolation polynomials over the GLL nodes, satisfying $\phi_i(s_j) = \delta_{ij}$. They are classified as:

- **Vertex functions**: non-zero in two adjacent elements (at shared boundary nodes).
- **Bubble functions**: non-zero in a single element only (at interior nodes).

The Galerkin approximation writes $\mathbf{r}(s,t) = \sum_{i=0}^{N-1} \mathbf{r}_i(t)\phi_i(s)$, and substituting into (29) using $w = \phi_j$ yields the ODE system:

$$\rho_0 \mathbf{M}\,\ddot{\mathbf{r}} = -\mathbf{K}^*\mathbf{F} + \mathbf{M}\,\hat{\mathbf{f}} + \boldsymbol{\Gamma} \tag{30}$$

where:

$$\mathbf{M}_{ij} = \int_0^L \phi_i\phi_j\,ds, \qquad \mathbf{K}^*_{ij} = \int_0^L \phi_i'\phi_j\,ds \tag{31}$$

and $\boldsymbol{\Gamma}$ accounts for the non-zero boundary term from integration by parts at the endpoints.

6.6 Derivative operator

Spatial derivatives at the nodes are computed via the global derivative matrix $\mathbf{D}_n$:

$$\begin{pmatrix} \mathbf{r}^{(n)}(s_0) \\ \vdots \\ \mathbf{r}^{(n)}(s_{N-1}) \end{pmatrix} = \mathbf{D}_n \begin{pmatrix} \mathbf{r}_0 \\ \vdots \\ \mathbf{r}_{N-1} \end{pmatrix} \tag{32}$$

At nodes shared between elements (where the basis function derivatives are discontinuous), the average of the left and right derivatives is used. The matrix $\mathbf{D}_n$ is assembled from the local derivative matrix $\mathbf{D}_n^L$ (computed from the GLL polynomial derivatives) using the same element-assembly procedure as for $\mathbf{M}$ and $\mathbf{K}^*$, with an additional division by two at shared nodes.

The system matrices are constant in time and computed only once (unless the line length changes due to a winch, in which case they scale with L).

6.7 Added mass iteration

Since the external force $\hat{\mathbf{f}}$ depends on the acceleration $\ddot{\mathbf{r}}$ through the added mass term, the system (30) is solved iteratively:

1. Set $\mathbf{a}^0$ as the acceleration from the previous time step.
2. Solve $\mathbf{M} \cdot \mathbf{a}^{k+1} = -\mathbf{K}^*\mathbf{F} + \mathbf{M} \cdot \hat{\mathbf{f}}(\mathbf{a}^k) + \boldsymbol{\Gamma}$.
3. Repeat until $\|\mathbf{a}^{k+1} - \mathbf{a}^k\| < \text{tol}$ or $k > k_{\max}$.

6.8 Boundary conditions

6.8.1 Anchor, fairlead or actuator

When the position of a line end is prescribed (fixed anchor, body fairlead, or lab actuator), the corresponding row of the mass matrix is replaced by an identity and the corresponding entry of the force vector by the prescribed acceleration.

6.8.2 Multi-line joint

When n lines converge at a common point, the joint acceleration is computed as the weighted average:

$$\mathbf{a}_J = \frac{\sum_{i=1}^{n} (\rho_0)_i \, \hat{\mathbf{a}}_i}{\sum_{i=1}^{n} (\rho_0)_i} \tag{33}$$

Before computing forces, the positions and velocities of all converging nodes are averaged and imposed.

6.8.3 Winch

A winch modifies the line length $L = L_0 + \theta R$ where θ is the winch rotation angle and R the winch radius. The winch rotation is governed by:

$$I \, \alpha = T \cdot R - M_{\text{engine}} \tag{34}$$

where $I = \frac{1}{2} m R^2$ is the moment of inertia, T is the line tension at the winch end, and M_{engine} is the engine torque. This ODE is integrated together with the line ODE. When L changes, the system matrices are updated by scaling: $\mathbf{D}_n \to (L_0/L)^n \mathbf{D}_n$ and $\mathbf{M} \to (L/L_0)\mathbf{M}$.

6.8.4 Elastic anchor

An elastic anchor provides an exponential restoring force:

$$\mathbf{F}_{\text{anchor}} = -c \left(1 - e^{-k \|\mathbf{r} - \mathbf{r}_{\text{ref}}\|}\right) \frac{\mathbf{r} - \mathbf{r}_{\text{ref}}}{\|\mathbf{r} - \mathbf{r}_{\text{ref}}\|} \tag{35}$$

where c is the ultimate holding capacity and k is a rate parameter.

6.9 Quasi-static initial conditions

The initial line configuration is computed using the catenary equations. Given the horizontal and vertical spans (x_F, z_F), the fairlead tensions (H_F, V_F) are found by solving a 2×2 nonlinear system with Newton–Raphson. Two cases are distinguished depending on whether part of the line rests on the seabed ($L_B = L - V_F/\omega > 0$) or not, where $\omega = (\rho_0 - \rho_w A) g$ is the submerged weight per unit length. The full catenary equations and their Jacobian are given in [7]. For configurations where the catenary solver does not converge (e.g. crowfoot junctions), a straight-line initial condition is used and a transient simulation finds the equilibrium.

7 Seafloor Interaction

The seafloor interaction model computes the projection of mooring line nodes onto the seabed surface to evaluate ground normal and friction forces [6]. Three seabed types are supported.

7.1 Flat seafloor

For a horizontal seabed at depth z_g, the projection direction is $\mathbf{d}_{P,i} = \mathbf{e}_3$ and the penetration depth is $d_{G,i} = r_{z,i} - z_g$.

7.2 Inclined plane

For a planar seabed defined by its unit normal $\mathbf{n}$ and a point on the plane, the projection direction is $\mathbf{d}_{P,i} = \mathbf{n}$ and the penetration depth is $d_{G,i} = (\mathbf{r}_i - \mathbf{r}'_i) \cdot \mathbf{n}$, where $\mathbf{r}'_i = \mathbf{r}_i + \mu_i \mathbf{n}$ is the projected point with $\mu_i = (k - \mathbf{r}_i \cdot \mathbf{n})$.

7.3 Complex bathymetry

For irregular seabeds described by a triangulation, the continuous projection algorithm of Orazi and Reggiani [11] is used:

1. The seabed is described by a triangulation with vertices $\{V_j\}$ and vertex normals computed as the average of the normals of adjacent triangles.
2. For each line node $\mathbf{r}_i$, a parallel triangle T_{point} is constructed using the vertex normals, and barycentric coordinates (s, t) are computed to determine which seabed triangle contains the projected point.
3. A change-of-frame matrix G is pre-computed for each triangle to lay it into the XY plane, simplifying the penetration depth calculation: $d_{G,i} = (G\,\mathbf{r}_i)_z$.
4. The projection direction $\mathbf{d}_{P,i}$ is obtained from the vertex-normal interpolation, providing a continuous variation across element boundaries.

7.4 Ground normal and friction forces

The ground normal force uses the exponential spring model with a smoothing coefficient α to avoid numerical discontinuities:

$$\alpha = \text{sm}(d_{G,i},\ -d/2,\ 0,\ 0,\ 1) \tag{36}$$

where sm is a smoothed step function based on a cubic polynomial. The friction force follows a Coulomb model:

$$\mathbf{f}_{fr} = \begin{cases} -C_k \|\mathbf{f}_n\| \dfrac{\mathbf{v}_\pi}{v_c} & \text{if } \|\mathbf{v}_\pi\| < v_c \\ -C_k \|\mathbf{f}_n\| \dfrac{\mathbf{v}_\pi}{\|\mathbf{v}_\pi\|} & \text{otherwise} \end{cases} \tag{37}$$

where C_k is the kinetic friction coefficient, v_c is a velocity threshold, $\mathbf{f}_n$ is the normal force, and $\mathbf{v}_\pi = \mathbf{v} - v_n \mathbf{d}_{P,i}$ is the velocity in the seabed plane.

8 Springs and Connectors

The spring model provides a versatile 6-DOF connector between two boundary connection points (BCPs). It can represent elastic connectors, contact surfaces, fenders, or piloted joints.

8.1 Deformation measurement

Each spring connects two points, each with its own local coordinate system defined by three orthogonal unit vectors $(\mathbf{x}, \mathbf{y}, \mathbf{z})$. The translational deformation is computed by projecting the vector $\mathbf{L}_{12}$ connecting both points onto the spring reference frame:

$$\begin{pmatrix} u_x \\ u_y \\ u_z \end{pmatrix} = \begin{pmatrix} \mathbf{L}_{12} \cdot \mathbf{x} \\ \mathbf{L}_{12} \cdot \mathbf{y} \\ \mathbf{L}_{12} \cdot \mathbf{z} \end{pmatrix} \tag{38}$$

The rotational deformation is obtained from the relative rotation between the two local frames:

$$\begin{pmatrix} u_\alpha \\ u_\beta \\ u_\gamma \end{pmatrix} = \begin{pmatrix} \operatorname{atan2}(\mathbf{y}_1 \cdot \mathbf{z}_2,\ \mathbf{z}_1 \cdot \mathbf{z}_2) \\ -\arcsin(\mathbf{x}_1 \cdot \mathbf{z}_2) \\ \operatorname{atan2}(\mathbf{x}_1 \cdot \mathbf{y}_2,\ \mathbf{x}_1 \cdot \mathbf{x}_2) \end{pmatrix} \tag{39}$$

8.2 Spring forces

The total spring force in the local frame is the sum of three contributions:

Elastic term:

$$\mathbf{F}_K = \mathbf{K} \cdot \mathbf{u} \quad \text{or} \quad \mathbf{F}_K = \mathbf{f}_K(\mathbf{u}) \quad \text{(from tabulated curves)} \tag{40}$$

Damping term:

$$\mathbf{F}_D = -\mathbf{D} \circ \|\mathbf{F}_K\| \circ \dot{\mathbf{u}} \tag{41}$$

Friction term (dynamic Coulomb friction with static/dynamic transition):

$$\mathbf{F}_F = -\left(\mathbf{M}_\mu \cdot \|\mathbf{F}_K\|\right) \circ \frac{\dot{\mathbf{u}}}{\|\dot{\mathbf{u}}\|}, \qquad \mathbf{M}_\mu = \mathbf{M}_{\mu,\text{dyn}} + \mathbf{M}_{\mu,\text{stat}} \cdot e^{-\ln(100)\,|\dot{\mathbf{u}}|/u_{100}} \tag{42}$$

where u_{100} is the velocity threshold below which the friction coefficient transitions from static to dynamic.

9 Winches

9.1 Mechanical model

A winch is a rotating drum with inertia I, radius R and drag coefficient. The rotation is governed by Newton's equation for rotations:

$$I\,\alpha = T \cdot R - M_{\text{engine}} \tag{43}$$

The winch state (θ, ω) is integrated as part of the global ODE system, and the line length is updated as $L = L_0 + \theta R$.

9.2 Controllers

Two controller types are available:

- **Constant tension**: a PID controller adjusts the engine torque to maintain a target tension at each winch.
- **Horizontal dynamic positioning**: a state-space reference model with lead-lag and integral gains computes desired forces, which are converted to winch tensions via an inversor block.

10 Oscillating Water Columns

10.1 Pneumatic chamber model

The OWC consists of a chamber body and a floater body. The relative heave displacement δz between the two bodies changes the air volume:

$$V_{\text{air}}(t) = V_{\text{ref}} + A_w\,\delta z(t) \tag{44}$$

where V_{ref} is the reference air volume and A_w is the waterplane area.

The chamber pressure evolves according to the adiabatic gas law. The pressure rate of change is:

$$\dot{p}_{\text{rel}} = f\left(p_{\text{air}},\ \dot{m},\ m_{\text{air}},\ \dot{V}_{\text{air}},\ V_{\text{air}}\right) \tag{45}$$

where $\dot{m}$ is the mass flow rate through the turbine or orifice, computed from the pressure difference and stagnation density.

10.2 Turbine model

The turbine operates according to dimensionless performance (buckling) curves that relate adimensional pressure, power, and mass flow rate. The angular velocity of the turbine rotor is governed by:

$$I_t \, \dot{\omega} = Q_{\text{air}} - Q_{\text{gen}} \tag{46}$$

where I_t is the rotor inertia, Q_{air} is the aerodynamic torque from the air flow, and Q_{gen} is the generator torque. Valve models (bypass, throttle) provide additional flow control.

11 Wind Turbines

OASIS couples to OpenFAST [10] for aero-servo-elastic simulation of floating offshore wind turbines. The coupling transfers:

- Platform 6-DOF kinematics (position, velocity, acceleration) from OASIS to OpenFAST.
- Aerodynamic and tower loads from OpenFAST to the body COG in OASIS as a 6×1 force vector.

The rotor state (position θ_r, speed ω_r, acceleration α_r) is integrated as part of the global ODE system. The rotor inertia is added to the body mass matrix. Yaw control modes (free, fixed, controlled) are supported.

12 Sinking

The sinking module models progressive flooding of a body's compartments. Each compartment group has a time-dependent filling schedule. As water enters the compartments:

1. The body mass, COG and inertia matrix are updated based on the water mass and its distribution within the compartment geometry.
2. Multiple hydrodynamic databases (pre-computed at different draft levels) are interpolated based on the total filling mass, updating the hydrostatic stiffness, added mass, damping and excitation coefficients.

This allows the simulation to capture dynamic effects during ballasting operations, including trim and heel variations.

13 Time Integration

The global ODE system has the form:

$$\frac{d\mathbf{u}}{dt} = f(\mathbf{u}, t), \qquad \mathbf{u}(0) = \mathbf{u}_0 \tag{47}$$

Two families of implicit solvers are available.

13.1 Adaptive BDF2

The second-order Backward Differentiation Formula [4] with variable time step is:

$$y_{n+2} - \frac{(1+w)^2}{1+2w}\, y_{n+1} + \frac{w^2}{1+2w}\, y_n = h_{n+2} \frac{1+w}{1+2w}\, f_{n+2} \tag{48}$$

where $w = h_{n+2}/h_{n+1}$ is the step ratio. The first step uses backward Euler (BDF1) for initialisation.

Newton–Raphson iteration: At each time step, the nonlinear system $F(y_{n+2}) = 0$ is solved iteratively:

$$\mathbf{J}_F(y^{(i)}) \cdot \delta y^{(i)} = -F(y^{(i)}), \qquad y^{(i+1)} = y^{(i)} + \delta y^{(i)} \tag{49}$$

where $\mathbf{J}_F$ is the Jacobian matrix, computed via finite differences with increment $\delta = 10^{-8}$. The Jacobian is recycled for multiple time steps to reduce computational cost.

Adaptive time stepping: The local truncation error (LTE) is estimated from the third derivative of the solution:

$$\text{LTE} \approx \frac{h_{n+2}^2 (h_{n+1} + h_{n+2})}{6}\, y_{n+2}^{(3)} \tag{50}$$

The new step size is $h_{n+3} = \sigma \cdot h_{n+2}$ where:

$$\sigma = \left(\sqrt{n} \cdot \left| \frac{\text{EWT}}{\text{LTE}} \right| \right)^{1/3}, \qquad \text{EWT} = r_{\text{tol}} \|\mathbf{y}\| + a_{\text{tol}} \tag{51}$$

and n is the system size, r_{tol} and a_{tol} are relative and absolute tolerances.

13.2 ESDIRK

The Explicit first-stage Singly Diagonal Implicit Runge–Kutta method [3] provides fourth-order accuracy with L-stability. An s-stage scheme is defined by the Butcher tableau coefficients $(\mathbf{a}, \mathbf{b}, \boldsymbol{\beta})$. An embedded lower-order solution provides error estimation for adaptive stepping. Multiple smoothing strategies for the step-size multiplier are available (min-max clamping, arctangent smoothing).

14 Coupling

14.1 Global state vector

All subsystem degrees of freedom are packed into a single state vector:

$$\mathbf{u} = \begin{pmatrix} \mathbf{r}_{\text{bodies}} \\ \mathbf{r}_{\text{lines}} \\ \theta_{\text{winches}} \\ \theta_{\text{turbines}} \\ \mathbf{v}_{\text{bodies}} \\ \mathbf{v}_{\text{lines}} \\ \omega_{\text{winches}} \\ \omega_{\text{turbines}} \\ p_{\text{OWC}} \\ \omega_{\text{OWC turb.}} \end{pmatrix} \tag{52}$$

This vector is passed to the ODE solver, which calls the right-hand side (RHS) function at each evaluation.

14.2 RHS evaluation order

The RHS function (`CalculateSystemDynamics`) follows a weak coupling algorithm:

1. **Unpack** state vector → body positions/velocities, line nodes, winch angles, rotor states, OWC pressures.
2. **Update locked bodies** (prescribed motions).
3. **Update BCPs**: body BCPs get kinematics from body state; line boundary conditions set from BCP values.
4. **Set joint/elastic-anchor BCs**: average positions from converging lines.
5. **Compute line forces**: internal + external forces via `SEM_computeF()` for each line.
6. **Compute spring forces**: `computeSpringForces()` for each spring.
7. **Update sinking hydrostatics** if applicable.
8. **Interpolate hydrodynamic forces**: Lagrange interpolation between last computed hydro time stamps.
9. **Accumulate BCP forces on bodies**: sum mooring/spring reactions at COG.
10. **Add wind turbine and OWC forces** on bodies.
11. **Solve body accelerations**: $\ddot{\boldsymbol{\nu}} = (\mathbf{M} + \mathbf{A})^{-1} \cdot \mathbf{F}_{\text{total}}$.

12. **Solve line accelerations**: assemble coupling matrix, solve $\mathbf{M}_{\text{lines}} \cdot \mathbf{a} = \mathbf{F}_{\text{lines}}$.
13. **Compute winch and turbine accelerations**.
14. **Compute OWC pressure derivatives**.
15. **Pack** derivatives into $\dot{\mathbf{u}}$.

14.3 Multi-rate coupling

Different subsystems are updated at different rates:

- **Hydrodynamic forces**: recomputed at intervals of `hydroTimeStep`; interpolated in between using 3-point Lagrange interpolation.
- **Wind turbine (OpenFAST)**: updated at `fastTimeStep`.
- **Winch controllers**: updated at `winchesContTimeStep`.

This multi-rate approach significantly reduces computational cost while maintaining accuracy for the slower-varying force components.

References

[1] O.M. Aamo and T.I. Fossen. Finite element modelling of mooring lines. *Mathematics and Computers in Simulation*, 53(4-6):415–422, 2000.

[2] H.T. Banks, G.A. Pintér, L.K. Potter, M.J. Gaitens, and L.C. Yanyo. Modeling of nonlinear hysteresis in elastomers under uniaxial tension. *Journal of Intelligent Material Systems and Structures*, 10(2):116–134, 1999.

[3] M.H. Carpenter and C.A. Kennedy. Fourth-order 2N-storage Runge–Kutta schemes. NASA Technical Memorandum, 2005.

[4] E.A. Celaya, J.J.A. Aguirrezabala, and P. Chatzipantelidis. Implementation of an adaptive BDF2 formula and comparison with the MATLAB ode15s. *Procedia Computer Science*, 29:1014–1026, 2014.

[5] W.E. Cummins. The impulse response function and ship motions. *Schiffstechnik*, 9:101–109, 1962.

[6] P. Desiré Valdor, A. Rodríguez-Luis, and R. Guanche. Simulation of mooring lines in complex bathymetries using a finite element method. *Ocean Engineering*, 2022.

[7] O.M. Faltinsen. *Sea Loads on Ships and Offshore Structures*. Cambridge University Press, 1990.

[8] M. González Alonso. Simulation of elasto-plasticity in mooring lines with finite elements. Bachelor's thesis, Universidad de Cantabria, 2025.

[9] G.E. Karniadakis and S.J. Sherwin. *Spectral/hp Element Methods for Computational Fluid Dynamics*. Oxford University Press, 2nd edition, 2005.

[10] NREL. OpenFAST documentation. `https://openfast.readthedocs.io/`, 2024.

[11] L. Orazi and B. Reggiani. A novel algorithm for a continuous and fast 3D projection of points on triangulated surfaces. *Journal of King Saud University – Computer and Information Sciences*, 34(4), 2020.

[12] A. Rodríguez-Luis, J.A. Armesto, R. Guanche, C. Barrera, and C. Vidal. Simulation of marine towing cable dynamics using a finite elements method. *Journal of Marine Science and Engineering*, 8(2):140, 2020.