TU Delft

Mitigating Offshore Monopile Installation Downtime through Rigging Optimization

A Parametric Study and Optimization Framework to Address Resonance-Induced Operational Constraints

Supervisors - TU Delft
P. Naaijen
P. Van Der Male
Supervisors - Boskalis
R. Beindorff
T. Volker

Olivier Lagas - 4595114

October 6, 2025

Abstract

Offshore monopile installation campaigns frequently face delays due to environmental loads during dynamic lifting operations. This problem is especially acute in the lift-off phase, when the monopile is suspended and the vessel is exposed to wave-induced motion. The goal of this thesis is to investigate whether optimization of the rigging geometry can reduce such downtime by mitigating dynamic amplification effects.

A planar, linearized Lagrangian model was developed to simulate the dynamic response of the rigging system suspended from a vessel-mounted crane. The model includes multiple pendulum bodies and uses a frequency-domain formulation to calculate the transfer function from crane tip motion to resulting sideloads. This transfer function is used to compute the most probable maximum (MPM) sideloads across a range of sea states. A grid-based workability analysis links these loads to operational thresholds, and a numerical optimization was implemented to tune the rigging geometry for maximum uptime.

The analysis identified a dominant resonance near $T_p = 8.16$s — the second natural mode of the suspended system — as the primary driver of limit exceedance. By modifying the lengths of the rigging elements, the natural frequencies could be shifted away from this critical range. The optimized configuration improved workability by 10–20 percentage points, depending on the metocean conditions.

Although the simplified model underestimates absolute load levels compared to a detailed OrcaFlex simulation, the relative response behavior is well captured. A conservative threshold correction was applied to enable consistent limit checking. The study shows that dynamic rigging optimization is a viable, low-effort strategy for reducing downtime in offshore monopile installation, and should be considered during project engineering.

Contents

List of Figures

List of Tables

1 Introduction

The rapid expansion of offshore wind energy is reshaping the future of sustainable electricity generation in Europe and beyond. As countries work to meet increasingly ambitious climate targets, the role of offshore wind farms is growing in both scale and strategic importance. This transition presents a wide range of engineering challenges, particularly in the installation phase of wind turbine foundations. This thesis explores one of these challenges — the operational bottlenecks during monopile installation — and proposes a method to mitigate associated downtime through dynamic modeling and rigging optimization.

1.1 Background of Offshore Wind Energy

The transition towards a low-carbon energy system is one of the major engineering and policy challenges of the coming decades. Offshore wind energy is expected to play a pivotal role in this transition, offering large-scale renewable electricity generation in regions with limited land availability. Offshore wind farms benefit from higher and more consistent wind speeds compared to onshore locations, as well as reduced spatial and visual constraints (IEA, 2019).

The Netherlands is well positioned to accelerate offshore wind deployment, thanks to its favourable North Sea conditions, strong maritime industry, and established offshore infrastructure. According to the Dutch national roadmap, the government aims to achieve 21 GW of installed offshore wind capacity by 2030 (Noordzeeloket, 2020). This ambition aligns with broader European climate targets and is supported by an increasingly mature and cost-competitive offshore wind market (WindEurope, 2020).

At a global level, technological progress has led to significant cost reductions and scaling-up of offshore wind systems. Turbine capacities now exceed 15 MW, and developments are pushing towards rotor diameters of 250 meters (IEA, 2019; World Forum Offshore Wind, 2024) as shown in Figure 1. These trends open the door to large-scale offshore energy hubs, but also introduce new engineering challenges related to foundation design, logistics, and marine operations.

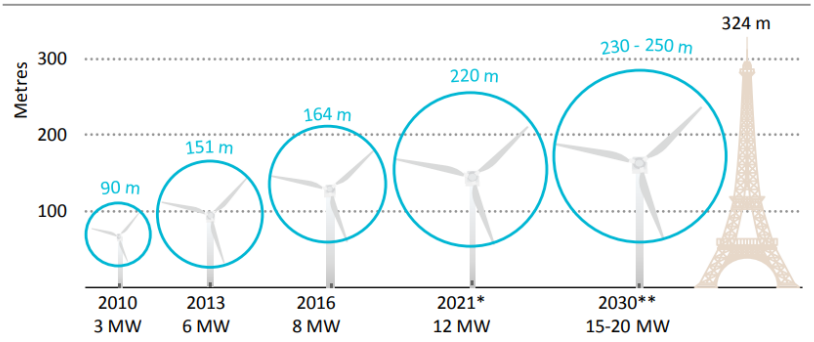


Figure 1: Evolution of offshore wind turbine capacity over time, including forecasted trends. Further technology improvements through to 2030 could see bigger turbines sizes of 15-20 MW. Adapted from (IEA, 2019).

Monopile foundations remain the dominant support structure for fixed-bottom offshore wind turbines, particularly in water depths up to 50 meters (IEA, 2019). As turbine size and hub heights continue to grow, the requirements for safe and efficient installation processes become more demanding. As a result, there is growing interest in optimizing these operations to reduce downtime and improve workability.

1.2 Monopile Installation and Operational Challenges

Monopile installation offshore is a highly coordinated operation involving transport, lifting, upending, and driving of large steel foundations into the seabed. Heavy lift vessels (HLVs), such as the Bokalift 2, play a central role in this process by handling monopiles that can exceed 100 meters in length and weigh upwards of 1,500 tonnes (DNV, 2024). Once lifted from a barge or transport deck, the pile is typically upended into a vertical position and lowered into a gripper or template before being driven using a hydraulic hammer.

Each step in this sequence must be executed within strict operational limits, many of which are governed by environmental conditions. Even moderate wave heights can induce vessel motions that disrupt the relative alignment between the crane tip and the suspended pile. These dynamic interactions are especially

critical during the lift-off and vertical stabbing phases, where excessive sideloads or uncontrolled contact can compromise both safety and equipment integrity (WindEurope, 2020).

The sensitivity of these operations to sea state conditions results in narrow weather windows — particularly during the North Sea winter and shoulder seasons — and contributes to frequent operational delays. Crane lifts are generally suspended when motion or load thresholds are exceeded, leading to costly standby periods. This challenge has only grown with the trend toward larger turbines and deeper sites, which demand longer and heavier monopiles and stretch the capacity limits of existing equipment (IEA, 2019).

As a result, improving the workability of offshore installation operations has become a priority. Even modest gains in allowable wave conditions can yield significant reductions in project risk and cost. This motivates ongoing efforts in dynamic modeling, motion prediction, and rigging optimization to minimize downtime and expand safe operational windows.

1.3 Motivation and Relevance for Boskalis

During discussions with the Marine Engineering team at Boskalis, the issue of pendulum-like motion during monopile lift-off was identified as a recurring operational bottleneck. This problem was particularly evident during the installation of the Revolution Wind Farm off the east coast of the United States in 2024. At this site, dynamic loading in the rigging — especially during the transfer and initial lift-off from the feeder barge as shown in Figure 2 — frequently approached or exceeded allowable limits, leading to unplanned operational delays.

Figure 2: Feedering operation during monopile installation: the Bokalift 2 lifts monopiles from a feeder barge.

A contributing factor to this bottleneck lies in the regulatory environment of U.S. offshore wind projects. Under the Jones Act, offshore transport between U.S. ports must be carried out by vessels that are U.S.-built, U.S.-owned, and U.S.-crewed (U.S. Customs and Border Protection, 2024). Since European heavy lift vessels (HLVs) like the Bokalift 2 do not meet these requirements, a feedering approach is typically used. In this process, monopiles are transported to the offshore site using Jones Act–compliant feeder barges or from Canadian ports, after which the HLV lifts them onboard for installation (McLean, 2023).

While feedering provides a viable legal workaround, it introduces operational complexity. The relative motion between the HLV and feeder barge is highly sensitive to wave conditions, especially during the lift-off phase. This motion not only amplifies side loads in the rigging, but also narrows the weather window for safe operations — contributing to increased downtime and elevated project costs.

From Boskalis' perspective, improving the workability of HLV-based monopile installation under these constraints is of both technical and economic relevance. Small changes in rigging configuration or dynamic behavior that reduce peak loads can lead to meaningful gains in project efficiency. Reducing unnecessary dynamic amplification helps avoid operational interruptions, preserves equipment longevity, and expands the operable sea state envelope — all of which support more competitive and predictable offshore construction planning.

1.4 Research Objectives and Questions

This thesis investigates how the configuration of a crane rigging system can be optimized to reduce weather-related downtime during offshore monopile installation. The focus is placed on the lift-off phase, where the suspended monopile exhibits pendulum-like motion that may lead to sideload exceedances and halted operations.

The study is motivated by operational challenges encountered aboard the Bokalift 2 vessel, where resonance-induced amplifications were observed to cause frequent constraint violations. These events highlight the need for improved understanding of the system's dynamic response — and whether its configuration can be tuned to improve uptime.

The central research question guiding this work is:

> *How can the rigging configuration of an offshore heavy lift vessel be optimized to reduce downtime during offshore monopile lift-off operations?*

To answer this question, the research addresses the following sub-questions:

1. Which operational constraint contributes most to weather-related downtime, and is suitable for optimization?
2. Can a simplified dynamic model accurately capture the behavior of the crane–rigging–monopile system?
3. Can this model be used to optimize the rigging geometry for improved workability?
4. Does this optimization result in meaningful operational improvement?

By combining dynamic modeling, workability evaluation, and parameter optimization, this thesis aims to demonstrate that even minor changes to the rigging setup can significantly improve offshore installation efficiency — with minimal engineering or hardware effort.

1.5 Thesis Outline

The structure of this thesis reflects the step-by-step approach taken to analyze, model, and optimize the dynamic rigging system used during offshore monopile installation. Following the introduction, the report begins with a detailed analysis of operational constraints based on simulation data, identifying the key bottlenecks contributing to downtime (**Section 2: Problem Analysis**). This forms the foundation for the theoretical background, which reviews the relevant dynamics and workability concepts needed to build a suitable model (**Section 3: Theoretical Background**).

The methodology chapter presents the development of a simplified Lagrangian rigging model and its integration into a frequency-domain framework for sideload prediction (**Section 4: Model Description and Methodology**). The model is then validated against an OrcaFlex simulation to assess its accuracy and limitations (**Section 5: Model Validation**). The subsequent results chapter applies this model to evaluate operational workability and explores how rigging parameters can be tuned to reduce downtime (**Section 6: Results**).

Finally, the discussion and conclusion chapter reflects on the implications of the findings, addresses model limitations, and offers recommendations for practical implementation and future research (**Section 7: Discussion and Conclusion**).

Note on AI-assisted tools. In preparing this thesis, I made limited use of AI-based assistants to improve efficiency and clarity. Specifically, I used them (i) as a writing aide to refine phrasing and consistency, (ii) to accelerate literature discovery by surfacing potentially relevant sources for me to review, and (iii) as a coding assistant and debugger to propose implementations, explain error traces, and suggest fixes. All modelling choices, equations, parameter values, and interpretations are my own; all results and figures were produced from my verified scripts and, where applicable, cross-checked against OrcaFlex simulations. No text, data, or citations were included without my review, and any sources identified with AI assistance were consulted directly and are cited in the references.

2 Problem Analysis

Offshore installation campaigns are often hindered by weather-related downtime, especially during critical lifting operations. For the Bokalift 2 vessel, this downtime is governed by the exceedance of various operational limits related to motions, forces, and clearances. These conditions are systematically evaluated using so-called Unity Check (UC) values. The UC for a given parameter is calculated by normalizing the predicted operational load against its allowable threshold:

$$UC = \frac{\text{Predicted Load}}{\text{Allowable Limit}} \tag{1}$$

If this dimensionless ratio exceeds 1.0, the operational limit is considered breached, and the operation cannot proceed safely.

To better understand which constraints most critically impact operational uptime, an extensive analysis was performed on time-domain simulation data. This analysis identifies which UC values are most frequently violated, how they interact, and where targeted improvements could yield the largest reductions in downtime. The findings form the foundation for the modeling and optimization strategies presented in later chapters.

This chapter introduces the methodology used for analyzing UC exceedance patterns, presents the most critical bottlenecks, and discusses their relevance to the dynamic rigging model developed in this thesis.

2.1 Methodology

To systematically identify the main contributors to weather-related downtime, a structured analysis was performed on UC exceedance data. The methodology consisted of five main steps:

1. **Data Collection**: The analysis uses a dataset containing UC values for a wide range of operational parameters. A value above 1.0 indicates that the corresponding operational limit has been exceeded.
2. **Visualization of Exceedance Frequencies**: Bar charts were created to visualize how often each UC was exceeded. One chart presents all exceedances, while another filters out rarely exceeded limits and highlights those surpassed more than 100 times (see Figure 3).
3. **Simultaneous Exceedances**: A heatmap was constructed to visualize patterns of *simultaneous exceedances*, indicating which UC values tend to be exceeded together. This helps identify interdependent constraints that may shift when others are relieved (see Figure 4).
4. **Exceedance Magnitude Analysis**: To evaluate how severely the UC values are violated, exceedance values were binned and statistically assessed. This provides insight into how close most operations are to the threshold and whether small reductions could yield significant improvements.
5. **Load Case Attribution**: Finally, the data was segmented by operational load case to determine which scenarios are most prone to UC exceedance. This supports targeted mitigation strategies aligned with actual operational phases.

The UC dataset was generated through unweighted time-domain simulations provided by Boskalis, based on predefined environmental and operational conditions for the Revolution site.

2.2 Results

2.2.1 Exceedance Frequency

The filtered UC exceedance frequencies (Figure 3) narrow down the analysis to limits exceeded more than 100 times. This allows the focus to shift toward the most critical constraints affecting workability. Notably, the top contributors include **Crane Sideload UC**, both **SlingA** and **SlingB Tension UCs**, **MP Declination UC**, and **HTV Roll UC**:

- **Crane Sideload UC**: This UC was exceeded over 800 times, making it the most frequent contributor to downtime.
- **SlingA and SlingB Tension UC**: These UCs occupy the second and third places, respectively. The high frequency reflects how the load dynamically shifts between slings as the rigging oscillates during operation.

- **MP Declination UC**: Ranked fourth, this UC indicates that the monopile undergoes significant angular displacements, pointing to pronounced pendulum-like swinging under certain conditions.
- **HTV Roll UC**: In fifth place, this UC highlights roll motions of the Heavy Transport Vessel (HTV), particularly relevant in beam seas or energetic swell conditions.

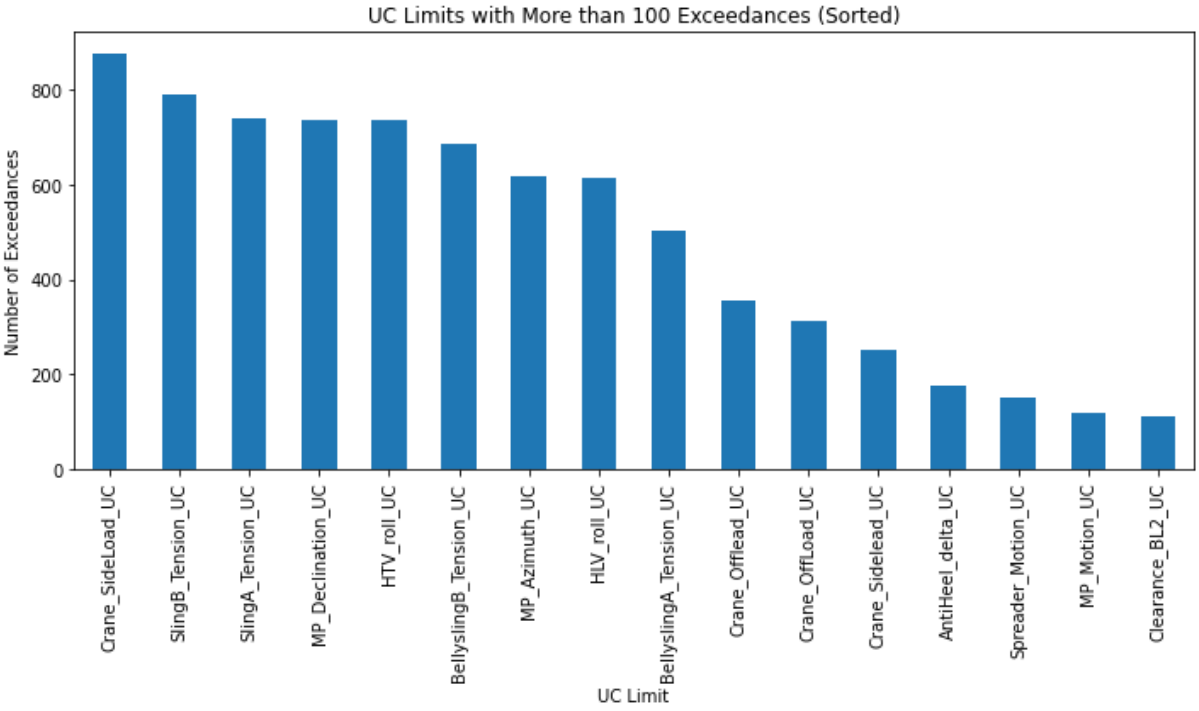


Figure 3: Exceedance Frequency of UC Limits (Filtered to Show > 100 Events)

2.2.2 Simultaneous Exceedances

The heatmap of simultaneous exceedances (Figure 4) reveals clear patterns of interdependence between various UC's. Several constraints are frequently violated together, highlighting the presence of coupled dynamic behavior in the system.

- Limits such as **HLV Roll UC** and **AntiHeel Delta UC** often co-exceed with others, indicating that they are part of broader system-wide motions. Improvements here may have limited standalone effect unless addressed in conjunction with related constraints.
- Of particular interest is the **Crane Sideload UC**, which frequently co-exceeds with motion and tension-related limits. This aligns with physical expectations: excessive side loading on the crane typically results from swinging of the suspended load, which simultaneously increases sling tensions and monopile orientation deviations.

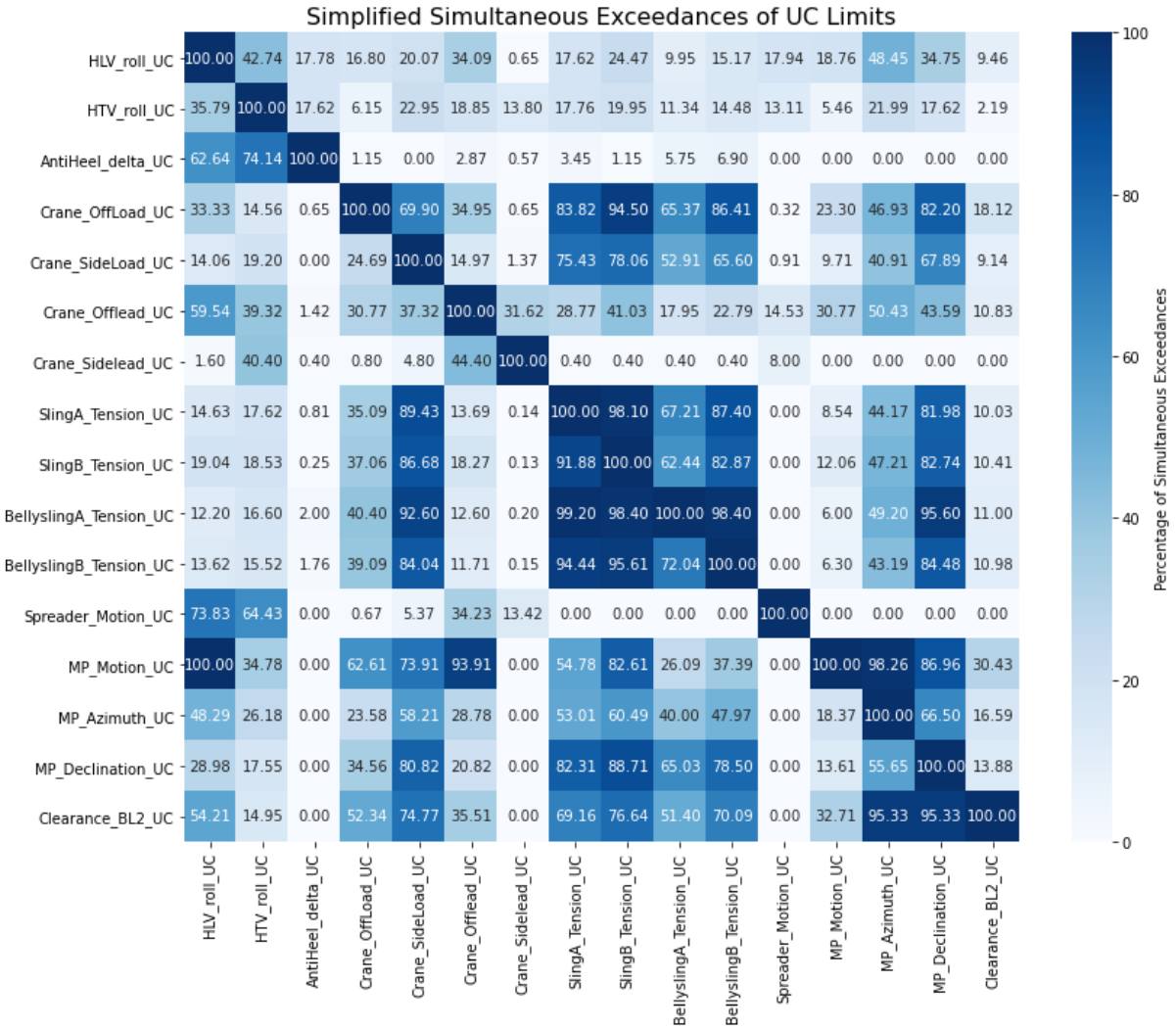


Figure 4: Simultaneous Exceedances of UC's

A more detailed co-occurrence analysis focused specifically on **Crane Sideload UC** is presented in the following subsection.

2.2.3 Simultaneous Exceedances with Crane Sideload UC

A focused co-occurrence analysis was conducted to identify which UC's are most commonly exceeded at the same time as the **Crane Sideload UC**. The results confirm a consistent dynamic pattern:

- **SlingA and SlingB Tension UCs** are exceeded in over 75% of the cases, reflecting the increased forces induced in the rigging system during swinging.
- **Bellysling Tension UCs** also show significant co-exceedance, particularly BellyslingB at 66%, further confirming that pendulum motion drives uneven load distribution across slings.
- **MP Declination UC** is exceeded in nearly 68% of the cases, suggesting large angular displacements of the monopile during side load exceedance events.
- Other motion-related constraints, including **MP Azimuth UC** and **Crane OffLoad UC**, also appear prominently.

These results confirm that the exceedances are not isolated incidents but stem from a common underlying cause: excessive dynamic motion of the suspended monopile. As expected, side load peaks coincide with increases in sling tension and pile inclination, reflecting the coupled nature of the system's response. This suggests that addressing the root cause — the pendulum-like swinging motion — could reduce multiple constraint violations simultaneously. By mitigating this dynamic behavior, not only the sideload UC but also related motion and tension-based UCs may fall below their thresholds, offering a compounding benefit in terms of operational workability.

UC	Percentage of Simultaneous Exceedances
HLV_roll_UC	14.35%
HTV_roll_UC	19.48%
AntiHeel_delta_UC	0.34%
Crane_OffLoad_UC	24.94%
Crane_Offlead_UC	15.26%
Crane_Sidelead_UC	1.71%
SlingA_Tension_UC	75.51%
SlingB_Tension_UC	78.13%
BellyslingA_Tension_UC	53.08%
BellyslingB_Tension_UC	65.72%
Spreader_Motion_UC	1.25%
MP_Motion_UC	10.02%
MP_Azimuth_UC	41.12%
MP_Declination_UC	67.99%
Clearance_BL2_UC	9.45%

Table 1: Percentage of Simultaneous Exceedances with Crane Sideload UC

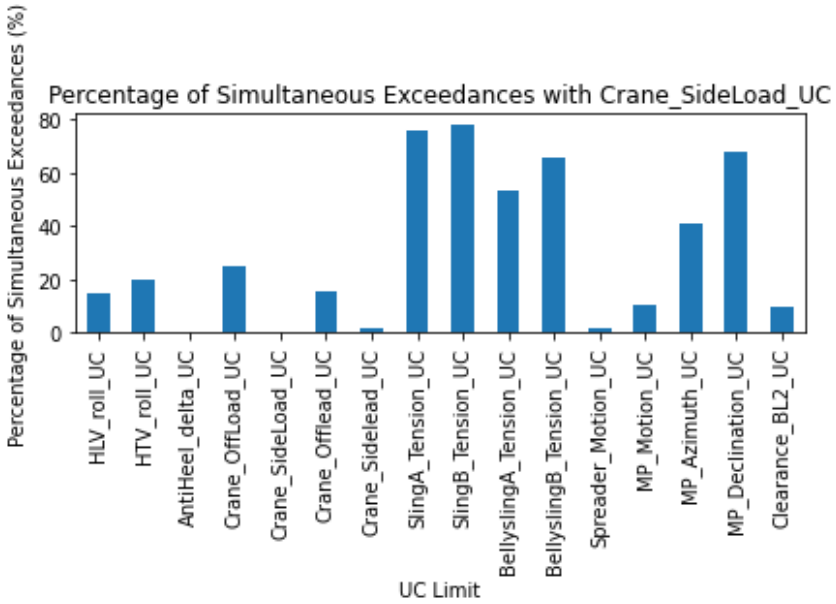


Figure 5: Percentage of Simultaneous Exceedances with Crane Sideload UC

2.2.4 Crane Sideload UC Exceedance Magnitude

Beyond frequency, it is valuable to understand *how much* the Crane Sideload UC is typically exceeded. Figure 6 shows the distribution of exceedances binned in 0.1 intervals from 1.0 to 3.0. The majority of exceedance cases are concentrated in the range between 1.0 and 1.5, with a steep drop-off beyond 1.6. The average exceedance value is 1.58. This implies that, on average, the sideload during exceedance events exceeds its allowable threshold by 58%.

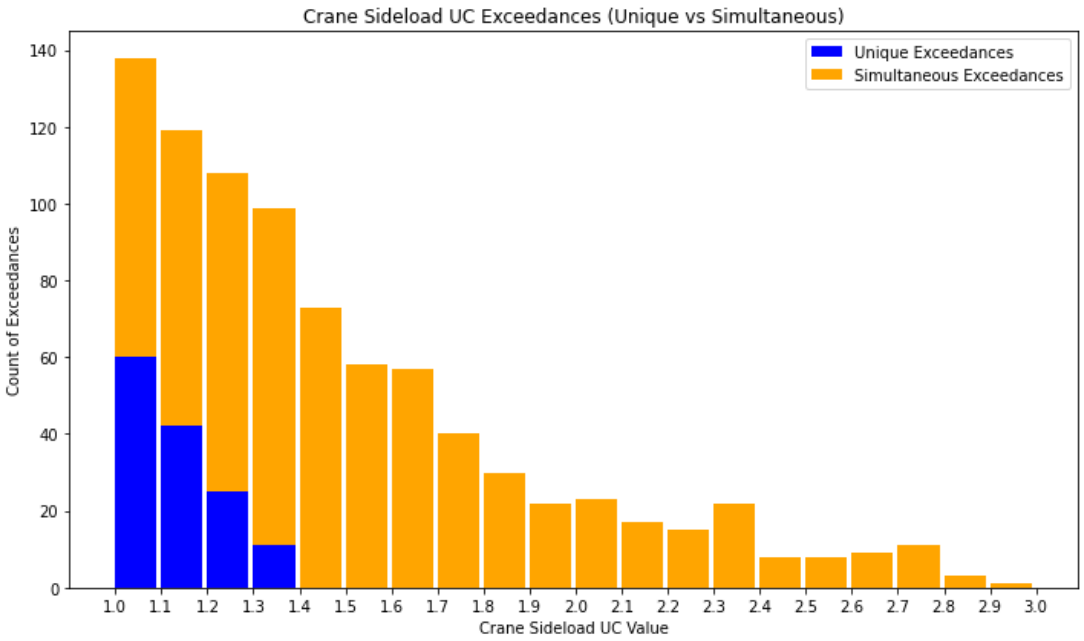


Figure 6: Distribution of Crane Sideload UC Exceedances by Magnitude

This concentration around low exceedance levels suggests that many events fall just above the allowable threshold. In practical terms, a modest reduction in dynamic sideload—through improved rigging layout or resonance avoidance—could shift a significant portion of these cases back within acceptable limits.

The figure also supports the earlier conclusion that exceedances are not isolated events: they arise in coupled motion states involving swinging, tension peaks, and pile orientation. While this is not the focus of this specific plot, the dominance of simultaneous exceedances (orange bars) is visible and expected.

In conclusion, the narrow margin by which most sideload exceedances occur reinforces the potential impact of even small optimizations. A reduction of 10–15% in peak side loading could eliminate a large number of downtime-driving violations.

2.2.5 Load Case Contributions and Exceedance Levels

To understand when sideload exceedances are most critical, we analyzed their distribution across individual load cases. Figure 7 shows that Load Case LC3b is responsible for 37% of all Crane Sideload UC exceedances, followed closely by LC3a at 32%. Together, these two load cases account for nearly 70% of all sideload-related downtime events.

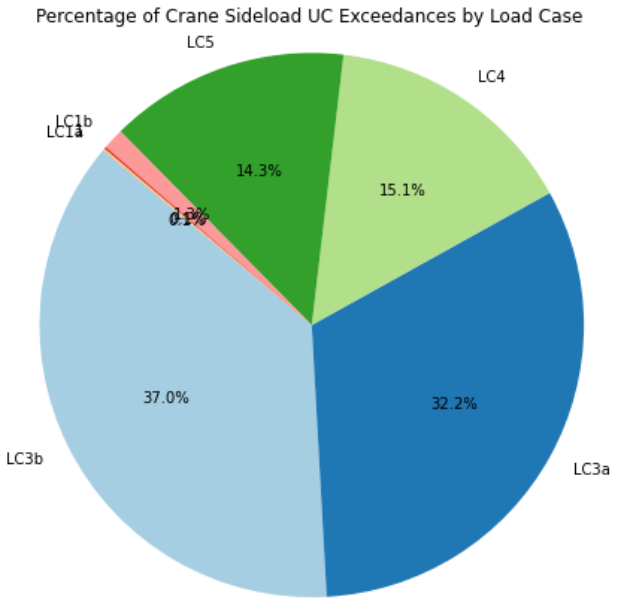


Figure 7: Crane Sideload UC Exceedances by Load Case

This trend is reflected in the broader dataset as well, where LC3a and LC3b also dominate total UC exceedances (see Figure 8).

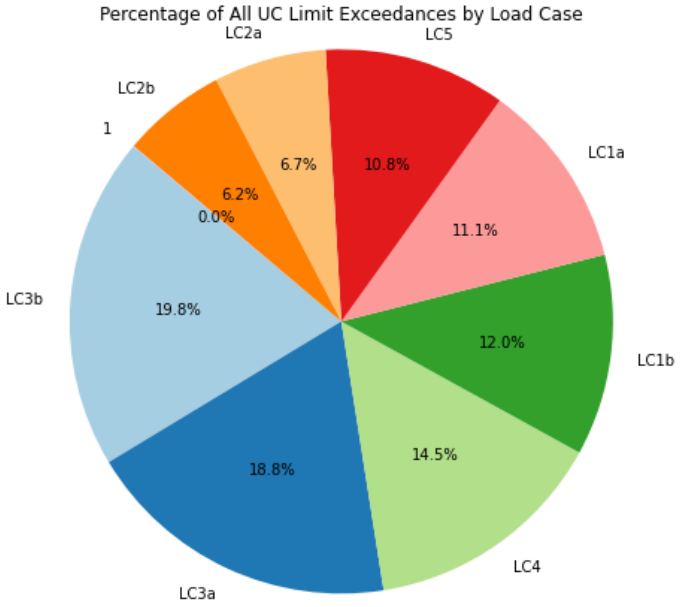


Figure 8: Total UC Exceedances by Load Case

Across all critical load cases, the majority of sideload exceedances occur in the 1.0 to 1.5 UC range, confirming the earlier finding that many violations are marginal. This strengthens the case for optimization efforts aimed at small reductions in dynamic loads, particularly in LC3a and LC3b scenarios.

2.2.6 Environmental Resonance and Dynamic Amplification

To investigate the environmental conditions under which sideload exceedances occur, a heatmap analysis was conducted for **Crane Sideload UC** values as a function of significant wave height (H_s) and peak wave period (T_p). The resulting heatmap in Figure 9 reveals a clear concentration of exceedances in the $T_p = 7$–9 s range.

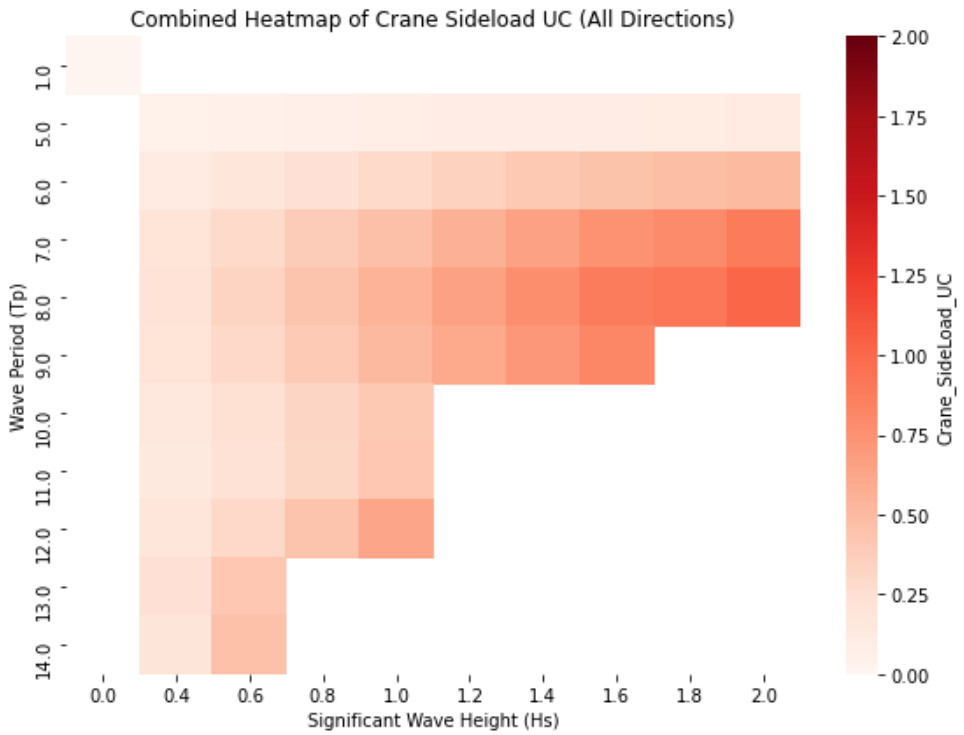


Figure 9: Crane Sideload UC as a Function of H_s and T_p (All Directions)

This clustering around $T_p = 7$–9 s is particularly relevant in light of the dynamic characteristics of Load Case LC3a. The second natural frequency of the system for LC3a, which corresponds to a dominant pendulum mode involving sidelead motion of the suspended monopile, has a period of $T_n = 8.16$ s. The proximity between this natural period and the peak of the exceedance envelope in the heatmap suggests that dynamic amplification due to resonance is a key contributor to the sideload limit violations observed in LC3a.

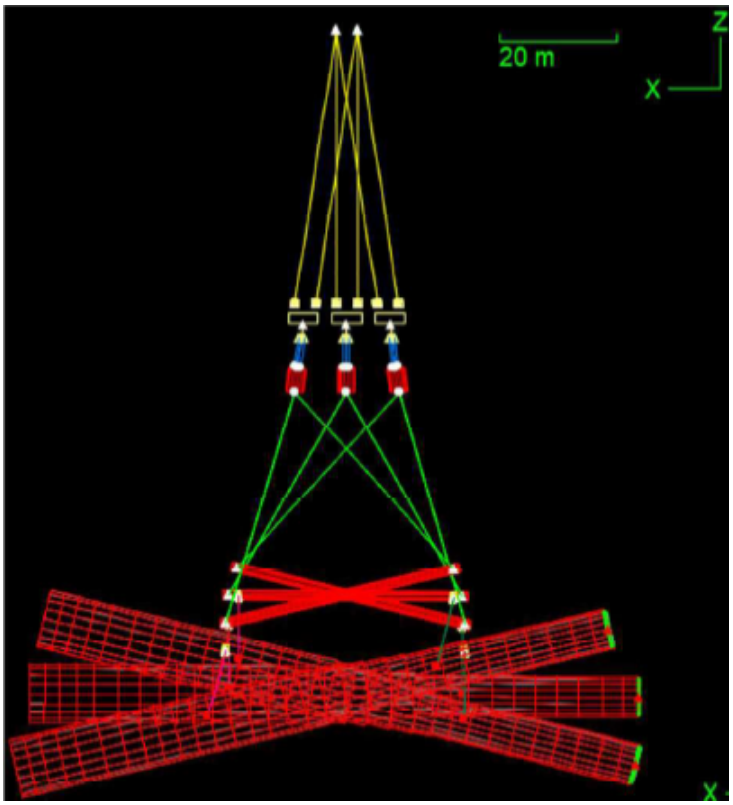


Figure 10: Illustration of the double pendulum sidelead motion in the crane rigging system

Figure 10 visualizes the double pendulum behavior responsible for this motion. As sea states with $T_p \approx 8$ s excite this mode, large oscillations of the rigging occur, increasing both sideload and sling tension demands. The geometry shown here highlights the sensitivity of the system to dynamic excitation.

This resonance hypothesis highlights a potentially problematic interaction between the crane rigging system and the offshore environment. Although the time-domain UC simulations are unweighted and systematically cover a broad range of conditions, the historical scatter diagram (Figure 11) shows that wave periods around $T_p = 8$ s occur frequently at the project location.

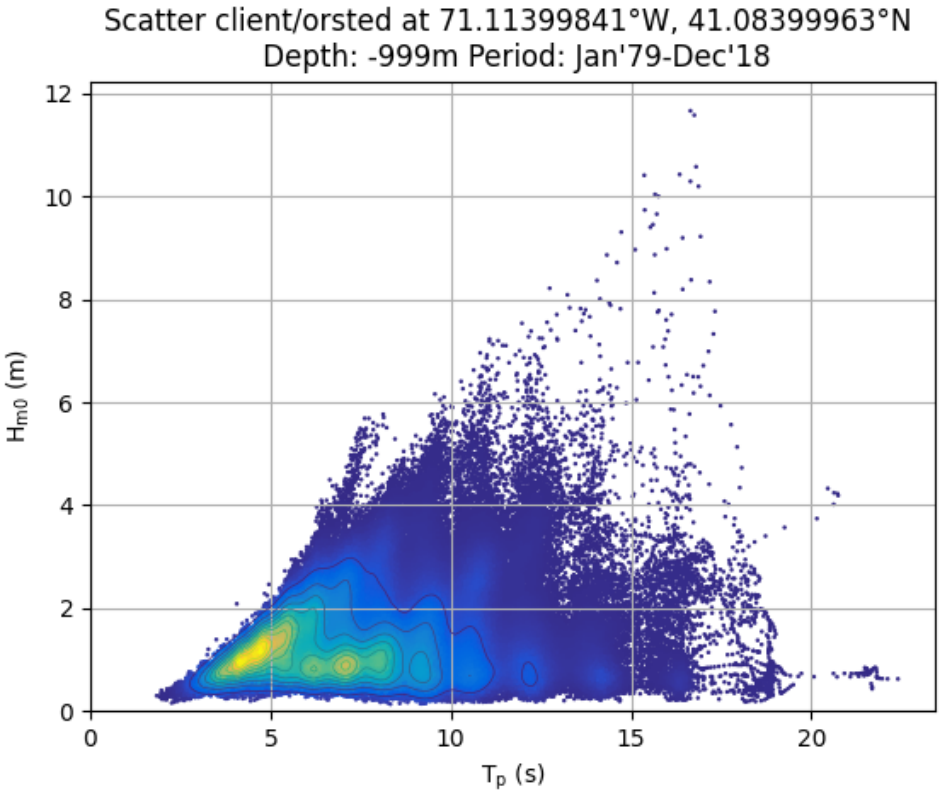


Figure 11: Historical Scatter Diagram of Project Site (South Fork)

This environmental clustering around the system's second natural period (8.16 s for LC3a) implies that resonance is not just a theoretical risk, but a realistic and recurring operational concern. As such, mitigation of resonance effects should be a central goal in future optimization efforts—either by shifting the natural frequency away from this range or introducing damping mechanisms to reduce dynamic amplification.

The optimization strategy developed in the following chapter builds directly on these findings, with a specific focus on Load Case LC3a and the avoidance of resonance near $T_p = 8.16$ s.

3 Theoretical Background

3.1 Modal Analysis

Modal analysis is a fundamental process in structural dynamics used to determine the inherent dynamic characteristics of a system in the form of natural frequencies and mode shapes. In essence, it describes how a structure or system tends to vibrate if it is disturbed from its rest position (Chopra, 2017).

For offshore heavy lifting, understanding these properties is critical. The suspended monopile and rigging form a complex multi-body pendulum system with a set of distinct natural periods of oscillation. If the period of an external excitation—such as the vessel motion induced by ocean waves—matches one of these natural periods, a state of **resonance** can occur. This condition leads to a dramatic amplification of motion and internal loads, which can compromise the safety and feasibility of the operation (Faltinsen, 1990). Therefore, modal analysis is the first and most critical step in predicting and mitigating such adverse dynamic behaviour.

3.1.1 Derivation of Natural Frequencies

The dynamic behaviour of the undamped pendulum system is described by the linearized matrix equation of motion for free vibration:

$$M\ddot{q}(t) + Kq(t) = 0 \tag{2}$$

Where:

- M is the system's **mass matrix**, which accounts for the inertial properties of the components.
- K is the **stiffness matrix**, which represents the gravitational and geometric restoring forces.
- $q(t)$ is the vector of generalized coordinates representing the system's degrees of freedom.

To find the natural frequencies, a harmonic solution is assumed, of the form $q(t) = \phi e^{i\omega t}$. Substituting this into the equation of motion yields the generalized eigenvalue problem (Clough & Penzien, 1993):

$$(K - \omega^2 M)\phi = 0 \quad \text{or} \quad K\phi = \omega^2 M\phi \tag{3}$$

Solving this problem yields a set of eigenvalues, $\lambda_i = \omega_i^2$, and corresponding eigenvectors, ϕ_i.

- The **eigenvalues** (ω_i^2) are the squares of the system's natural angular frequencies (ω). The natural frequencies (f_n) and periods (T_n) are then calculated as $f_n = \omega_n/(2\pi)$ and $T_n = 1/f_n$.
- The **eigenvectors** (ϕ_i) are the **mode shapes**, which describe the characteristic pattern of physical displacement for each natural frequency.

In this thesis, this eigenvalue problem is solved numerically using the mass and stiffness matrices derived from the Lagrangian model. While this problem can be solved analytically by finding the roots of the characteristic equation, $\det(K-\omega^2 M) = 0$, this approach is computationally impractical for multi-degree-of-freedom systems.

Instead, a robust numerical solver from the SciPy library is used. The implementation directly solves the generalized eigenvalue problem using the following command:

```
w2, Phi = LA.eigh(K, M)
```

Here, `LA.eigh` is a highly optimized function from the `scipy.linalg` module designed to solve the eigenvalue problem for real symmetric or complex Hermitian matrices (The SciPy community, 2024a). The function takes the stiffness matrix `K` and mass matrix `M` as input and returns:

- `w2`: A vector containing the eigenvalues ($\lambda_i = \omega_i^2$) of the system.
- `Phi`: A matrix whose columns are the corresponding normalized eigenvectors (ϕ_i), or mode shapes.

The natural angular frequencies are then obtained by taking the square root of the eigenvalues:

```
wn = np.sqrt(w2)
```

This numerical approach provides an efficient and accurate solution for the system's modal properties.

3.2 Lagrangian Mechanics

Lagrangian mechanics is a formulation of classical mechanics that provides a powerful and systematic procedure for deriving the equations of motion for a system. Unlike the Newtonian approach, which focuses on forces and vector quantities, the Lagrangian method is based on a scalar quantity known as the **Lagrangian**, which characterizes the system's dynamics in terms of its kinetic and potential energies (Goldstein, Poole, & Safko, 2002). This approach is particularly advantageous for complex systems with multiple interconnected bodies and mechanical constraints, such as the pendulum model used in this thesis (Hand & Finch, 1998).

The primary advantage of this method is its use of **generalized coordinates**.

3.2.1 Generalized Coordinates and the Lagrangian

A set of generalized coordinates, denoted as q_i, is any collection of parameters that uniquely defines the configuration of a system at any instant in time. For the monopile rigging, these coordinates are the angles of the various sling elements (Chopra, 2017). The key benefit of this approach is that it implicitly incorporates kinematic constraints. For example, by defining the system's geometry with angles, the fixed lengths of the rigging cables are automatically satisfied, eliminating the need to solve for the internal tension forces that maintain these constraints (Goldstein et al., 2002). This significantly simplifies the derivation of the equations of motion.

The central function in this method is the **Lagrangian ($\mathcal{L}$)**, defined as the difference between the total kinetic energy (T) and the total potential energy (V) of the system:

$$\mathcal{L}(q, \dot{q}, t) = T(q, \dot{q}, t) - V(q, t) \tag{4}$$

where $\dot{q}$ represents the time derivatives of the generalized coordinates (generalized velocities).

3.2.2 The Euler-Lagrange Equation

The equations of motion are derived from the Lagrangian using the **Euler-Lagrange equation**. For each generalized coordinate q_i, the equation is given by:

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{q}_i}\right) - \frac{\partial \mathcal{L}}{\partial q_i} = Q_i \tag{5}$$

where Q_i represents the generalized non-conservative forces (such as friction, damping, or external actuation) associated with the coordinate q_i. For a conservative system, $Q_i = 0$ for all coordinates (Hand & Finch, 1998).

In this thesis, this framework is applied to derive the equations of motion for the multi-body pendulum system. The kinetic energy (T) is formulated from the translational and rotational velocities of the masses, while the potential energy (V) arises from the change in vertical position of the masses in the Earth's gravitational field. Applying the Euler-Lagrange equation for each angular coordinate systematically yields the coupled second-order differential equations that govern the system's dynamics. This derivation is performed symbolically in Python, which facilitates the automatic generation of the mass (M) and stiffness (K) matrices used for the subsequent dynamic analysis.

3.3 JONSWAP Wave Spectrum

Ocean waves are inherently irregular and are best described as a random process. To analyze the effect of such an environment on an offshore structure, the sea state is characterized by a **wave energy spectrum**, or Power Spectral Density (PSD). A wave spectrum, $S(\omega)$, describes how the total energy of the sea state is distributed across different wave frequencies (ω) (Faltinsen, 1990). The area under the spectrum is related to the significant wave height (H_s), while the frequency at which the spectrum has its maximum value corresponds to the peak period (T_p).

While several mathematical models for wave spectra exist, the **JONSWAP spectrum** is widely adopted in offshore engineering for its realistic representation of wind-generated seas and is used by Boskalis for their analyses. It is an empirical spectrum, developed based on extensive data collected during the Joint North Sea Wave Project in 1973 (Hasselmann et al., 1973).

The JONSWAP spectrum is fundamentally a modification of the earlier **Pierson-Moskowitz (PM) spectrum**. The PM spectrum describes a "fully developed sea," a theoretical equilibrium state where

the waves have received the maximum possible energy from a wind that has been blowing at a constant speed over an infinite fetch (distance) and duration (DNV GL, 2019). However, many real-world sea states, particularly in locations like the North Sea, are "fetch-limited," meaning they are still developing and have not reached this equilibrium.

In such developing seas, the wave energy is concentrated more sharply around the peak frequency than predicted by the PM spectrum. The key innovation of the JONSWAP formulation is the inclusion of a **peak enhancement factor (γ)** to account for this. This factor creates a sharper, more pronounced spectral peak, which is critical for the resonance analysis central to this thesis. A higher value of γ indicates that a larger proportion of the wave energy is concentrated near the peak period, increasing the likelihood of exciting a structural mode if its natural period is close to T_p (Faltinsen, 1990).

Due to its ability to model these peaked, developing sea states, the JONSWAP spectrum provides a more realistic and often more conservative basis for the dynamic analysis of offshore structures than the broader PM spectrum. The complete mathematical formulation used in this thesis is detailed in Section 4.7.1.

3.4 Frequency-Domain Analysis

While time-domain analysis simulates a system's response step-by-step, which can be computationally prohibitive for assessing long-term operational statistics, **frequency-domain analysis** offers an efficient alternative. This approach is exceptionally well-suited for determining the statistical response of a linear system subjected to a stationary random process, such as a floating structure in a given sea state (Journée & Massie, 2008). Instead of a detailed time history, this method yields a statistical description of the response, such as its variance and expected peak values, by analysing how the system behaves across a spectrum of frequencies.

3.4.1 Transfer Functions and RAOs

The foundation of frequency-domain analysis for a linear system is the **Transfer Function**, denoted $H(\omega)$. For a system that responds linearly, a transfer function is a complex-valued mathematical representation that describes how it modifies a sinusoidal input of a given frequency (ω) to produce a sinusoidal output. Its magnitude, $|H(\omega)|$, represents the ratio of the output amplitude to the input amplitude, while its argument, $\arg(H(\omega))$, represents the phase shift between the input and output signals. The transfer function thus completely characterizes the dynamic response of a linear system (Journée & Massie, 2008).

In naval architecture and offshore engineering, a specific and widely used form of transfer function is the **Response Amplitude Operator (RAO)**. The RAO of a vessel quantifies the amplitude and phase of its motion in each of the six degrees of freedom (surge, sway, heave, roll, pitch, and yaw) as a response to a unit-amplitude ocean wave of a specific frequency and direction (Faltinsen, 1990).

In this thesis, transfer functions are used in two key stages:

1. **Vessel RAOs** are used to translate the incident wave field into the motion experienced at the crane tip, as detailed in Section 4.7.2.
2. A dynamically derived **motion-to-load transfer function**, $H_{tip \to load}(\omega)$, is calculated to describe how the motion of the crane tip is converted into a horizontal side load by the pendulum dynamics of the rigging, as derived in Section 5.3.

3.4.2 Spectral Analysis

Spectral analysis provides the framework for applying the transfer function concept to random processes like ocean waves. The core principle for a linear system is that the Power Spectral Density (PSD) of the response can be calculated directly from the PSD of the input and the system's transfer function (Journée & Massie, 2008). This fundamental relationship is expressed as:

$$S_{response}(\omega) = |H(\omega)|^2 S_{input}(\omega) \tag{6}$$

In this thesis, this principle is applied to determine the side-load response spectrum ($S_F(\omega)$) from the wave spectrum ($S_\eta(\omega)$) through a series of transformations involving the vessel and rigging dynamics. The complete formulation for this calculation is detailed in Section 4.7.

The resulting side-load spectrum contains all the necessary information to describe the load as a statistical process. From it, key statistical properties are calculated via its **spectral moments**, defined as

$$m_n = \int_0^\infty \omega^n S_F(\omega) d\omega \tag{7}$$

The zeroth moment (m_0) is particularly important, as it represents the variance (σ^2) of the side-load response. This variance is fundamental for estimating the expected maximum loads required for the workability assessment (DNV GL, 2019).

3.5 Optimization with Differential Evolution

To improve the workability, a numerical optimization routine is used to find the best possible rigging geometry. The goal is to find the set of design parameters (the rigging lengths) that maximizes an objective function (workability) while satisfying a series of physical and operational constraints.

Given the complex, non-linear, and potentially multi-modal relationship between rigging geometry and workability, a global optimization algorithm known as **Differential Evolution (DE)** was selected. DE is a powerful and widely used evolutionary algorithm that is particularly effective for continuous optimization problems and does not require gradient information, making it robust against finding false solutions in local minima (Storn & Price, 1997).

The algorithm works by maintaining a population of candidate solutions and iteratively improving them through a process analogous to natural evolution:

- **Initialization:** A population of parameter vectors is randomly initialized within the feasible solution space.
- **Mutation:** For each "target" vector in the population, a "mutant" vector is generated by taking the difference between two other randomly chosen vectors and adding it to a third. This use of vector differences is the signature feature of DE.
- **Crossover:** A new "trial" vector is created by mixing the parameters of the mutant vector with the original target vector, which helps to explore the search space effectively.
- **Selection:** The objective function is evaluated for the trial vector. If the trial vector yields a better result than the original target vector, it replaces the target in the population for the next generation.

This cycle is repeated for a set number of generations, allowing the population to converge towards the global optimum.

In this thesis, the implementation of this algorithm is handled by the `differential_evolution` function from Python's `scipy.optimize` library (The SciPy community, 2024b). The solver requires the user to provide an objective function to be minimized, along with the bounds that define the search space for the input parameters. The function then executes the DE algorithm internally, iteratively calling the objective function to evaluate new candidate solutions until it converges or reaches a maximum number of iterations.

4 Model Description and Methodology

4.1 Analytical Model Overview

4.1.1 System of units

In this model, all the calculations are performed in SI units; special attention has been given to the conversion of units in Hertz and Rad/s as they are both used throughout the model.

4.1.2 Simulation Time

A simulation duration of 3 hours for the most probable maximum was chosen. This is a standard duration for short-term statistical analysis in offshore engineering, representing a period over which a sea state can be considered stationary (DNV, 2021). It also reflects a realistic and conservative window for completing the lifting operation.

4.1.3 Coordinate system and vessel motions

The model is formulated in a right-handed, vessel-fixed coordinate system. The origin is located at the stern of the vessel, on the centerline and at keel level. Axes are defined as:

- The **x-axis** points forward along the vessel's longitudinal direction (towards the bow),
- The **y-axis** points port-side (transverse to the left),
- The **z-axis** points vertically upward, normal to the main deck.

This local frame is used to describe the vessel's motions, as well as the position and dynamics of the crane.

The vessel's motion response due to wave excitation is modeled using all six degrees of freedom (DOF), defined according to standard marine terminology:

- **Surge** (x) – translational motion along the x-axis,
- **Sway** (y) – translational motion along the y-axis,
- **Heave** (z) – translational motion along the z-axis,
- **Roll** (ϕ) – rotation about the x-axis,
- **Pitch** (θ) – rotation about the y-axis,
- **Yaw** (ψ) – rotation about the z-axis.

These motions are visualized in Figure 12. The frequency-dependent response amplitude operators (RAOs) for each DOF are used in the model to reconstruct crane tip motions and, ultimately, to compute the resulting side-load spectra.

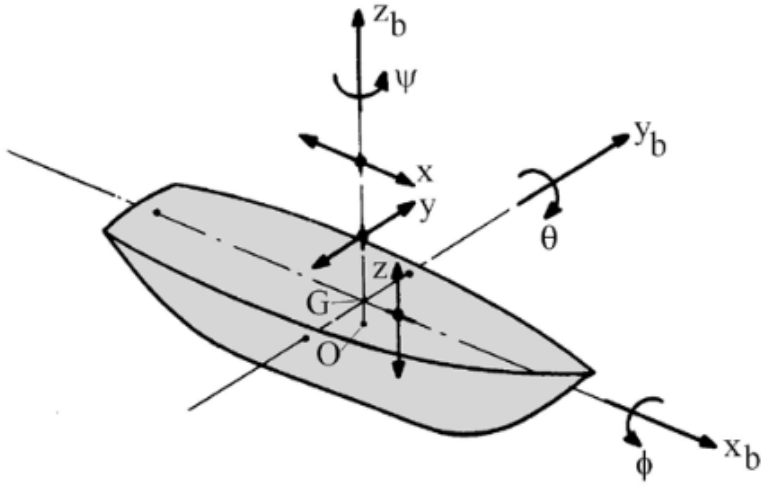


Figure 12: Definition of vessel motions in six degrees of freedom (DOF).

Adapted from (Journée & Massie, 2008).

The wave direction is expressed through the *angle of attack* γ, which is defined as the angle between the vessel's x-axis (bow direction) and the principal wave direction. An angle of $\gamma = 180°$ corresponds to head seas, while $\gamma = 0°$ corresponds to following seas. This convention is illustrated in Figure 13.

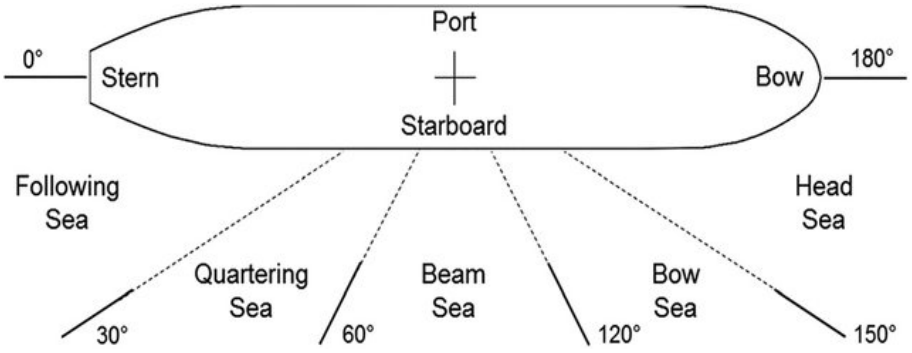


Figure 13: Top view of vessel showing heading and wave approach angle γ.

Adapted from (Ishihara & Kashiwagi, 2018).

In addition to the vessel-fixed reference frame, a secondary coordinate system is defined at the crane tip. This local frame is used to evaluate motions and loads at the connection point between the crane and the suspended monopile. The origin of this system lies at the crane tip. Its orientation rotates along with the crane boom, such that a change in crane slewing angle directly reorients the local frame. The local x-axis points in the offlead direction of the crane (along the boom), the y-axis in the sidelead direction, and the z-axis aligns vertically with the hoist line.

For the load case considered in this study, the crane is rotated 90° with respect to the vessel-fixed x-axis. This coordinate setup is illustrated in Figure 14.

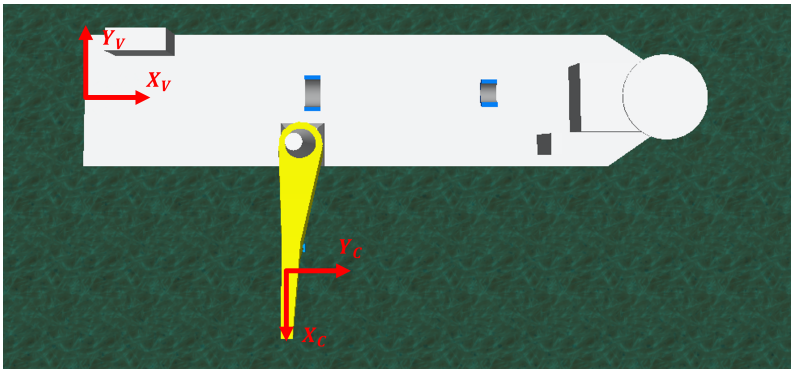


Figure 14: Definition of crane tip–fixed coordinate system, rotated 90° relative to vessel-fixed x-axis.

4.2 Considered Load Case

Several operational phases can be distinguished during the monopile installation sequence, each with its own dynamic characteristics and load exposure on the crane and rigging system. Lifting from the transport grillage, upending, and pile stabbing each involve different load distributions, rigging geometries, and motion constraints. For this thesis, the focus is placed on the *lift-off phase* — the moment the monopile is lifted from the heavy transport vessel (HTV) and becomes fully suspended in the rigging.

This decision is based on the findings from the problem analysis, where the lift-off phase was identified as a dominant contributor to operational downtime. During this phase, the suspended monopile is subject to the full range of relative vessel motions between the HTV and heavy lift vessel (HLV). The dynamic amplification of side loads on the crane tip, combined with a narrow allowable motion envelope near the HTV grillage, makes this phase particularly sensitive to wave direction, period, and amplitude.

The full sequence of considered load cases is summarised below, with the lift-off condition marked as the focus of this study:

- **Empty spreader:** Spreader bar suspended approximately 10 m above the monopile; belly slings not yet connected.

- **Pre-tension:** Rigging attached to the monopile; spreader approximately 5 m above the pile; 50% of the monopile's weight in the hook.
- **Lift-off (focus case):** Monopile lifted 3 m above the HTV grillage; 100% of the weight is suspended; full dynamic loading applies.
- **Storage lane:** Monopile lifted from the storage lane on the HLV deck; again, approximately 3 m above the grillage.
- **Upend hinge:** Monopile positioned above the hinge before lowering; 3 m above the upending support structure.

A depiction of the load case and relative positioning of the HTV and HLV, is shown in Figure 15.

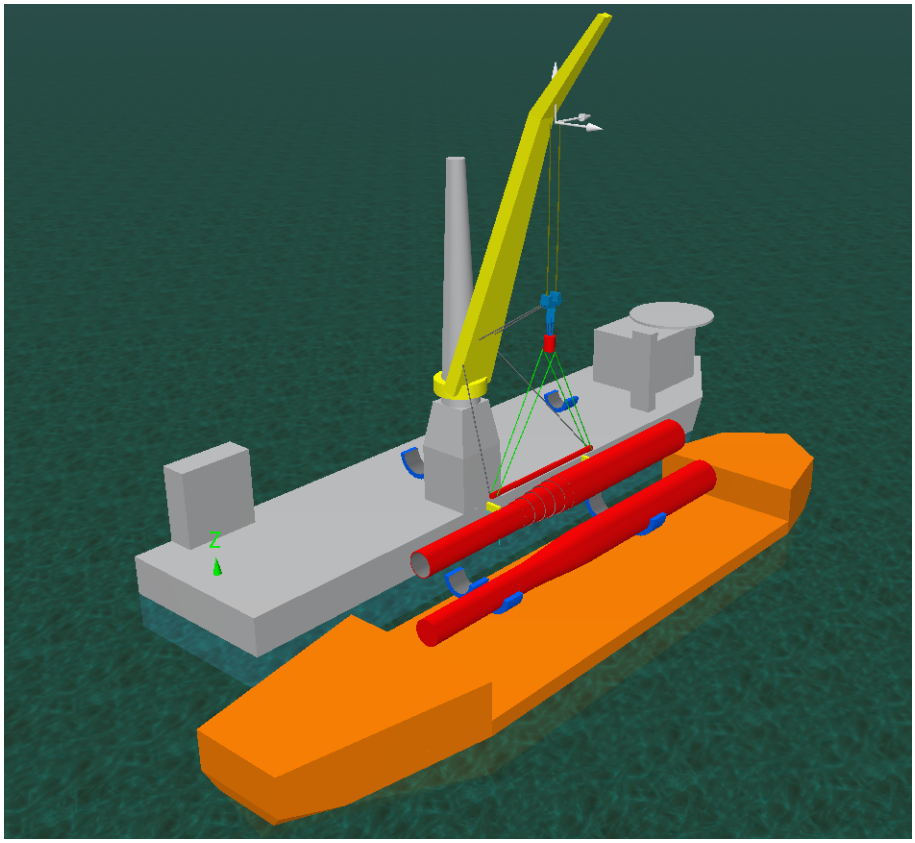


Figure 15: Illustration of the lift-off load case considered in this study.

4.3 Objects

The main objects considered in this study are the Bokalift 2 and the B10 Monopile. Their characteristics are presented here. The nominal values, used in the dynamic analysis, can be found in Table 2.

4.3.1 Vessel

The vessel in this model represents a generic heavy lift vessel (HLV) used for offshore monopile installation. It is modeled as a rigid body with six degrees of freedom (DOF), excited by wave motions defined through frequency-dependent response amplitude operators (RAOs). These RAOs are used to reconstruct the crane tip motion as input to the dynamic model.

Nonlinear damping effects (e.g. in roll) are not explicitly included. The vessel is assumed to maintain zero heel and trim, and mean drift forces are compensated by the dynamic positioning (DP) system.

Representative vessel characteristics:

- Length overall: *231* m
- Beam: *49* m
- Operational draft: *10* m

- DP system: *Kongsberg DP-2*

Figure 16: Side view of the Bokalift 2 heavy lift vessel with crane configuration.

4.3.2 Monopile

The monopile is suspended through a rigging system of slings and a spreader bar. The monopile is modeled as a rigid mass with its center of gravity located below the spreader. No hydrodynamic loads are considered, as the monopile remains fully above the waterline during lift-off.

Representative monopile characteristics:

- Monopile mass: *[1868,3]* t
- Monopile length: *[107]* m

Figure 17: Illustration of the monopile.

The dynamic behavior of the monopile is further described in the next section using a Lagrangian multi-body framework.

4.4 Lagrangian Approach

This section presents the derivation of the equations of motion using a Lagrangian formulation. The system is modeled as a four-body planar pendulum, suspended from a crane tip that moves horizontally due to vessel-induced excitation. The generalized coordinates include both the crane tip displacement and the internal angular motion of each body in the system:

$$q(t) = \left[\theta_1(t),\ \theta_2(t),\ \theta_3(t),\ \theta_4(t), R(t),\right]^T$$

The inclusion of $R(t)$ as a generalized coordinate allows direct coupling between the base excitation and the internal dynamics of the system. This not only enables the derivation of inertial coupling terms between horizontal motion and pendulum swing, but also allows direct computation of the crane side load from the Lagrangian formulation — the critical parameter targeted for reduction in this study.

4.4.1 Motivation for the Lagrangian Method

The Lagrangian formulation was chosen due to the hierarchical structure of the multibody system and the presence of both translational and rotational degrees of freedom. While Newtonian force balances would require complex vector constraint handling at each joint, the Lagrangian approach enables a scalar energy-based formulation where constraints are implicitly satisfied by the generalized coordinate choice. This makes the model compact, symbolically tractable, and easier to linearize.

In this implementation, symbolic derivation of the mass and stiffness matrices in Python enables automation of eigenmode analysis and integration into the optimization routine.

4.4.2 Generalized Coordinates and Position Definitions

The five generalized coordinates are:

- $\theta_1(t)$: angle of the upper slings,
- $\theta_2(t)$: angle of the lower slings,
- $\theta_3(t)$: angle of the spreader bar,
- $\theta_4(t)$: angle of the monopile,
- $R(t)$: horizontal motion of the crane tip (driven by vessel).

These coordinates correspond to the motion of four suspended rigid bodies forming a serial pendulum system below the crane tip. The associated masses are:

- m_1: crane hook,
- m_2: quick connect element,
- m_3: spreader bar,
- m_4: monopile.

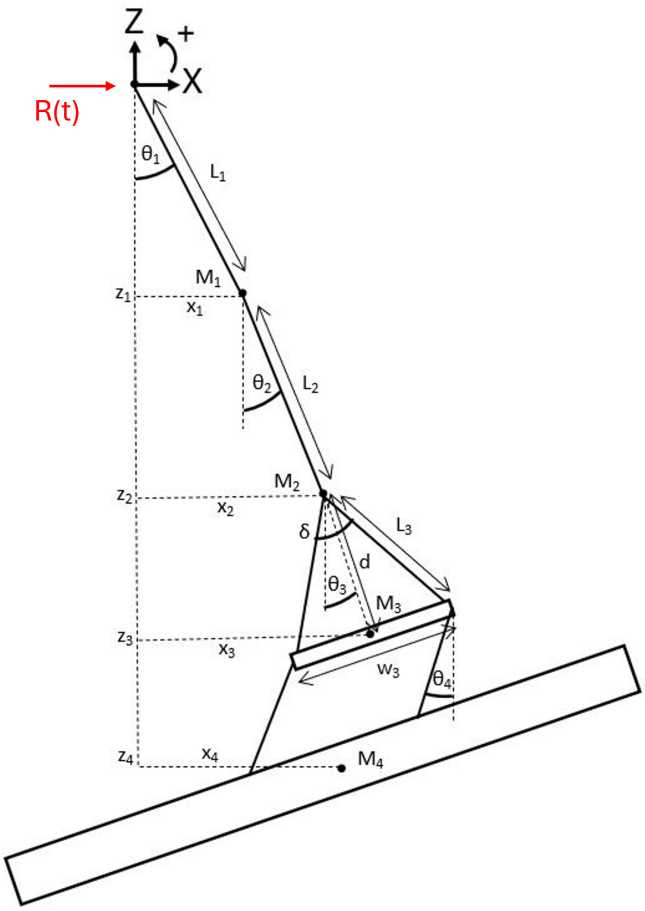


Figure 18: Generalized coordinates and mass positions in the pendulum system.

The model assumes small angular displacements of all elements, using the linearized trigonometric approximations:

$$\sin(\theta_i) \approx \theta_i, \qquad \cos(\theta_i) \approx 1 - \frac{\theta_i^2}{2} \tag{8}$$

Modeling Assumptions The physical setup and mathematical formulation are based on the following simplifying assumptions:

- **Planar motion:** All movement is confined to a vertical plane defined by the crane boom and gravity vector. Out-of-plane motions (e.g., torsion or yaw) are neglected.
- **Small-angle approximation:** Angular displacements are assumed to remain small, enabling linearization of trigonometric functions ($\sin(\theta) \approx \theta$ and $\cos(\theta) \approx 1 - \theta^2/2$). This assumption is well-justified by the operational constraints of the system. The maximum allowable sidelead angle for the crane is 2.0 degrees; any greater angle would cause other Unity Checks (UCs) to be exceeded, halting the operation. For an angle of 2.0 degrees (or 0.035 radians), the error introduced by the approximation $\sin(\theta) \approx \theta$ is less than 0.02% (Taylor, 2005). Therefore, within the entire workable envelope of the operation, the small-angle approximation holds with very high accuracy, ensuring the validity of the linearized system matrices.
- **Rigid body assumption:** All suspended components — including slings, spreader, and monopile — are treated as rigid and undeformable.
- **Tensioned system:** All cables remain taut and are modeled as inextensible and massless, with effectively infinite axial stiffness ($\sim 700 \times 10^9$ kN).
- **Symmetric geometry:** The mass distribution and rigging configuration are symmetric about the vertical plane of motion. This ensures that sidelead dynamics dominate and simplifies the formulation.

Under the small-angle assumption, the horizontal positions of the mass centers are defined recursively as:

$$
\begin{aligned}
x_1(t) &= R(t) + L_1\theta_1(t) \\
x_2(t) &= R(t) + L_1\theta_1(t) + L_2\theta_2(t) \\
x_3(t) &= R(t) + L_1\theta_1(t) + L_2\theta_2(t) + d\theta_3(t) \\
x_4(t) &= R(t) + L_1\theta_1(t) + L_2\theta_2(t) + d\theta_3(t) + L_4\theta_4(t)
\end{aligned}
$$

with corresponding velocities:

$$
\begin{aligned}
\dot{x}_1 &= \dot{R} + L_1\dot{\theta}_1 \\
\dot{x}_2 &= \dot{R} + L_1\dot{\theta}_1 + L_2\dot{\theta}_2 \\
\dot{x}_3 &= \dot{R} + L_1\dot{\theta}_1 + L_2\dot{\theta}_2 + d\dot{\theta}_3 \\
\dot{x}_4 &= \dot{R} + L_1\dot{\theta}_1 + L_2\dot{\theta}_2 + d\dot{\theta}_3 + L_4\dot{\theta}_4
\end{aligned}
$$

Vertical positions are constant under small-angle assumptions, hence $\dot{z}_i \approx 0$. The recursive formulation introduces strong coupling in the mass matrix, particularly between the crane base excitation and the suspended bodies. Note that the entire dynamic response is driven by the combined motion of $R(t)$ and the angular displacements of the rigging and monopile.

It is important to note that dynamic coupling from the suspended monopile back to the vessel is neglected in this model. In other words, the mass of the monopile does not influence the vessel motion. This simplification is justified by the relatively small mass of the payload compared to the total displacement of the vessel, making its feedback effect on vessel dynamics negligible in the context of side load prediction.

However, during the development of this model, indications of vessel–monopile coupling were observed near resonance frequencies. While this phenomenon could be captured by extending the model to include bidirectional coupling, it was considered beyond the scope of this work. Since the optimization approach is aimed at averting resonance by shifting natural frequencies away from dominant wave energy, the practical impact of the coupling is mitigated. This point is revisited in the validation section, where differences between the analytical and OrcaFlex models are discussed.

4.4.3 Kinetic Energy Formulation

The total kinetic energy consists of translational terms for each mass and rotational contributions from the suspended bodies:

$$
T = \sum_{i=1}^{4} \left(\frac{1}{2} m_i \dot{x}_i^2 \right) + \frac{1}{2} I_1 \dot{\theta}_1^2 + \frac{1}{2} I_2 \dot{\theta}_2^2 + \frac{1}{2} I_3 \dot{\theta}_3^2 + \frac{1}{2} I_4 \dot{\theta}_3^2 \tag{9}
$$

The kinetic energy of each mass m_i includes its horizontal velocity $\dot{x}_i$, derived previously under small-angle assumptions. The terms $I_i \dot{\theta}_i^2$ represent the rotational energy around each body's center of mass.

It is important to note that the rotational energy contribution of the fourth mass (the monopile) is treated differently. Rather than using its own angle θ_4, its rotation is assumed to be governed by θ_3, the rotation of the spreader bar. This modeling choice reflects the assumption that both belly slings remain taut at all times — i.e., no line slackening is allowed.

Under this assumption, the monopile remains effectively parallel to the spreader bar throughout the motion. As a result, its angular motion is fully captured by the spreader bar rotation θ_3, and its rotational kinetic energy is expressed as:

$$
\frac{1}{2} I_4 \dot{\theta}_3^2
$$

4.4.4 Potential Energy Formulation

The potential energy is based on the vertical displacement of each mass from its equilibrium height:

$$
V = g \sum_{i=1}^{4} m_i z_i \tag{10}
$$

Using the small-angle approximation introduced in Equation (8), the vertical positions can be written as:

$$z_i \approx z_{i,0} - L_i \frac{\theta_i^2}{2}$$

which yields:

$$V \approx V_0 + \sum_{i=1}^{4} \frac{1}{2} g m_i L_i \theta_i^2$$

Here, V_0 is the constant potential energy at equilibrium, and the remaining terms form a quadratic expression in the generalized coordinates θ_i. While the potential energy function itself is quadratic in θ_i, its second derivative with respect to the coordinates — which defines the stiffness matrix K — is constant. This is what makes the resulting system linear in $q(t)$, as it leads to a constant matrix in the linearized equations of motion:

$$K = \frac{\partial^2 V}{\partial q^2}$$

Hence, the use of small-angle approximations transforms the trigonometric potential into a quadratic polynomial, from which a linear restoring force model can be directly obtained.

4.4.5 Linearized System and Matrix Form

the Lagrangian is linearized around the equilibrium configuration $q = 0$, leading to the classical linear second-order system:

$$M\,\ddot{q}(t) + K\,q(t) = 0 \tag{11}$$

Here, $q = [\theta_1, \theta_2, \theta_3, \theta_4, R(t)]^\mathsf{T}$ is the generalized coordinate vector. The system matrices are obtained as second derivatives of the energy terms. In matrix notation:

$$M = \frac{\partial^2 T}{\partial \dot{q}^2}, \qquad K = \frac{\partial^2 V}{\partial q^2} \tag{12}$$

In index notation, this reads:

$$M_{ij} = \frac{\partial^2 T}{\partial \dot{q}_i\, \partial \dot{q}_j}, \qquad K_{ij} = \frac{\partial^2 V}{\partial q_i\, \partial q_j} \tag{13}$$

This formulation enables symbolic derivation of the system matrices in Python, allowing for automated analysis, modal decomposition, and parametric optimization.

4.5 Constants and Initial Parameters

After the symbolic derivation of the system dynamics using the Lagrangian formulation, numerical values were assigned to all physical parameters to perform simulations and evaluate modal behavior. These constants represent the offshore monopile installation configuration used in the Revolution offshore wind project in the US completed by Boskalis in 2025

The physical meaning and layout of these parameters are illustrated in Figure 18, which was introduced in the Lagrangian model subsection. It shows the mass locations, segment lengths, and generalized coordinates used throughout the model.

The numerical values used in the implementation are summarized in Table 2.

Table 2: Nominal model parameters used in the pendulum system.

Parameter	Description	Value
m_1	Mass of crane hook	226,790 kg
L_1	Length of upper slings	43.1 m
I_1	Inertia of the crane hook	529,800 kg·m^2
m_2	Mass of quick connect	39,000 kg
L_2	Length of lower slings	5.5 m
I_2	Inertia of quick connect	58,240 kg·m^2
m_3	Mass of spreader bar	170,000 kg
L_3	Length of the diagonal spreader bar cables	40.0 m
w_3	Width of spreader bar	42.0 m
I_3	Inertia of spreader bar	16,920,000 kg·m^2
m_4	Mass of monopile	1,868,300 kg
L_4	Length of the bellyslings	15.5 m
I_4	Inertia of monopile	1,789,000,000 kg·m^2
g	Gravitational acceleration	9.81 m/s^2

The vertical distance d between the quick connect and the center of mass of the spreader is calculated using geometric relations:

$$d = \sqrt{L_3^2 - \left(\frac{w_3}{2}\right)^2} \tag{14}$$

These parameter values are used in the evaluation of the mass and stiffness matrices, natural frequencies, and subsequent frequency-domain analysis.

4.6 Rayleigh Damping and Modal Response

To capture energy dissipation in the dynamic model, proportional (Rayleigh) damping is applied. This approach assumes the damping matrix C is a linear combination of the mass and stiffness matrices:

$$C = \alpha M + \beta K \tag{15}$$

Here, α and β are Rayleigh damping coefficients, calibrated such that the first two natural frequencies of the system exhibit a target damping ratio of $\zeta = 2\%$. These modes dominate the dynamic response of the crane–rigging–monopile system in the frequency range relevant for operational sea states. Higher modes typically lie outside the bandwidth of wave excitation and contribute less to overall motion and side loads.

The damping ratio of 2% is chosen to reflect realistic total damping levels observed in offshore lifting, which include structural damping, hydrodynamic energy dissipation, and control-induced effects. This value is consistent with industry practice and modeling assumptions used in tools such as OrcaFlex and (DNV, 2021).

The Rayleigh coefficients are computed from the closed-form expressions:

$$\begin{bmatrix} \frac{1}{\omega_1} & \omega_1 \\ \frac{1}{\omega_2} & \omega_2 \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = 2\zeta \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

where ω_1 and ω_2 are the angular natural frequencies of the first and second modes, obtained from the modal analysis.

Once the coefficients are determined, the damping matrix C is constructed and projected back into modal space to calculate the damping ratios for all modes:

$$\zeta_i = \frac{\phi_i^T C \phi_i}{2\omega_i}$$

Figure 19 shows the resulting damping ratio as a function of response frequency. As expected, the modal damping closely matches the target near modes 1 and 2, while it increases for higher-frequency modes. This frequency-dependent damping behavior is a known characteristic of Rayleigh damping.

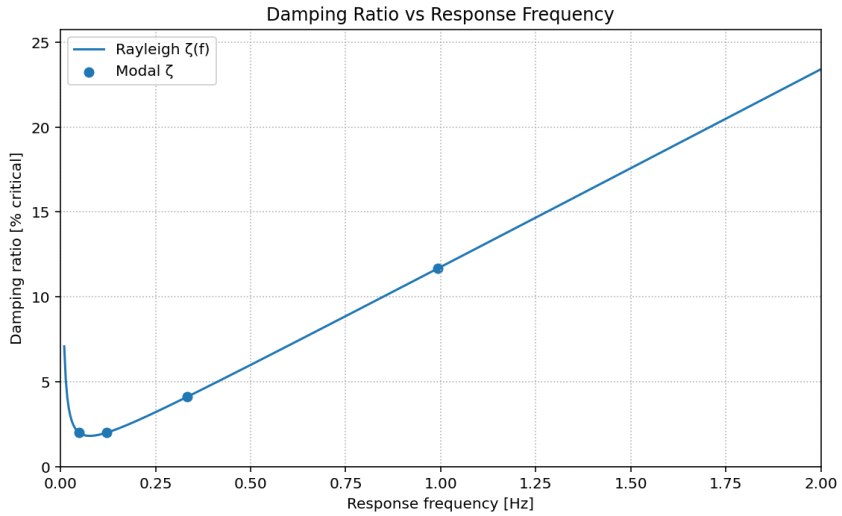


Figure 19: Frequency-dependent damping ratios resulting from Rayleigh damping. Target damping of 2% is matched at the first two modes.

In the context of this model, the use of Rayleigh damping ensures a physically realistic and numerically stable representation of dynamic behavior. Its inclusion is essential, especially given the resonance-sensitive nature of the system: without damping, frequency-domain response peaks would grow unrealistically large, diverging toward infinity near natural frequencies. The damping provides a bounded and interpretable response that better reflects real offshore dynamics.

4.7 Side-load Spectrum Formulation

The side-load power spectral density spectrum (PSD), $S_F(\omega)$, represents the frequency-dependent distribution of lateral dynamic loading at the crane tip due to wave-induced vessel motion. It quantifies how much load energy is expected to occur at each frequency and forms the basis for estimating peak loads in a probabilistic sense.

This spectral representation is particularly suited for analyzing offshore lifting operations in irregular seas. Instead of computing time-domain forces for every sea state, the frequency-domain approach allows direct calculation of the expected load statistics using a known wave spectrum and linear system response.

The side-load spectrum is defined as:

$$S_F(\omega) = S_\eta(\omega) \cdot |\mathbf{u}_{tip,x}(\omega) \cdot H_{tip \to load}(\omega)|^2 \tag{16}$$

Where:

- $S_F(\omega)$ is the side-load PSD [N^2/Hz],
- $S_\eta(\omega)$ is the wave elevation spectrum (JONSWAP) [m^2/Hz],
- $\mathbf{u}_{tip,x}(\omega)$ is the horizontal crane tip RAO [m/m],
- $H_{tip \to load}(\omega)$ is the frequency-dependent transfer function from tip motion to side-load [N/m].

By integrating this spectrum, the most probable maximum side-load F_{max} can be estimated for a given sea state and rigging configuration. This is then used to determine whether operational limits are exceeded. Repeating this evaluation across a range of wave conditions enables calculation of workability — the percentage of time a lift can be safely executed.

The components of Equation (16) are detailed in the following subsections:

- The JONSWAP wave spectrum $S_\eta(\omega)$,

- The crane tip motion response amplitude operator $\mathbf{u}_{tip,x}(\omega)$,
- The motion-to-force transfer function $H_{tip \to load}(\omega)$.

4.7.1 JONSWAP Spectrum

The wave energy spectrum is modeled using the JONSWAP formulation:

$$S(f) = \alpha \cdot \frac{g^2}{(2\pi)^4} \cdot f^{-5} \cdot \exp\left(-\frac{5}{4}\left(\frac{f_m}{f}\right)^4\right) \cdot \gamma^b$$

With:

$$f_m = \frac{1}{T_p} \quad \text{(peak frequency)}$$

$$\gamma = \text{peak enhancement factor}$$

$$b = \exp\left(\frac{-(f - f_m)^2}{2\sigma^2 f_m^2}\right) \quad \text{(peak sharpness)}$$

$$\sigma = \begin{cases} 0.07 & \text{if } f \leq f_m \\ 0.09 & \text{if } f > f_m \end{cases}$$

$$\alpha \text{ is chosen such that } \int S(f)\, df = \frac{H_s^2}{16}$$

The JONSWAP spectrum is widely used in offshore engineering to model wind-generated sea states with a pronounced spectral peak, typical of fetch-limited environments like the North Sea. Compared to the Pierson–Moskowitz spectrum, JONSWAP introduces a peak enhancement factor γ that better captures the sharp energy concentration observed in real sea conditions. Its parametric simplicity and empirical foundation make it a practical and physically meaningful choice for evaluating operational response in offshore installation scenarios.

This spectrum $S(f)$ describes the distribution of wave energy across frequencies and forms the basis of the input excitation in the side-load spectrum. A typical example is shown in Figure 20, using $H_s = 1$ m and $T_p = 8$ s.

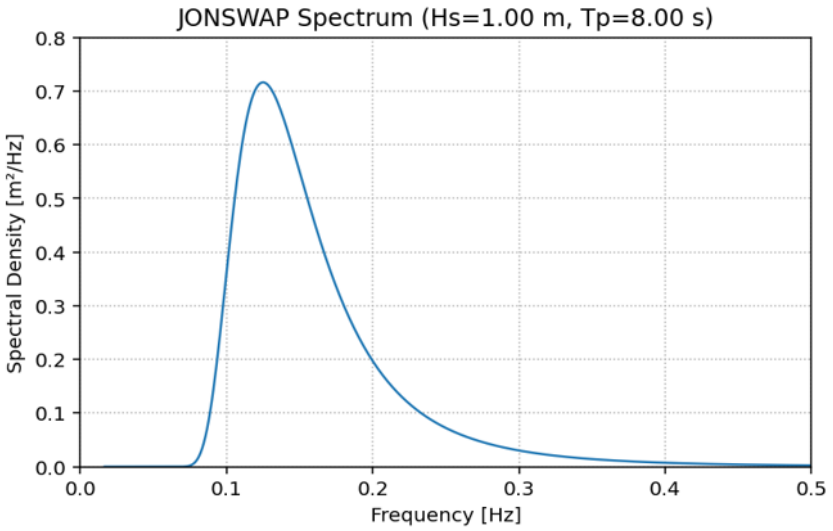


Figure 20: Example JONSWAP spectrum with $H_s = 1$ m and $T_p = 8$ s.

4.7.2 Crane Tip Motion

The second term in the side-load spectrum formulation is the frequency response of the crane tip to wave excitation. This response is obtained using a linear RAO-based method, where the vessel's 6-degree-of-freedom (6DOF) response to unit wave amplitude is defined in the frequency domain for each degree of freedom. These RAOs were precomputed using the hydromechanical software AQWA for the Bokalift 2 vessel and stored in complex form.

Each complex RAO $\hat{u}(\omega)$ is defined as:

$$\hat{u}(\omega) = A(\omega) \cdot e^{-i\phi(\omega)}$$

where $A(\omega)$ is the amplitude and $\phi(\omega)$ is the phase (in radians). These complex RAOs are interpolated over the relevant frequency range using the recorded data for the six global degrees of freedom: Surge, Sway, Heave, Roll, Pitch, and Yaw.

To compute the motion at the crane tip rather than at the vessel center of gravity (CoG), a coordinate transformation is applied. This transformation accounts for the position vector $\vec{P}$ between the vessel reference point and the crane tip:

$$\vec{P} = \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{bmatrix} = \texttt{Cranetip} - \texttt{RAO origin}$$

The transformation matrix $\mathbf{T}$ maps rigid-body vessel motion to local motion at the crane tip:

$$\mathbf{T} = \begin{bmatrix} 1 & 0 & 0 & 0 & P_z & -P_y \\ 0 & 1 & 0 & -P_z & 0 & P_x \\ 0 & 0 & 1 & P_y & -P_x & 0 \end{bmatrix}$$

Applying this transformation to the 6DOF vessel RAO vector gives the tip RAO:

$$\hat{\mathbf{u}}_{tip}(\omega) = \mathbf{T} \cdot \hat{\mathbf{u}}_{vessel}(\omega)$$

Since the side-load calculation only requires the horizontal motion of the crane tip, the x-component is extracted:

$$\hat{u}_{tip,x}(\omega) = [\hat{\mathbf{u}}_{tip}(\omega)]_x$$

This function is complex-valued and varies with frequency. The modulus $|\hat{u}_{tip,x}(\omega)|$ represents the amplitude of horizontal crane tip motion per unit wave amplitude. A typical RAO curve for the Bokalift 2 with wave direction 0° is shown in Figure 21, both in frequency and period representation.

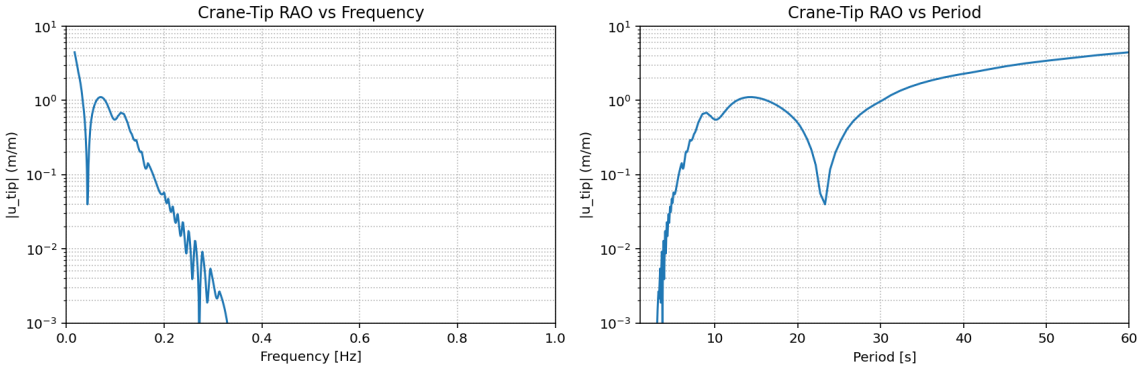


Figure 21: Crane-tip RAO for horizontal (x) motion of the Bokalift 2 with wave direction 0°.

The horizontal tip RAO encapsulates the vessel's dynamic amplification due to wave–structure interaction, and is critical in shaping the resulting side-load spectrum.

4.7.3 $H_{\text{tip}\to\text{load}}$ Transfer Function

The final element in the side-load spectrum expression is the transfer function $H_{\text{tip}\to\text{load}}(\omega)$, which maps horizontal motion at the crane tip to the resulting horizontal force in the rigging. This relationship is derived from the full equations of motion using Lagrangian dynamics, treating the horizontal crane base motion $R(t)$ as a generalized coordinate.

The generalized coordinates are:

$$q = \begin{bmatrix} \theta_1 & \theta_2 & \theta_3 & \theta_4 & R \end{bmatrix}^{\mathsf{T}}$$

Kinetic Energy T The total kinetic energy includes both translational and rotational components for all masses:

$$\begin{aligned} T = {} & \frac{1}{2} M_{bb} \dot{R}^2 + \dot{R}(M_{b1}\dot{\theta}_1 + M_{b2}\dot{\theta}_2 + M_{b3}\dot{\theta}_3 + M_{b4}\dot{\theta}_4) \\ & + \frac{1}{2} I_{p1}\dot{\theta}_1^2 + \frac{1}{2} I_{p2}\dot{\theta}_2^2 + \frac{1}{2} I_{p3}\dot{\theta}_3^2 + \frac{1}{2} I_{p4}\dot{\theta}_4^2 \\ & + I_{12}\dot{\theta}_1\dot{\theta}_2 + I_{13}\dot{\theta}_1\dot{\theta}_3 + I_{14}\dot{\theta}_1\dot{\theta}_4 + I_{23}\dot{\theta}_2\dot{\theta}_3 + I_{24}\dot{\theta}_2\dot{\theta}_4 + I_{34}\dot{\theta}_3\dot{\theta}_4 \end{aligned} \tag{17}$$

Potential Energy V The potential energy is purely gravitational and quadratic under the small-angle approximation:

$$\begin{aligned} V = \frac{1}{2} g \Big(& m_1 L_1 \theta_1^2 + m_2(L_1\theta_1^2 + L_2\theta_2^2) + m_3(L_1\theta_1^2 + L_2\theta_2^2 + d\theta_3^2) \\ & + m_4(L_1\theta_1^2 + L_2\theta_2^2 + d\theta_3^2 + L_4\theta_4^2) \Big) \end{aligned} \tag{18}$$

Lagrangian $\mathcal{L} = T - V$

$$\mathcal{L}(q, \dot{q}) = T - V$$

This Lagrangian forms the basis for deriving the equations of motion via the Euler–Lagrange equations.

Definitions of Mass and Inertia Terms The compact expressions above make use of the following definitions:

- **Total suspended mass:**

$$M_{bb} = m_1 + m_2 + m_3 + m_4$$

- **Base–angle coupling terms:**

$$\begin{aligned} M_{b1} &= (m_1 + m_2 + m_3 + m_4)L_1 \\ M_{b2} &= (m_2 + m_3 + m_4)L_2 \\ M_{b3} &= (m_3 + m_4)d \\ M_{b4} &= m_4 L_4 \end{aligned}$$

- **Effective rotational inertias (including mass distribution):**

$$\begin{aligned} I_{p1} &= I_1 + (m_1 + m_2 + m_3 + m_4)L_1^2 \\ I_{p2} &= I_2 + (m_2 + m_3 + m_4)L_2^2 \\ I_{p3} &= I_3 + I_4 + (m_3 + m_4)d^2 \\ I_{p4} &= m_4 L_4^2 \end{aligned}$$

- **Cross-coupling inertias (off-diagonal):**

$$\begin{aligned} & I_{12} = L_1 L_2(m_2 + m_3 + m_4), \quad I_{13} = L_1 d(m_3 + m_4), \quad I_{14} = L_1 L_4 m_4 \\ & I_{23} = L_2 d(m_3 + m_4), \quad I_{24} = L_2 L_4 m_4, \quad I_{34} = d L_4 m_4 \end{aligned}$$

These terms emerge from a careful grouping of kinetic energy terms involving products of angular velocities and their coupling through geometry and mass distribution.

The resulting Lagrangian $\mathcal{L}$ is used to derive the full set of coupled second-order equations of motion using the Euler-Lagrange equation for each coordinate:

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{q}_i}\right) - \frac{\partial \mathcal{L}}{\partial q_i} = 0$$

This yields the following coupled second-order equations of motion:

$$\theta_1: \quad \ddot{R}M_{b1} + I_{p1}\ddot{\theta}_1 + I_{12}\ddot{\theta}_2 + I_{13}\ddot{\theta}_3 + I_{14}\ddot{\theta}_4 + gL_1\theta_1(m_1 + m_2 + m_3 + m_4) = 0 \tag{19}$$

$$\theta_2: \quad \ddot{R}M_{b2} + I_{12}\ddot{\theta}_1 + I_{p2}\ddot{\theta}_2 + I_{23}\ddot{\theta}_3 + I_{24}\ddot{\theta}_4 + gL_2\theta_2(m_2 + m_3 + m_4) = 0 \tag{20}$$

$$\theta_3: \quad \ddot{R}M_{b3} + I_{13}\ddot{\theta}_1 + I_{23}\ddot{\theta}_2 + I_{p3}\ddot{\theta}_3 + I_{34}\ddot{\theta}_4 + gd\theta_3(m_3 + m_4) = 0 \tag{21}$$

$$\theta_4: \quad \ddot{R}M_{b4} + I_{14}\ddot{\theta}_1 + I_{24}\ddot{\theta}_2 + I_{34}\ddot{\theta}_3 + I_{p4}\ddot{\theta}_4 + gL_4\theta_4 m_4 = 0 \tag{22}$$

Equation of Motion for the Base Coordinate $R(t)$ Unlike the angular coordinates θ_i, the generalized coordinate $R(t)$ is externally actuated via the crane base motion. It contributes to the system's kinetic energy, but does not appear in the potential energy due to the small-angle approximation. As a result, the Euler–Lagrange equation for R includes a nonzero generalized force:

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{R}}\right) - \frac{\partial \mathcal{L}}{\partial R} = F_{\text{tip}} \tag{23}$$

Evaluating the left-hand side yields:

$$M_{bb}\ddot{R} + M_{b1}\ddot{\theta}_1 + M_{b2}\ddot{\theta}_2 + M_{b3}\ddot{\theta}_3 + M_{b4}\ddot{\theta}_4 = F_{\text{tip}} \tag{24}$$

This result can also be interpreted using D'Alembert's principle, where the external force balances the total inertial response of the multibody system projected along the horizontal base direction. Both derivations lead to the same expression.

4.7.4 Frequency-Domain Representation

To obtain the transfer function from crane base motion $R(t)$ to horizontal rigging force F_{tip}, the system of equations derived from the Lagrangian formulation is translated into the frequency domain using a unilateral Laplace transform with $s = i\omega$.

The generalized coordinates $\theta_i(t)$ and $R(t)$ become $\theta_i(s)$ and $R(s)$, with second time derivatives mapping as:

$$\ddot{\theta}_i(t) \to s^2\theta_i(s), \quad \ddot{R}(t) \to s^2R(s)$$

Applying this to the time-domain equations yields the following system:

$$\theta_1: \quad M_{b1}s^2R + I_{p1}s^2\theta_1 + I_{12}s^2\theta_2 + I_{13}s^2\theta_3 + I_{14}s^2\theta_4 + gL_1\theta_1(m_1 + m_2 + m_3 + m_4) = 0 \tag{25}$$

$$\theta_2: \quad M_{b2}s^2R + I_{p2}s^2\theta_2 + I_{12}s^2\theta_1 + I_{23}s^2\theta_3 + I_{24}s^2\theta_4 + gL_2\theta_2(m_2 + m_3 + m_4) = 0 \tag{26}$$

$$\theta_3: \quad M_{b3}s^2R + I_{p3}s^2\theta_3 + I_{13}s^2\theta_1 + I_{23}s^2\theta_2 + I_{34}s^2\theta_4 + gd\theta_3(m_3 + m_4) = 0 \tag{27}$$

$$\theta_4: \quad M_{b4}s^2R + I_{p4}s^2\theta_4 + I_{14}s^2\theta_1 + I_{24}s^2\theta_2 + I_{34}s^2\theta_3 + gL_4\theta_4(m_4) = 0 \tag{28}$$

$$R: \quad M_{bb}s^2R + M_{b1}s^2\theta_1 + M_{b2}s^2\theta_2 + M_{b3}s^2\theta_3 + M_{b4}s^2\theta_4 - F_{\text{tip}}(s) = 0 \tag{29}$$

4.7.5 Matrix Formulation and Transfer Function Derivation

The five Laplace-transformed equations from the previous step are now collected into matrix form. We define a dynamic stiffness matrix $\mathbf{D}(s)$ such that:

$$\underbrace{\begin{bmatrix} \theta_1(s) \\ \theta_2(s) \\ \theta_3(s) \\ \theta_4(s) \\ F_{\text{tip}}(s) \end{bmatrix}}_{\text{State vector}} = \mathbf{D}(s)^{-1} \cdot \left(-s^2 \underbrace{\begin{bmatrix} M_{b1} \\ M_{b2} \\ M_{b3} \\ M_{b4} \\ M_{bb} \end{bmatrix}}_{\text{Base motion coupling vector}} \cdot R(s) \right) \tag{30}$$

The system matrix $\mathbf{D}(s)$ is defined as:

$$\mathbf{D}(s) = \begin{bmatrix} I_{p1}s^2 + gL_1(m_1+m_2+m_3+m_4) & I_{12}s^2 & I_{13}s^2 & I_{14}s^2 & 0 \\ I_{12}s^2 & I_{p2}s^2 + gL_2(m_2+m_3+m_4) & I_{23}s^2 & I_{24}s^2 & 0 \\ I_{13}s^2 & I_{23}s^2 & I_{p3}s^2 + gd(m_3+m_4) & I_{34}s^2 & 0 \\ I_{14}s^2 & I_{24}s^2 & I_{34}s^2 & I_{p4}s^2 + gL_4m_4 & 0 \\ M_{b1}s^2 & M_{b2}s^2 & M_{b3}s^2 & M_{b4}s^2 & -1 \end{bmatrix} \tag{31}$$

This formulation allows solving for both the modal displacements $\theta_i(s)$ and the resulting tip force $F_{\text{tip}}(s)$ directly from any given base motion spectrum $R(s)$.

From this, we isolate the bottom row of the state vector to define the transfer function:

$$H_{\text{tip}\to\text{load}}(s) = \frac{F_{\text{tip}}(s)}{R(s)} = -s^2 \begin{bmatrix} M_{b1} & M_{b2} & M_{b3} & M_{b4} & M_{bb} \end{bmatrix} \cdot \mathbf{D}(s)^{-1} \cdot \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \tag{32}$$

This transfer function can now be used to compute the force spectrum $S_F(\omega)$ resulting from an arbitrary wave-induced motion spectrum $S_\eta(\omega)$, as described in the next subsection.

4.8 Most Probable Maximum Side Load

Once the side-load spectrum $S_F(\omega)$ as defined in 16 is computed, we are interested in estimating the most probable maximum horizontal load experienced during a sea state of given duration. This is a crucial metric for evaluating whether a given lifting configuration remains within allowable limits under operational conditions.

The spectral moment of order zero is defined as:

$$m_0 = \int_0^\infty S_F(\omega)\, d\omega \tag{33}$$

This moment represents the variance of the side load in the time domain. Assuming the side load can be approximated as a zero-mean Gaussian stationary random process, the most probable maximum (MPM) value during a sea state of duration T is estimated as:

$$F_{\max} = \sqrt{2 \ln\left(\frac{T}{T_z}\right)} \cdot \sqrt{m_0} \tag{34}$$

Here:

- T is the total simulation time (e.g., 3 hours),
- T_z is the mean zero-crossing period, approximated as:

$$T_z = \sqrt{\frac{m_0}{m_2}} \tag{35}$$

where m_2 is the second spectral moment:

$$m_2 = \int_0^\infty \omega^2 S_F(\omega)\, d\omega \tag{36}$$

The final result $F_{\max}$ gives the statistically most likely peak force observed over the duration T, not necessarily the absolute maximum. This value can be compared directly to the allowable side load limits for the crane and rigging system.

The MPM method is widely used in offshore engineering for operability assessments and aligns with DNV guidelines for short-term extreme value analysis.

4.9 Grid-Based Optimization

To assess and optimize the dynamic behavior of different rigging configurations under realistic offshore conditions, a grid-based approach is used. This method evaluates the maximum dynamic side load for a discrete set of sea states defined by combinations of significant wave height (H_s) and peak period (T_p).

Each cell in the (H_s, T_p) grid represents a unique sea state for which the side-load spectrum is computed using the full dynamic model described previously. From this spectrum, the most probable maximum (MPM) side load is calculated. By comparing this value to a user-defined limit load (F_{lim}), the operability of the lifting setup can be assessed.

The grid-based format enables systematic evaluation of lifting performance across a representative range of environmental conditions. The grid resolution was chosen based on the available metocean data, specifically the format of wave scatter diagrams commonly used in offshore site assessments. For this study, the grid consists of:

- $H_s \in [0.0, 4.25]$ m with 0.25 m increments (up to 17 bins),
- $T_p \in [2.0, 23.0]$ s with 1.0 s increments (21 bins).

To reduce computational load and focus on practically relevant conditions, the wave height range is optionally truncated to $H_s \leq 2.5$ m, reflecting common operational limits for monopile lifting activities.

For each sea state:

1. The side-load spectrum is computed from the JONSWAP wave spectrum, vessel RAO, and dynamic transfer function $H_{\text{tip}\to\text{load}}$.
2. The most probable maximum side load is calculated over a fixed duration (e.g., 3 hours).
3. This load is compared to a threshold value F_{lim} to determine operability.
4. The binary result is weighted by the probability of occurrence for the given sea state (from a monthly scatter diagram).

The total workability is then expressed as the sum of weighted operable states in the grid. This workability score serves as the objective function in the rigging optimization procedure, which seeks to maximize operational uptime by tuning geometric parameters in the rigging layout.

4.9.1 MPM Calculation

For each sea state combination (H_s, T_p), the most probable maximum (MPM) side load $F_{\max}$ is computed from the side-load spectrum $S_F(f)$, using the statistical formulation introduced in Section 4.8.

This process relies on the spectral moments m_0 and m_2, defined in Equations (33) and (36). From these, the zero-crossing period T_z is derived using Equation (35), and the MPM is computed using Equation (34).

This calculation is performed for all sea states in the (H_s, T_p) grid, resulting in a matrix of expected peak loads across the relevant offshore conditions.

Hs/Tp	0.00-0.25	0.25-0.50	0.50-0.75	0.75-1.00	1.00-1.25	1.25-1.50	1.50-1.75	1.75-2.00	2.00-2.25	2.25-2.50
2.0-3.0	0,053659	0,137977	0,205429	0,287601	0,369772	0,451944	0,534116	0,616287	0,698459	0,78063
3.0-4.0	0,494818	1,484453	2,263325	2,751033	3,41948	4,179365	4,939249	5,699134	6,459018	7,218903
4.0-5.0	3,039163	9,117489	15,19582	20,79293	24,64168	28,09398	31,75561	36,64109	41,52657	46,41204
5.0-6.0	20,58392	61,75175	102,9196	144,0874	185,2553	215,9224	238,4492	258,5845	277,0597	296,7493
6.0-7.0	57,43266	172,298	287,1633	402,0286	516,894	631,7593	746,6246	832,6845	902,5897	966,9637
7.0-8.0	83,90194	251,7058	419,5097	587,3136	755,1175	922,9214	1090,725	1258,529	1426,333	1624,374
8.0-9.0	92,5007	277,5021	462,5035	647,5049	832,5063	1017,508	1202,509	1387,511	1572,512	1757,513
9.0-10.0	90,06038	270,1811	450,3019	630,4227	810,5434	990,6642	1170,785	1350,906	1531,026	1711,147
10.0-11.0	83,05017	249,1505	415,2509	581,3512	747,4515	913,5519	1079,652	1245,753	1411,853	1577,953
11.0-12.0	75,14268	225,428	375,7134	525,9987	676,2841	826,5695	976,8548	1127,14	1277,426	1427,711
12.0-13.0	68,22276	204,6683	341,1138	477,5593	614,0048	750,4503	886,8958	1023,341	1159,787	1296,232
13.0-14.0	63,50024	190,5007	317,5012	444,5017	571,5022	698,5027	825,5032	952,5037	1079,504	1206,505
14.0-15.0	61,8211	185,4633	309,1055	432,7477	556,3899	680,0322	803,6744	927,3166	1050,959	1174,601
15.0-16.0	63,01323	189,0397	315,0661	441,0926	567,119	693,1455	819,1719	945,1984	1071,225	1197,251
16.0-17.0	65,78351	197,3505	328,9175	460,4846	592,0516	723,6186	855,1856	986,7526	1118,32	1249,887
17.0-18.0	68,67792	206,0338	343,3896	480,7455	618,1013	755,4572	892,813	1030,169	1167,525	1304,881
18.0-19.0	70,78821	212,3646	353,941	495,5174	637,0938	778,6703	920,2467	1061,823	1203,399	1344,976
19.0-20.0	71,78427	215,3528	358,9213	502,4899	646,0584	789,6269	933,1955	1076,764	1220,333	1363,901
20.0-21.0	71,69353	215,0806	358,4677	501,8547	645,2418	788,6289	932,016	1075,403	1218,79	1362,177
21.0-22.0	70,70363	212,1109	353,5181	494,9254	636,3327	777,7399	919,1472	1060,554	1201,962	1343,369
22.0-23.0	69,04394	207,1318	345,2197	483,3076	621,3955	759,4834	897,5713	1035,659	1173,747	1311,835

Figure 22: Grid of most probable maximum side loads $F_{\max}$ [kN] for the nominal rigging configuration.

The resulting grid illustrates how the dynamic side load increases with significant wave height H_s and varies with peak period T_p, highlighting the sensitivity of the suspended system to sea state characteristics. This matrix forms the input for the limit-checking step.

4.9.2 Limit Check

After computing the matrix of most probable maximum (MPM) side loads, each value $F_{\max}$ is evaluated against a threshold limit load $F_{\lim}$. This threshold determines whether a given sea state in the (H_s, T_p) grid is considered operationally safe for lifting.

In this study, the threshold is defined as:

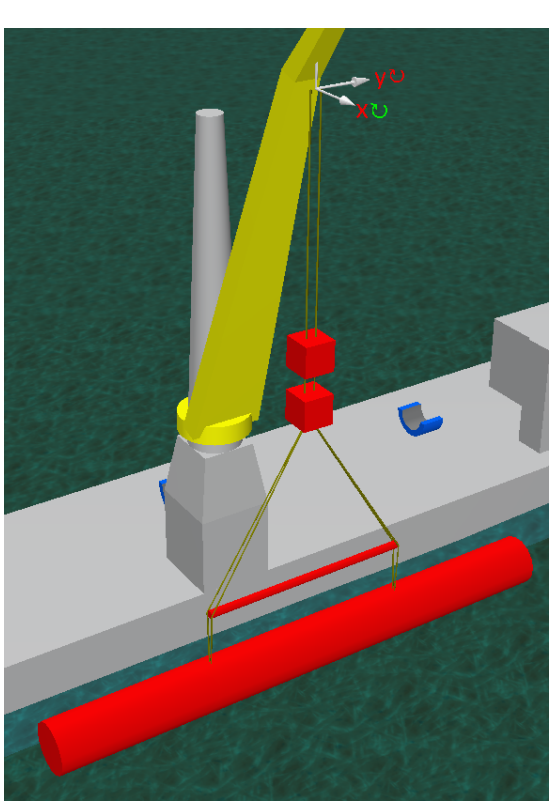
$$F_{\lim} = 0.57 \cdot 1200 \text{ kN} = 684 \text{ kN} \tag{37}$$

The nominal sideload capacity of the crane and rigging system is specified as 1200 kN in the operational documentation of the Bokalift 2. However, the analytical model consistently underestimates peak side loads when compared to high-fidelity frequency-domain simulations performed in OrcaFlex. These OrcaFlex simulations include fully detailed rigging elements such as grommets, stretchable slings, and realistic damping behavior.

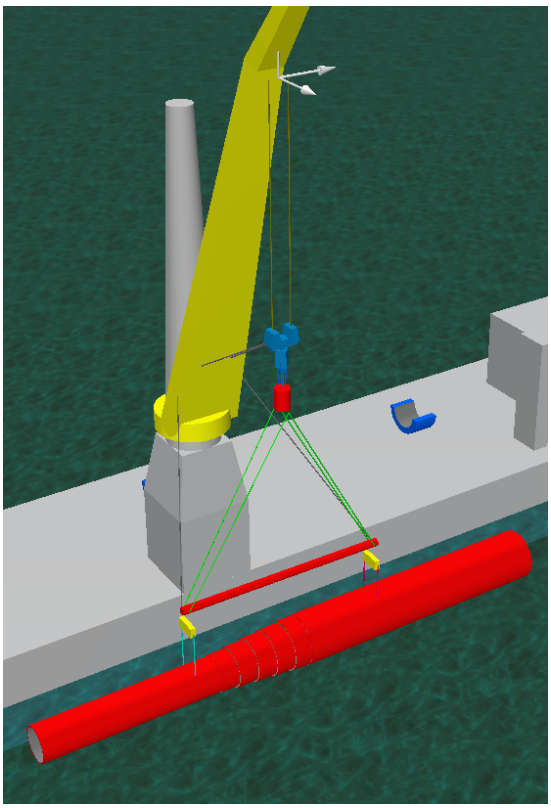
To illustrate this discrepancy, two OrcaFlex models were constructed:

- An **advanced model** featuring detailed rigging elements,
- A **simplified model** using inextensible cables and assumptions aligned with the analytical model.

The difference in predicted MPM loads between these models is shown in Figure 24, where each cell denotes the percentage difference across the same (H_s, T_p) grid. On average, the simplified model underestimates loads by approximately 43%.



(a) Simplified OrcaFlex model



(b) Advanced OrcaFlex model

Figure 23: Comparison of sideload prediction between simplified and advanced OrcaFlex models.

Hs/Tp	0,125	0,375	0,625	0,875	1,125	1,375	1,625	1,875	2,125	2,375
2,5										
3,5										
4,5	16%	16%	16%							
5,5	20%	20%	20%	20%	20%					
6,5	56%	56%	56%	56%	56%	56%	56%			
7,5	48%	48%	48%	48%	48%	48%	48%	48%	48%	
8,5	46%	46%	46%	46%	46%	46%	46%	46%	46%	46%
9,5	44%	44%	44%	44%	44%	44%	44%	44%	44%	44%
10,5	43%	43%	43%	43%	43%	43%	43%	43%	43%	43%
11,5	42%	42%	42%	42%	42%	42%	42%	42%	42%	42%
12,5	42%	42%	42%	42%	42%	42%	42%	42%	42%	42%
13,5	41%	41%	41%	41%	41%	41%	41%	41%	41%	41%
14,5	40%	40%	40%	40%	40%	40%	40%	40%	40%	40%
15,5	39%	39%	39%	39%	39%	39%	39%	39%	39%	39%
16,5	41%	41%	41%	41%	41%	41%	41%	41%	41%	41%
17,5	42%	42%	42%	42%	42%	42%	42%	42%	42%	42%
18,5	43%	43%	43%	43%	43%	43%	43%	43%	43%	43%
19,5	43%	43%	43%	43%	43%	43%	43%	43%	43%	43%
20,5	43%	43%	43%	43%	43%	43%	43%	43%	43%	43%
21,5	43%	43%	43%	43%	43%	43%	43%	43%	43%	43%
22,5	43%	43%	43%	43%	43%	43%	43%	43%	43%	43%

Figure 24: Relative difference between simplified and advanced OrcaFlex models, showing consistent underprediction by the simplified model.

The discrepancy is primarily attributed to differences in damping and stiffness modeling between the analytical approach and OrcaFlex:

- **Damping treatment:** OrcaFlex applies damping only to `Line` objects (Orcina Ltd, 2024). In the **advanced model**, only the hoisting cable L_1 is modeled as a `Line`, while the remaining rigging elements (L2, L3, L4) are modeled as grommets and slings without damping. In contrast, the **simplified model** uses `Line` objects for all rigging connections, so damping is applied throughout the system — more in line with the Rayleigh damping approach used in the analytical model.
- **Cable stiffness:** In the advanced model, rigging elements have finite stiffness values based on their actual construction (e.g., braided grommets), which introduces flexibility into the system.

The simplified OrcaFlex model, like the analytical model, uses effectively infinite stiffness for all cables, removing elastic deformation from the response.

To reconcile the difference while maintaining the computational efficiency of the analytical model, the threshold limit was conservatively reduced by 43%. This scaling ensures that workability predictions from the simplified model remain consistent with safety margins implied by the advanced simulations.

While this adjustment is empirical and not derived from design codes (e.g., DNV-RP-C205), it offers a pragmatic compromise. The choice allows meaningful optimization and comparison of rigging configurations, while acknowledging the limitations of the simplified model. A more rigorous treatment — e.g., based on full nonlinear time-domain simulations or physical measurements — is recommended for detailed engineering assessments beyond the scope of this thesis.

Each grid cell is assigned a binary value according to:

$$\text{Unity}_{i,j} = \begin{cases} 1, & \text{if } F_{\max}(H_{s_i}, T_{p_j}) \leq F_{\lim} \\ 0, & \text{otherwise} \end{cases}$$

This produces a binary operability matrix, shown in Figure 25. Green cells indicate acceptable sea states where the side load remains below the limit, while red cells represent conditions where the load threshold is exceeded and operations must be postponed.

Hs/Tp	0.00-0.25	0.25-0.50	0.50-0.75	0.75-1.00	1.00-1.25	1.25-1.50	1.50-1.75	1.75-2.00	2.00-2.25	2.25-2.50
2.0-3.0	1	1	1	1	1	1	1	1	1	1
3.0-4.0	1	1	1	1	1	1	1	1	1	1
4.0-5.0	1	1	1	1	1	1	1	1	1	1
5.0-6.0	1	1	1	1	1	1	1	1	1	1
6.0-7.0	1	1	1	1	1	1	0	0	0	0
7.0-8.0	1	1	1	1	0	0	0	0	0	0
8.0-9.0	1	1	1	1	0	0	0	0	0	0
9.0-10.0	1	1	1	1	0	0	0	0	0	0
10.0-11.0	1	1	1	1	0	0	0	0	0	0
11.0-12.0	1	1	1	1	1	0	0	0	0	0
12.0-13.0	1	1	1	1	1	0	0	0	0	0
13.0-14.0	1	1	1	1	1	0	0	0	0	0
14.0-15.0	1	1	1	1	1	1	0	0	0	0
15.0-16.0	1	1	1	1	1	0	0	0	0	0
16.0-17.0	1	1	1	1	1	0	0	0	0	0
17.0-18.0	1	1	1	1	1	0	0	0	0	0
18.0-19.0	1	1	1	1	1	0	0	0	0	0
19.0-20.0	1	1	1	1	1	0	0	0	0	0
20.0-21.0	1	1	1	1	1	0	0	0	0	0
21.0-22.0	1	1	1	1	1	0	0	0	0	0
22.0-23.0	1	1	1	1	1	0	0	0	0	0

Figure 25: Binary limit-check result for all sea states. A value of 1 (green) indicates operational feasibility, 0 (red) indicates exceedance of the load limit.

This matrix forms the basis for the weighted workability calculation in the next stage.

4.9.3 Weighting and Summation

Once the binary operability matrix is constructed, each cell is weighted by the probability of occurrence of its corresponding sea state. These probabilities are obtained from metocean scatter diagrams specific to the offshore site and time of year. In this study, the month of August is used as an illustrative example.

The wave climate is characterized by a joint probability distribution over H_s and T_p, with each bin associated with a midpoint probability P_{ij}. The binary feasibility value $u_{ij} \in \{0, 1\}$ indicates whether the operation is allowed for that sea state. The weighted operability for each cell is given by:

$$W_{ij} = u_{ij} \cdot P_{ij} \tag{38}$$

Summing over the full matrix yields the total expected workability:

$$\text{Workability} = \sum_{i,j} W_{ij} \tag{39}$$

For the nominal rigging configuration and wave statistics for August, the overall workability evaluates to:

$$\text{Workability} = 79.92\%$$

Hs/Tp	0.0-0.25	0.25-0.5	0.5-0.75	0.75-1.0	1.0-1.25	1.25-1.5	1.5-1.75	1.75-2.0	2.0-2.25	2.25-2.5	total
2.0-3.0	0,00%	0,02%	0,07%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,09%
3.0-4.0	0,00%	0,35%	2,48%	1,75%	0,12%	0,00%	0,00%	0,00%	0,00%	0,00%	4,71%
4.0-5.0	0,00%	0,13%	2,45%	4,93%	3,76%	0,91%	0,15%	0,02%	0,00%	0,00%	12,34%
5.0-6.0	0,00%	0,44%	3,89%	4,44%	3,62%	2,58%	1,30%	0,43%	0,12%	0,04%	16,85%
6.0-7.0	0,00%	0,70%	5,93%	5,59%	3,31%	1,93%	1,38%	1,00%	0,58%	0,30%	20,71%
7.0-8.0	0,00%	0,62%	4,03%	5,16%	3,14%	1,33%	0,66%	0,50%	0,36%	0,29%	16,09%
8.0-9.0	0,01%	0,86%	1,91%	1,70%	1,94%	1,08%	0,50%	0,31%	0,13%	0,06%	8,51%
9.0-10.0	0,00%	0,85%	2,31%	1,07%	0,81%	0,48%	0,22%	0,16%	0,04%	0,03%	5,96%
10.0-11.0	0,00%	0,43%	2,17%	1,05%	0,51%	0,32%	0,23%	0,13%	0,08%	0,04%	4,96%
11.0-12.0	0,00%	0,07%	0,55%	0,33%	0,18%	0,10%	0,05%	0,08%	0,06%	0,04%	1,45%
12.0-13.0	0,00%	0,26%	0,69%	0,62%	0,25%	0,12%	0,05%	0,10%	0,12%	0,05%	2,27%
13.0-14.0	0,00%	0,13%	0,32%	0,15%	0,18%	0,11%	0,02%	0,01%	0,02%	0,02%	0,97%
14.0-15.0	0,00%	0,22%	0,59%	0,29%	0,23%	0,04%	0,06%	0,04%	0,03%	0,06%	1,58%
15.0-16.0	0,00%	0,02%	0,06%	0,06%	0,07%	0,02%	0,04%	0,07%	0,04%	0,00%	0,38%
16.0-17.0	0,00%	0,03%	0,23%	0,22%	0,05%	0,02%	0,04%	0,03%	0,05%	0,02%	0,70%
17.0-18.0	0,00%	0,01%	0,02%	0,04%	0,00%	0,02%	0,01%	0,01%	0,00%	0,03%	0,13%
18.0-19.0	0,00%	0,07%	0,08%	0,08%	0,00%	0,00%	0,02%	0,03%	0,01%	0,01%	0,31%
19.0-20.0	0,00%	0,00%	0,02%	0,02%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,04%
20.0-21.0	0,00%	0,00%	0,01%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,01%
21.0-22.0	0,00%	0,00%	0,04%	0,03%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,08%
22.0-23.0	0,00%	0,00%	0,01%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,01%
total	0,01%	5,22%	27,86%	27,54%	18,19%	9,07%	4,73%	2,90%	1,65%	0,98%	100,00%

(a) Joint wave probability distribution for August.

Hs/Tp	0.00-0.25	0.25-0.50	0.50-0.75	0.75-1.00	1.00-1.25	1.25-1.50	1.50-1.75	1.75-2.00	2.00-2.25	2.25-2.50	total
2.0-3.0	0	0,023522	0,070565	0	0	0	0	0	0	0	0,094086
3.0-4.0	0	0,349462	2,483199	1,754032	0,124328	0	0	0	0	0	4,711022
4.0-5.0	0	0,127688	2,446237	4,932796	3,75672	0,910618	0,147849	0,016801	0	0	12,33871
5.0-6.0	0	0,443548	3,887769	4,438844	3,618952	2,580645	1,297043	0,430108	0,120968	0,036962	16,85484
6.0-7.0	0	0,695565	5,927419	5,588038	3,313172	1,928763	0	0	0	0	17,45296
7.0-8.0	0	0,62164	4,028898	5,16129	0	0	0	0	0	0	9,811828
8.0-9.0	0,00672	0,863575	1,911962	1,703629	0	0	0	0	0	0	4,485887
9.0-10.0	0	0,850134	2,308468	1,065188	0	0	0	0	0	0	4,22379
10.0-11.0	0	0,430108	2,174059	1,051747	0	0	0	0	0	0	3,655914
11.0-12.0	0	0,070565	0,554435	0,325941	0,181452	0	0	0	0	0	1,132392
12.0-13.0	0	0,258737	0,688844	0,61828	0,252016	0	0	0	0	0	1,817876
13.0-14.0	0	0,134409	0,31922	0,15457	0,181452	0	0	0	0	0	0,789651
14.0-15.0	0	0,221774	0,591398	0,292339	0,231855	0,040323	0	0	0	0	1,377688
15.0-16.0	0	0,023522	0,063844	0,057124	0,070565	0	0	0	0	0	0,215054
16.0-17.0	0	0,026882	0,225134	0,221774	0,053763	0	0	0	0	0	0,527554
17.0-18.0	0	0,00672	0,020161	0,036962	0	0	0	0	0	0	0,063844
18.0-19.0	0	0,070565	0,084005	0,080645	0,00336	0	0	0	0	0	0,238575
19.0-20.0	0	0	0,016801	0,023522	0	0	0	0	0	0	0,040323
20.0-21.0	0	0	0,00672	0	0	0	0	0	0	0	0,00672
21.0-22.0	0	0	0,043683	0,033602	0	0	0	0	0	0	0,077285
22.0-23.0	0	0	0,00672	0	0	0	0	0	0	0	0,00672
total	0,00672	5,218414	27,85954	27,54032	11,78763	5,460349	1,444892	0,446909	0,120968	0,036962	79,92272

(b) Resulting weighted workability matrix.

Figure 26: The joint probability (a) is multiplied by the binary operability matrix to obtain the weighted workability grid (b).

This formulation integrates both physical feasibility and environmental likelihood, enabling robust comparisons of different rigging configurations based on expected operational uptime.

4.10 Optimization Method

To maximize the offshore workability of the lifting configuration, we use a numerical optimization method to adjust the rigging geometry. The goal is to find the optimal combination of lengths L_1, L_2, L_3 (and consequently L_4) that yields the highest weighted uptime, while satisfying geometric and operational constraints.

Chosen Variables and Fixed Quantities The optimization considers only the rigging *lengths* as free variables. Physical properties such as mass and inertia are held constant for each element. This decision reflects both practical and modeling reasons:

- The masses and inertias are tied to specific rigging hardware and load configuration, and are not easily altered in practice.
- Variations in dynamic response due to geometry (lever arms, coupling) are much more significant than marginal effects from mass variation.
- The inertia terms appear linearly in the equations of motion and are recalculated symbolically based on geometry, making the length parameters sufficient for dynamic tuning.

The spreader bar width w_3 is kept fixed during the optimization. This decision is grounded in engineering practicality: modifying the spreader geometry would require the design and fabrication of a new structural element, introducing additional complexity and cost. Furthermore, internal testing has shown that variations in w_3 lead to only marginal improvements in total workability—gains that can largely be achieved through adjustment of cable lengths instead. This makes the spreader width a logical and effective parameter to constrain.

Constant Total Rigging Length Rather than allowing all lengths to vary independently, the total vertical rigging length L_{total} is held constant throughout the optimization:

$$L_{\text{total}} = L_1 + L_2 + d + L_4$$

where $d = \sqrt{L_3^2 - (w_3/2)^2}$ is the horizontal offset below the lower sling triangle. The remaining degree of freedom L_4 is then determined implicitly.

This constraint reflects the operational reality that lifting height requirements are fixed by the vessel and pile positions at lift-off. In this research, the focus is on the *lift-off* phase of the operation — a known bottleneck based on the problem analysis — and the total geometry must remain compatible with that load case. The constraint also has a practical side effect: it allows optimization across different lift heights by treating L_{total} as a parameter to sweep externally, without altering the optimization mechanism.

Rigging Angle Constraint To ensure a physically feasible and structurally sound configuration, the optimization enforces a constraint on the included rigging angle δ, which is defined as the angle between the two lower slings of length L_3. This is illustrated in Figure 18, where the horizontal spreader bar and its connection to the lower slings form the characteristic sling triangle.

This angle δ is computed as:

$$\delta = 2 \cdot \arcsin\left(\frac{w_3}{2L_3}\right)$$

Extremely flat or steep angles are undesirable in offshore lifting operations. Flat slings result in high tensile forces and increased risk of sling overload, while steep slings reduce horizontal stiffness and lateral control. To maintain operational reliability, we impose a constraint:

$$\delta \in [60^\circ, 90^\circ]$$

This is enforced in the optimizer by constraining the geometric ratio $\frac{w_3}{2L_3}$ to lie between $\sin(30^\circ)$ and $\sin(45^\circ)$, ensuring that δ remains within acceptable physical bounds.

Optimization Method: Differential Evolution To navigate the complex, nonlinear landscape of the objective function, we use **differential evolution** (DE) from the `scipy` package. DE is a global, population-based evolutionary algorithm that does not require gradient information and is robust against local minima — making it well-suited for problems involving discontinuities, coupling, and multi-modal behavior.

The objective function is defined as the negative total workability:

$$\text{Objective}(\mathbf{x}) = -\text{Workability} \tag{40}$$

Each candidate solution is evaluated by:

1. Updating the rigging geometry and recalculating mass and stiffness matrices M, K,
2. Recomputing the damping matrix C,
3. Generating the transfer function $H_{\text{tip}\to\text{load}}$,

4. Evaluating the MPM loads for all (H_s, T_p) combinations,
5. Applying the limit check and weighting by the monthly wave scatter data,
6. Summing all contributions to obtain the total workability.

A population size of 20, 50 iterations, and a fixed seed ensure repeatability. The optimizer converges reliably for this problem, and post-optimization results are cross-validated by rerunning the full model on the optimal configuration.

All intermediate results, including updated modal frequencies, optimized load grids, and workability improvements, are stored and exported for comparison.

5 Model Validation

To ensure the correctness and applicability of the developed dynamic model, a series of validation steps are performed. These aim to compare the output of the analytical Python-based formulation with reference results obtained from Boskalis' industry-standard simulation software, OrcaFlex. By validating multiple aspects of the model, confidence is gained in its ability to accurately capture the key dynamic phenomena involved in offshore monopile lifting.

All validation steps are performed using the nominal rigging parameters, as currently used in operations and defined in Section 4.5. The wave spectrum follows a standard JONSWAP formulation with wave direction $\beta = 180°$, significant wave height $H_s = 1.0$ m, and peak period $T_p = 8.0$ s.

The following elements of the model are validated:

- Natural frequencies (modal analysis),
- Crane tip response amplitude operator (RAO),
- Motion-to-load transfer function $H_{\text{tip}\to\text{load}}(\omega)$,
- Frequency-domain side-load spectrum,
- Grid representation based on most probable maximum load estimates.

Each validation step compares model outputs quantitatively and visually where possible, with explanation of any observed deviations and their physical or numerical origins. The goal is not perfect matching, but confirmation that the model reliably captures the dominant behaviors and dynamic sensitivities relevant for lifting performance analysis.

5.1 Modal Analysis

The first step in validating the dynamic model is a comparison of its natural frequencies with those produced by OrcaFlex. This step is critical because the modal periods in the Python model result directly from the structure and values of the mass and stiffness matrices derived via the Lagrangian formulation. A mismatch in these periods would indicate an error in the foundational equations of motion or in the physical parameterization of the system.

Modal periods in the Python model are computed by solving the generalized eigenvalue problem:

$$\mathbf{K}\phi_i = \omega_i^2 \mathbf{M}\phi_i, \qquad T_i = \frac{2\pi}{\omega_i}$$

The same rigging configuration is modeled in OrcaFlex using the static calculation feature, and the modal periods of the rigging are extracted. The results are presented in Table 3.

Table 3: Comparison of modal periods between Python model and OrcaFlex (nominal configuration).

Mode	**Python Model [s]**	**OrcaFlex [s]**
Mode 1	20.57	20.55
Mode 2	8.16	8.16
Mode 3	3.00	3.05
Mode 4	1.01	1.02

The excellent agreement between the two sets of results confirms the correctness of the derived mass ($\mathbf{M}$) and stiffness ($\mathbf{K}$) matrices. This gives confidence in the structural fidelity of the analytical model and forms a solid basis for the subsequent dynamic simulations.

5.2 Crane Tip RAO

To further validate the implementation of vessel motion input into the analytical model, the horizontal crane tip RAO is compared between the Python implementation and the OrcaFlex frequency-domain simulation.

The calculation of the tip RAO in Python is described in Section 4.7.2. In short, the 6DOF RAOs for the vessel's center of gravity are loaded from AQWA output and translated to the crane tip location

via rigid-body transformation. This transformation uses the position vector between the vessel reference point and the crane tip, ensuring that the rotational effects are properly included in the local motion.

The horizontal tip RAO component $u_{\text{tip},x}(\omega)$ is extracted and plotted over frequency.

In OrcaFlex, a dynamic frequency-domain simulation is used to extract the RAO of the crane tip in the vessel-fixed frame. Specifically, the response component `MH Crane Tip In-frame X` (m/m) is used as the horizontal motion of interest. This response is exported and compared directly to the Python result.

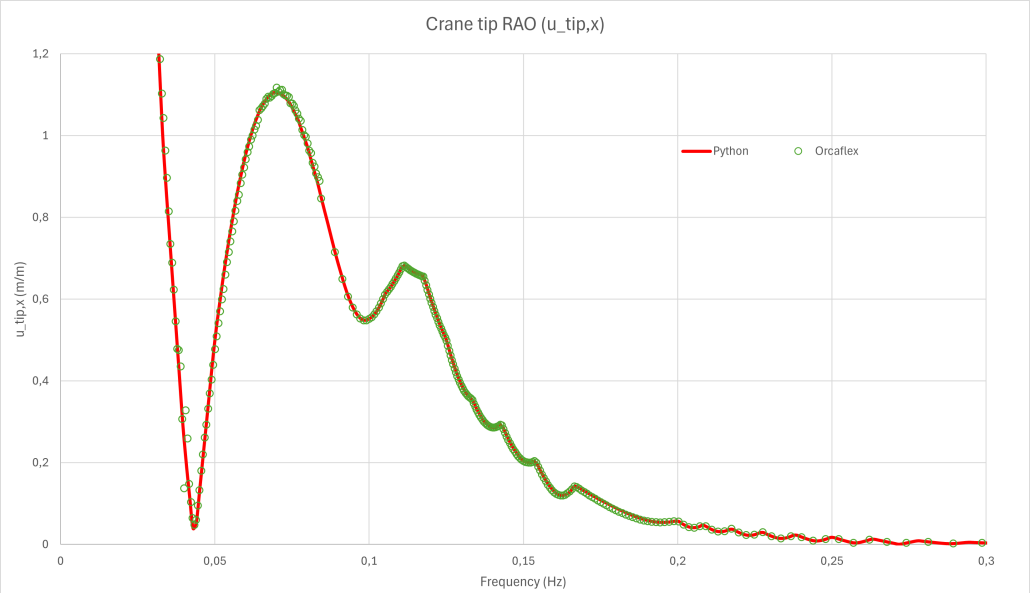


Figure 27: Comparison of horizontal crane tip RAO $u_{\text{tip},x}(\omega)$ between Python model and OrcaFlex.

As shown in Figure 27, the Python and OrcaFlex results align nearly perfectly across the full frequency range. This confirms the correct interpolation of RAO data, the rigid-body transformation, and the extraction of the correct horizontal response component in the Python implementation.

It can also be observed that for frequencies higher than approximately 0.15 Hz, the crane tip motion becomes negligible. This is consistent with physical expectations: at these higher frequencies, the inertia of the Bokalift 2 dominates the response, and wave excitations are unable to induce significant motion at the crane tip.

5.3 Transfer Function $H_{\text{tip}\to\text{load}}$

The third validation step compares the dynamic transfer function from crane tip motion to side load, denoted $H_{\text{tip}\to\text{load}}(\omega)$, between the Python implementation and the OrcaFlex model.

This function is central to the frequency-domain formulation of the side-load spectrum, as described in Section 5.3. It quantifies the frequency-dependent force response at the crane tip due to horizontal excitation $R(t)$, and incorporates full dynamic coupling between the pendulum bodies.

Python Implementation In Python, this transfer function is defined explicitly from the Lagrangian system via:

$$H_{\text{tip}\to\text{load}}(s) = \frac{\mathrm{F}_{\text{tip,x}}(s)}{\mathrm{R}(s)} = -s^2 \begin{bmatrix} M_{b1} & M_{b2} & M_{b3} & M_{b4} & M_{bb} \end{bmatrix} \cdot \mathbf{D}(s)^{-1} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

with $s = i\omega$ and $\mathbf{D}(s)$ the dynamic matrix of the coupled system as is shown in 5.3. The function is evaluated in Python for a frequency range up to 1.0 Hz.

OrcaFlex Evaluation In OrcaFlex, the transfer function is derived by extracting the RAO for horizontal force at the crane tip and dividing it by the RAO for horizontal motion:

$$H_{\text{tip}\to\text{load}}(\omega) = \frac{\text{RAO}_{\text{Force}}(\omega)}{\text{RAO}_{\text{Motion}}(\omega)}$$

Specifically, we use:

- `MH Crane Tip In-frame connection Ly force (kN/m)` — corresponding to horizontal side force in the local crane tip frame.
- `MH Crane Tip In-frame X (m/m)` — corresponding to horizontal tip motion.

This ratio directly provides the frequency-domain amplification factor from motion to force.

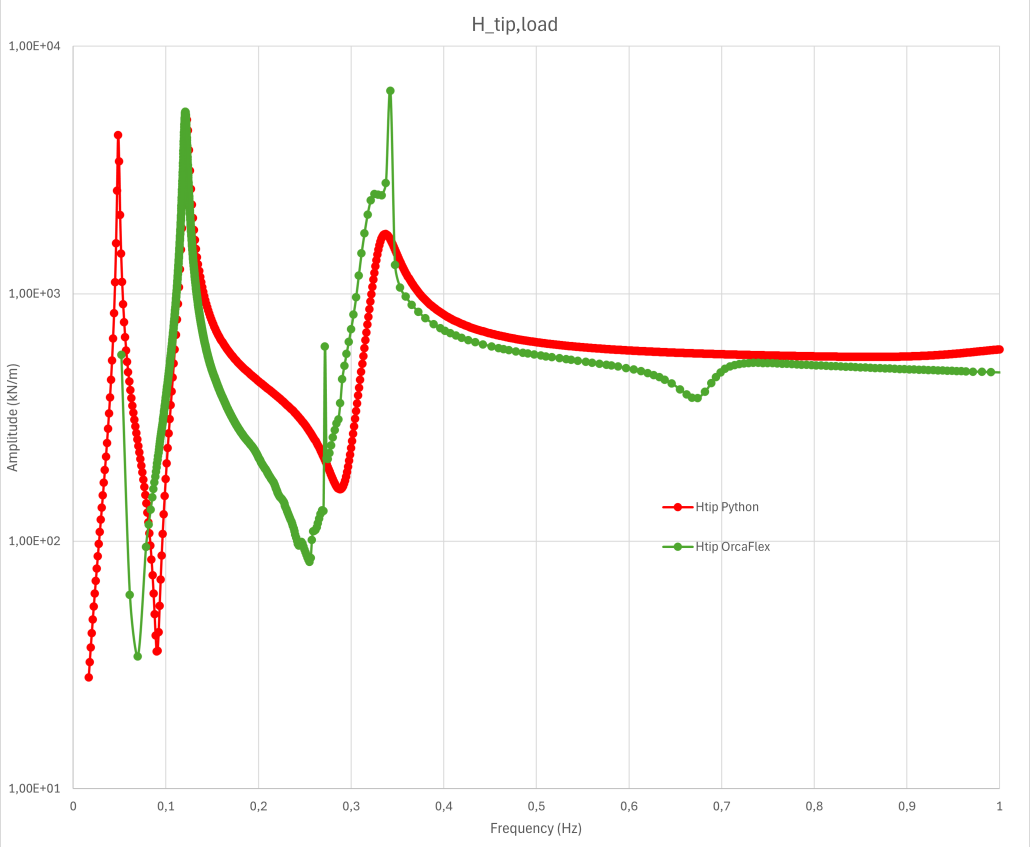


Figure 28: Comparison of transfer function $H_{\text{tip}\to\text{load}}(\omega)$ between Python and OrcaFlex.

Comparison and Observations As shown in Figure 28, both models exhibit peaks at identical frequencies, confirming that the dynamic coupling and natural frequencies are correctly captured. However, there is a noticeable discrepancy in the amplitude of the transfer function between the two implementations.

Several explanations may contribute to this difference:

- Structural flexibility present in OrcaFlex but not included in the simplified analytical model. The analytical model is based on constraints built into the Lagrangian; there is no axial stiffness or cable weight. In Orcaflex the axial stiffness is chosen to be as high as possible, but model stability prevents the use of massless, infinitely stiff cables.

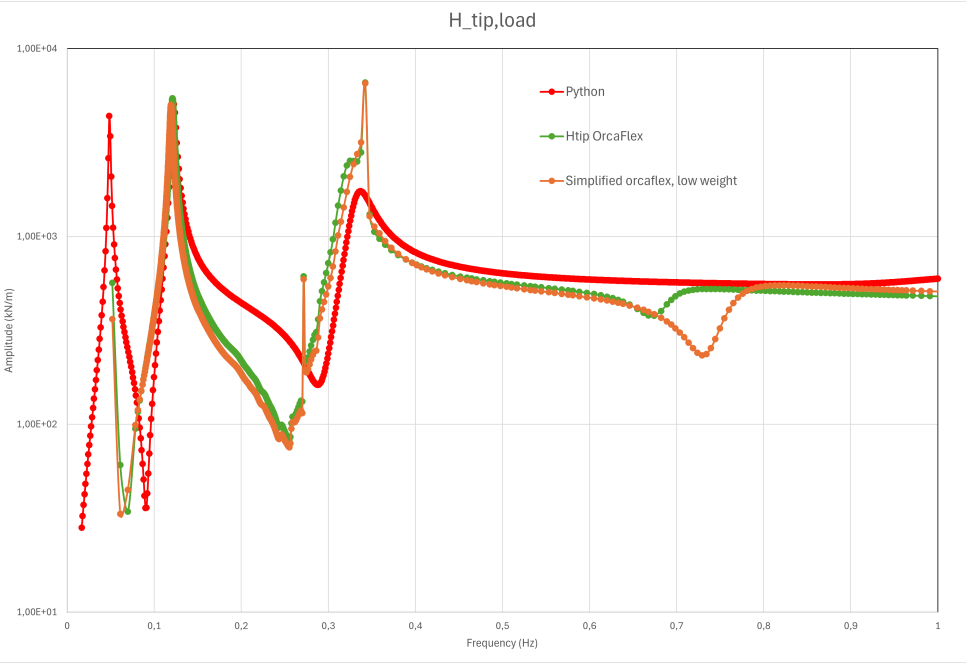


Figure 29: OrcaFlex transferfunction with low weigth

When the mass the cables is decreased heavily to 100e-6 te/m (0 is not possible in OrcaFlex) and the axial stiffness is increased, we can observe that the impact on the transferfunction is limited but the resonant frequencies shift slightly. This is mainly due to the stretching of the cables that influence the total length of the system

- The local axis system at the crane tip exhibits 3D motion due to vessel motions; this effect is very small but discrepancies between the global horizontal crane load and the local horizontal crane load have been observed. These effects are captured in the transfer function produced by OrcaFlex, but not by the transfer function produced by Python due to the assumption of planar motion.

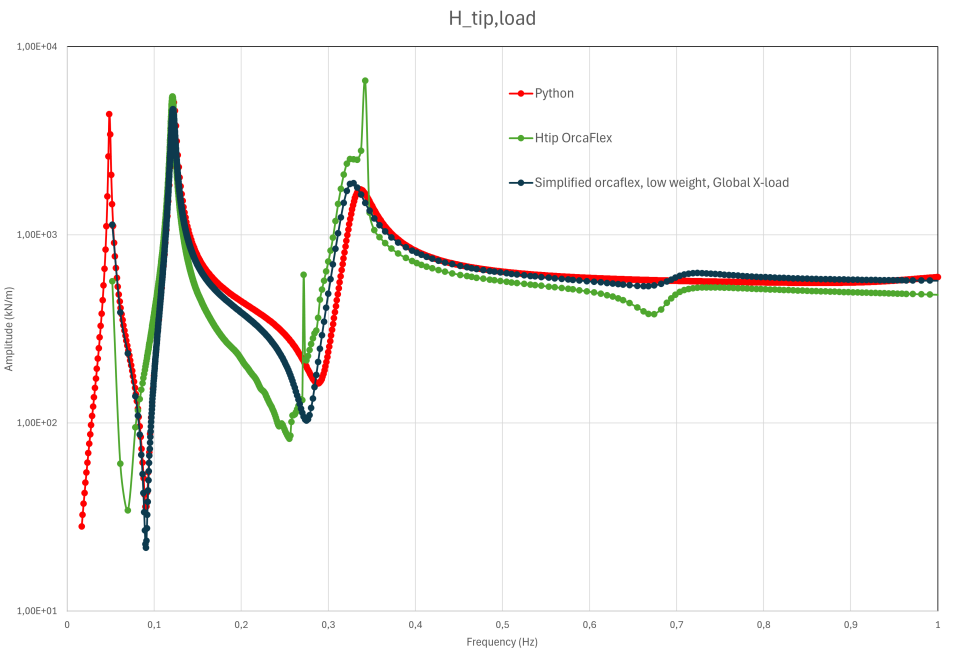


Figure 30: OrcaFlex transferfunction analyzing the side load in global X-direction

- The last difference between the OrcaFlex model and the python model that is responsible for the transferfunction discrepancy is the attachment of the cables to the massblocks. In Python, the model is build up by point masses with inertia that are connected through constraints fundamental

to the Lagrangian approach. In Python however it is impossible to connect the the cables exactly to the CoG of the massblocks due to the stability of the model. There needs to be a distance between connections for it to reach an equilibrium and run a stable dynamic analysis. This problem acculumates at the L2 connection the model gets highly unstable if not give a significant spacing between the CoG and the cable attachment. For this reason the choice has been made to increase the L2 length in Python (from L2= 5,5m to L2 = 9,5m)to account for this spacing in OrcaFlex, which is 2 m at mass block 2 and 2m and mass block 3. This results in the transferfunction in

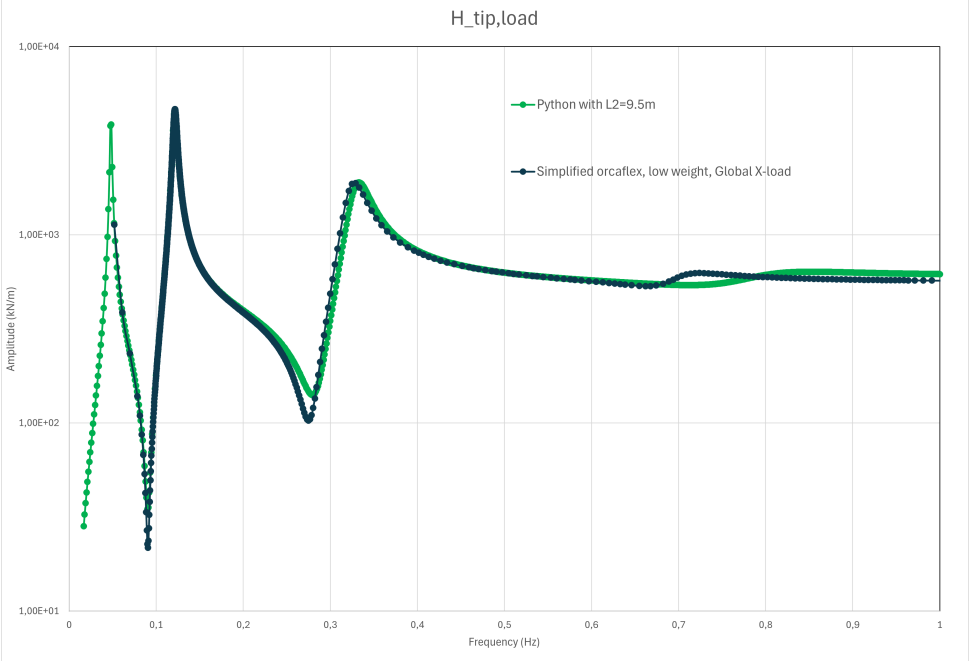


Figure 31: Python transferfunction with adapted L2

5.4 Side-load Spectrum Comparison

To further validate the dynamic model, the predicted crane side-load power spectral density (PSD) is compared between the Python and OrcaFlex implementations. This spectrum quantifies how the lateral loading at the crane tip is distributed across frequency, based on a representative sea state with $H_s = 1.0$ m and $T_p = 8.0$ s.

Python Calculation In Python, the side-load spectrum is computed using the formulation introduced in Equation (16):

$$S_F(\omega) = S_\eta(\omega) \cdot |\mathbf{u}_{tip,x}(\omega) \cdot H_{\text{tip}\to\text{load}}(\omega)|^2$$

where:

- $S_\eta(\omega)$ is the JONSWAP wave spectrum for the selected sea state,
- $\mathbf{u}_{tip,x}(\omega)$ is the horizontal RAO of the crane tip, validated in Section 4.7.2,
- $H_{\text{tip}\to\text{load}}(\omega)$ is the motion-to-force transfer function, discussed in Section 5.3.

This spectrum is evaluated across a frequency range from 0.00 to 1.00 Hz. The amplitude above 0.25 Hz is approaching 0, so it is left out in 33. The resulting PSD is expressed in kN^2/Hz.

OrcaFlex Calculation In OrcaFlex, the spectral response for the GX-direction force at the crane tip is directly computed in the frequency domain using the built-in spectral response analysis. This provides the PSD of the in-frame connection force, aligned with the vessel's transverse axis (local Ly), which corresponds to the global x-direction in this simulation setup.

Validation Across Multiple Sea States To assess robustness across different dynamic regimes, the sideload spectrum was computed for three distinct peak periods:

- $T_p = 6.0$ s: A sea state, with its significant wave height below the primary resonance range. See Figure 32
- $T_p = 8.0$ s: Near the system's second natural frequency, expected to show peak amplification. See Figure 33
- $T_p = 10.0$ s: A longer-period state, where the most of the excitation lies beyond the dominant resonant mode. See Figure 34

Figures 32, 33, and 34 compare the PSD outputs from Python and OrcaFlex for each respective case. Across all scenarios, the models show good agreement in spectral shape and peak location.

Figure 32: Side-load PSD comparison for $T_p = 6.0$ s, $H_s = 1.0$ m.

Figure 33: Side-load PSD comparison for $T_p = 8.0$ s, $H_s = 1.0$ m.

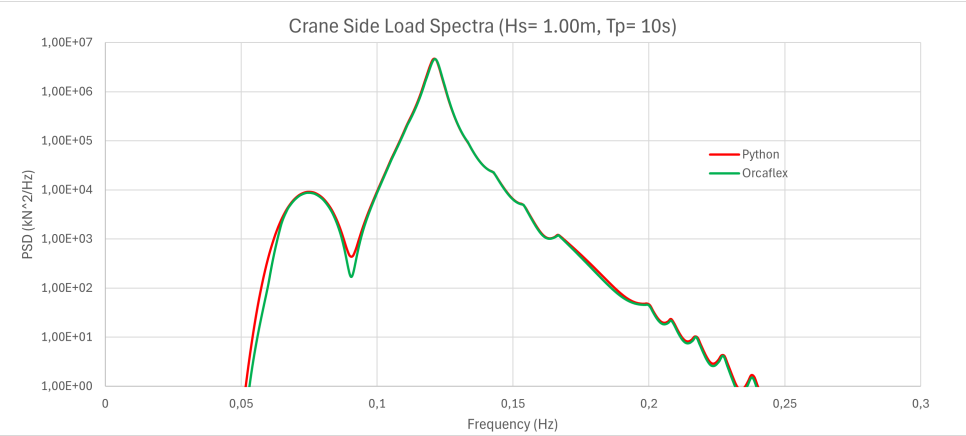


Figure 34: Side-load PSD comparison for $T_p = 10.0$ s, $H_s = 1.0$ m.

Interpretation The sideload spectra from Python and OrcaFlex show strong agreement across all three sea states. Peak locations align well, and the overall spectral shapes are consistent, validating the accuracy between the OrcaFlex and the analytical model.

This close alignment is primarily due to the fact that the same modeling simplifications were explicitly implemented in the OrcaFlex setup used for this comparison. These simplifications — documented in Section 5.3 — ensure that both models are evaluated on equivalent assumptions, allowing for a meaningful one-to-one comparison of the transfer function and spectral output.

In the next subsection, we perform a grid-based comparison between the analytical model and the full advanced OrcaFlex model, without adjusting for these simplifications. While small differences emerge in that case, they remain within acceptable bounds and do not compromise the model's ability to guide optimization decisions.

5.5 Grid-Based Validation of Most Probable Maximum Loads

To validate the complete pipeline of the Python-based model, a comparison is made between the computed most probable maximum (MPM) side-loads and those obtained using the OrcaFlex frequency-domain model. This is done for a representative range of sea states, covering the same (H_s, T_p) domain used in the workability evaluation.

Python Implementation The Python-based MPM grid is generated using the methodology described in Section 4.9. For each wave height and peak period combination, the side-load spectrum $S_F(f)$ is calculated, the spectral moments m_0 and m_2 are evaluated, and the corresponding most probable maximum load $F_{\max}$ is computed using Equation (34). The result is visualized in Figure 35.

Hs/Tp	0,125	0,375	0,625	0,875	1,125	1,375	1,625	1,875	2,125	2,375
2,5										
3,5										
4,5	3,039163	9,117489	15,19582							
5,5	20,58392	61,75175	102,9196	144,0874	185,2553					
6,5	57,43266	172,298	287,1633	402,0286	516,894	631,7593	746,6246			
7,5	83,90194	251,7058	419,5097	587,3136	755,1175	922,9214	1090,725	1258,529	1426,333	
8,5	92,5007	277,5021	462,5035	647,5049	832,5063	1017,508	1202,509	1387,511	1572,512	1757,513
9,5	90,06038	270,1811	450,3019	630,4227	810,5434	990,6642	1170,785	1350,906	1531,026	1711,147
10,5	83,05017	249,1505	415,2509	581,3512	747,4515	913,5519	1079,652	1245,753	1411,853	1577,953
11,5	75,14268	225,428	375,7134	525,9987	676,2841	826,5695	976,8548	1127,14	1277,426	1427,711
12,5	68,22276	204,6683	341,1138	477,5593	614,0048	750,4503	886,8958	1023,341	1159,787	1296,232
13,5	63,50024	190,5007	317,5012	444,5017	571,5022	698,5027	825,5032	952,5037	1079,504	1206,505
14,5	61,8211	185,4633	309,1055	432,7477	556,3899	680,0322	803,6744	927,3166	1050,959	1174,601
15,5	63,01323	189,0397	315,0661	441,0926	567,119	693,1455	819,1719	945,1984	1071,225	1197,251
16,5	65,78351	197,3505	328,9175	460,4846	592,0516	723,6186	855,1856	986,7526	1118,32	1249,887
17,5	68,67792	206,0338	343,3896	480,7455	618,1013	755,4572	892,813	1030,169	1167,525	1304,881
18,5	70,78821	212,3646	353,941	495,5174	637,0938	778,6703	920,2467	1061,823	1203,399	1344,976
19,5	71,78427	215,3528	358,9213	502,4899	646,0584	789,6269	933,1955	1076,764	1220,333	1363,901
20,5	71,69353	215,0806	358,4677	501,8547	645,2418	788,6289	932,016	1075,403	1218,79	1362,177
21,5	70,70363	212,1109	353,5181	494,9254	636,3327	777,7399	919,1472	1060,554	1201,962	1343,369
22,5	69,04394	207,1318	345,2197	483,3076	621,3955	759,4834	897,5713	1035,659	1173,747	1311,835

Figure 35: MPM load grid generated by the Python model.

OrcaFlex Implementation In OrcaFlex, the spectral response of the crane tip Ly-force (aligned with the global x-direction) is computed for each peak period T_p. Since the load response is linearly proportional to wave height H_s, the full grid can be constructed by scaling results accordingly. The same MPM estimation method is then applied, yielding the result in Figure 36.

Hs/Tp	0,125	0,375	0,625	0,875	1,125	1,375	1,625	1,875	2,125	2,375
2,5										
3,5										
4,5	1,462651	4,387952	7,313253							
5,5	14,5712	43,71359	72,85599	101,9984	131,1408					
6,5	50,69897	152,0969	253,4948	354,8928	456,2907	557,6886	659,0866			
7,5	80,86723	242,6017	404,3362	566,0706	727,8051	889,5395	1051,274	1213,008	1374,743	
8,5	93,2714	279,8142	466,357	652,8998	839,4426	1025,985	1212,528	1399,071	1585,614	1772,157
9,5	93,13587	279,4076	465,6794	651,9511	838,2229	1024,495	1210,766	1397,038	1583,31	1769,582
10,5	87,01939	261,0582	435,0969	609,1357	783,1745	957,2133	1131,252	1305,291	1479,33	1653,368
11,5	78,84574	236,5372	394,2287	551,9202	709,6117	867,3032	1024,995	1182,686	1340,378	1498,069
12,5	70,50559	211,5168	352,528	493,5392	634,5503	775,5615	916,5727	1057,584	1198,595	1339,606
13,5	63,09102	189,2731	315,4551	441,6371	567,8192	694,0012	820,1832	946,3653	1072,547	1198,729
14,5	58,00368	174,011	290,0184	406,0258	522,0331	638,0405	754,0478	870,0552	986,0626	1102,07
15,5	57,72107	173,1632	288,6053	404,0475	519,4896	634,9317	750,3739	865,816	981,2581	1096,7
16,5	61,11275	183,3382	305,5637	427,7892	550,0147	672,2402	794,4657	916,6912	1038,917	1161,142
17,5	66,64916	199,9475	333,2458	466,5441	599,8425	733,1408	866,4391	999,7374	1133,036	1266,334
18,5	72,2084	216,6252	361,042	505,4588	649,8756	794,2924	938,7092	1083,126	1227,543	1371,96
19,5	76,60541	229,8162	383,027	536,2379	689,4487	842,6595	995,8703	1149,081	1302,292	1455,503
20,5	79,43529	238,3059	397,1764	556,047	714,9176	873,7881	1032,659	1191,529	1350,4	1509,27
21,5	80,73449	242,2035	403,6725	565,1414	726,6104	888,0794	1049,548	1211,017	1372,486	1533,955
22,5	80,7302	242,1906	403,651	565,1114	726,5718	888,0322	1049,493	1210,953	1372,413	1533,874

Figure 36: MPM load grid generated by the OrcaFlex model.

Comparison and Interpretation Figure 37 presents the difference between both models, and Figure 38 shows the relative deviation as a percentage of the OrcaFlex result.

Hs/Tp	0,125	0,375	0,625	0,875	1,125	1,375	1,625	1,875	2,125	2,375
2,5										
3,5										
4,5	-1,57651	-4,72954	-7,88256							
5,5	-6,01272	-18,0382	-30,0636	-42,089	-54,1145					
6,5	-6,7337	-20,2011	-33,6685	-47,1359	-60,6033	-74,0707	-87,538			
7,5	-3,03471	-9,10413	-15,1736	-21,243	-27,3124	-33,3818	-39,4512	-45,5207	-51,5901	
8,5	0,770693	2,31208	3,853467	5,394854	6,936241	8,477627	10,01901	11,5604	13,10179	14,64317
9,5	3,075493	9,226479	15,37747	21,52845	27,67944	33,83042	39,98141	46,1324	52,28338	58,43437
10,5	3,969217	11,90765	19,84608	27,78452	35,72295	43,66138	51,59982	59,53825	67,47668	75,41512
11,5	3,703063	11,10919	18,51532	25,92144	33,32757	40,7337	48,13982	55,54595	62,95208	70,3582
12,5	2,282835	6,848506	11,41418	15,97985	20,54552	25,11119	29,67686	34,24253	38,8082	43,37387
13,5	-0,40922	-1,22767	-2,04612	-2,86457	-3,68302	-4,50147	-5,31992	-6,13837	-6,95682	-7,77527
14,5	-3,81742	-11,4523	-19,0871	-26,722	-34,3568	-41,9917	-49,6265	-57,2614	-64,8962	-72,5311
15,5	-5,29216	-15,8765	-26,4608	-37,0451	-47,6294	-58,2137	-68,7981	-79,3824	-89,9667	-100,551
16,5	-4,67076	-14,0123	-23,3538	-32,6953	-42,0369	-51,3784	-60,7199	-70,0614	-79,403	-88,7445
17,5	-2,02876	-6,08628	-10,1438	-14,2013	-18,2588	-22,3164	-26,3739	-30,4314	-34,4889	-38,5464
18,5	1,420197	4,26059	7,100984	9,941377	12,78177	15,62216	18,46256	21,30295	24,14334	26,98374
19,5	4,821142	14,46343	24,10571	33,748	43,39028	53,03256	62,67485	72,31713	81,95942	91,6017
20,5	7,741751	23,22525	38,70875	54,19226	69,67576	85,15926	100,6428	116,1263	131,6098	147,0933
21,5	10,03086	30,09259	50,15431	70,21603	90,27776	110,3395	130,4012	150,4629	170,5247	190,5864
22,5	11,68626	35,05877	58,43128	81,80379	105,1763	128,5488	151,9213	175,2938	198,6663	222,0389

Figure 37: Absolute difference in MPM load values between Python and OrcaFlex models [kN].

Hs/Tp	0,125	0,375	0,625	0,875	1,125	1,375	1,625	1,875	2,125	2,375
2,5										
3,5										
4,5	52%	52%	52%							
5,5	29%	29%	29%	29%	29%					
6,5	12%	12%	12%	12%	12%	12%	12%			
7,5	4%	4%	4%	4%	4%	4%	4%	4%	4%	
8,5	1%	1%	1%	1%	1%	1%	1%	1%	1%	1%
9,5	3%	3%	3%	3%	3%	3%	3%	3%	3%	3%
10,5	5%	5%	5%	5%	5%	5%	5%	5%	5%	5%
11,5	5%	5%	5%	5%	5%	5%	5%	5%	5%	5%
12,5	3%	3%	3%	3%	3%	3%	3%	3%	3%	3%
13,5	1%	1%	1%	1%	1%	1%	1%	1%	1%	1%
14,5	6%	6%	6%	6%	6%	6%	6%	6%	6%	6%
15,5	8%	8%	8%	8%	8%	8%	8%	8%	8%	8%
16,5	7%	7%	7%	7%	7%	7%	7%	7%	7%	7%
17,5	3%	3%	3%	3%	3%	3%	3%	3%	3%	3%
18,5	2%	2%	2%	2%	2%	2%	2%	2%	2%	2%
19,5	7%	7%	7%	7%	7%	7%	7%	7%	7%	7%
20,5	11%	11%	11%	11%	11%	11%	11%	11%	11%	11%
21,5	14%	14%	14%	14%	14%	14%	14%	14%	14%	14%
22,5	17%	17%	17%	17%	17%	17%	17%	17%	17%	17%

Figure 38: Relative difference in MPM load values as percentage of OrcaFlex reference.

While minor differences occur — particularly at high T_p and low T_p (due to very small absolute values) — the overall structure and trend of the grid are highly consistent. The shape of the solution surface is preserved, which is essential for the intended optimization process. Since the model aims to identify limiting configurations that reduce operational uptime, the ability to capture relative trends across the sea state space is more critical than the exact matching of individual load values.

The consistency across the grid validates the use of the Python model as a reliable tool for optimizing rigging configurations with respect to downtime risk.

6 Results

This section presents the outcomes of the optimization process aimed at improving offshore lifting workability through rigging geometry adjustments. Using the model and methodology described in the previous chapters, the Differential Evolution (DE) algorithm was applied to maximize the expected uptime over the August sea state distribution.

All results are based on the nominal crane, payload, and vessel properties as outlined earlier, with only the rigging lengths L_1, L_2, L_3, L_4 allowed to vary. The spreader width w_3 was fixed, and the total rigging length was constrained as discussed in Section 4.10.

The following subsections provide a detailed comparison between the nominal and optimized rigging setups for the month of August. Key performance indicators such as modal behavior, transfer function characteristics, and resulting workability are evaluated to quantify the gains achieved through optimization.

6.1 Optimization Outcomes for August

The optimization procedure described in Section 4.10 was applied to the August sea state distribution, resulting in a notable improvement in overall workability. This subsection compares the nominal and optimized rigging configurations and interprets the effects on system dynamics.

6.1.1 Nominal vs. Optimized Rigging Parameters

Table 4 presents the nominal and optimized values of the rigging parameters. The optimization led to a significant redistribution of the segment lengths while maintaining the total rigging length and satisfying the imposed angle constraint on δ. Notably:

- The upper sling L_1 increased from 43.1 m to 56.6 m.
- The lower sling L_2 was extended moderately.
- The diagonal slings L_3 shortened from 40.0 m to 31.2 m, increasing the included sling angle δ from 63.3° to 84.5°.
- The final leg L_4 shortened to preserve the total geometry constraint.

These changes reflect a strategic reshaping of the pendulum system: the increased angle δ stiffens the rigging horizontally and shifts dynamic coupling patterns.

Table 4: Nominal vs. optimized rigging parameters for August.

Parameter	**Nominal**	**Optimized**
L_1 [m]	43.10	56.62
L_2 [m]	5.50	7.01
L_3 [m]	40.00	31.24
L_4 [m]	15.50	11.39
w_3 [m] (fixed)	42.00	42.00
δ [°]	63.34	84.48

6.1.2 Modal Behavior

The modal periods of the pendulum system provide further insight into the effect of the optimized layout. Table 5 shows the natural periods before and after optimization. Most notably, the second mode — which was originally located at 8.16 seconds — shifted upward to 10.47 seconds. This change was one of the key mechanisms for increasing workability.

The August sea state distribution features a peak wave period around 8 seconds, meaning that resonant amplification around this frequency range is particularly undesirable. By shifting the second mode above this critical band, the system avoids resonance and reduces dynamic loading in the most probable sea conditions.

Table 5: Modal periods before and after optimization.

Mode	$\mathbf{T_{nom}}$ [s]	$\mathbf{T_{opt}}$ [s]
1	20.57	20.74
2	8.16	10.47
3	3.00	2.81
4	1.01	1.11

This upward modal shift illustrates how subtle changes in geometry can be used to tune the system's dynamic response to the offshore wave climate. The remaining modes remain relatively unchanged, ensuring that the structural behavior is not fundamentally altered outside of the target range.

6.1.3 Effect on Transfer Function

The shift in modal behavior is directly reflected in the motion-to-load transfer function $H_{\text{tip}\to\text{load}}(\omega)$, previously defined in Section 5.3. In the optimized case, the peak in $H_{\text{tip}\to\text{load}}$ moves to a lower frequency and flattens slightly. This reshaping reduces the amplification of dynamic loads at frequencies dominant in the August wave climate.

Figure 39 presents the transfer functions for both the nominal and optimized configurations. The reduction in magnitude around 0.12–0.13 Hz, corresponding to 8-second waves, confirms the effectiveness of the optimization.

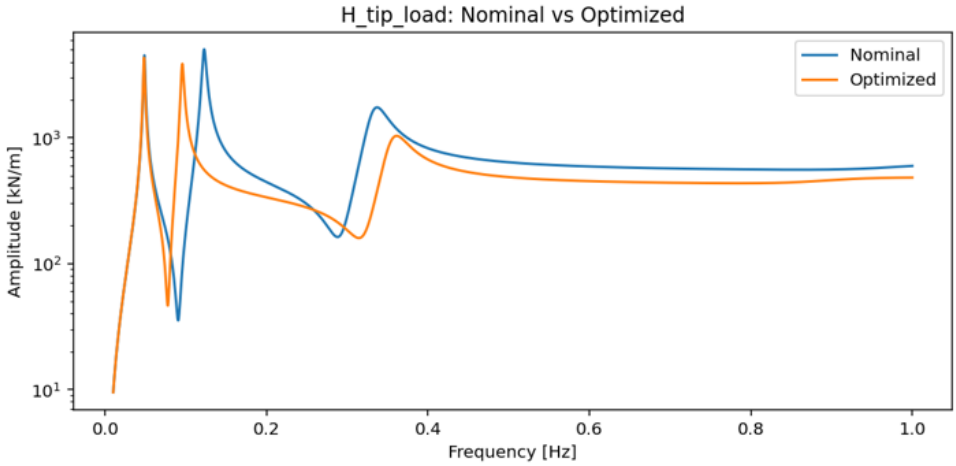


Figure 39: Comparison of transfer function $H_{\text{tip}\to\text{load}}$ for nominal and optimized rigging.

6.1.4 Side Load Grid and Workability

The improvement in dynamic behavior due to the optimized rigging layout is reflected in the grid of most probable maximum (MPM) side loads across all (H_s, T_p) sea states. Figure 40 shows the resulting load matrix after applying the optimization. Compared to the nominal case (Figure 22), substantial load reductions are observed in the critical range around $T_p = 8$ seconds, where resonance originally occurred.

Hs/Tp	0.00-0.25	0.25-0.50	0.50-0.75	0.75-1.00	1.00-1.25	1.25-1.50	1.50-1.75	1.75-2.00	2.00-2.25	2.25-2.50
2.0-3.0	0,029403	0,076101	0,113839	0,159374	0,20491	0,250445	0,295981	0,341516	0,387051	0,432587
3.0-4.0	0,389711	1,169134	1,793626	2,20635	2,751611	3,36308	3,974549	4,586018	5,197487	5,808956
4.0-5.0	1,836164	5,508492	9,18082	12,63135	15,27502	17,73878	20,3023	23,42574	26,54917	29,6726
5.0-6.0	5,806794	17,42038	29,03397	40,64756	52,26115	62,19929	70,84845	79,14162	87,2032	95,47773
6.0-7.0	15,41082	46,23247	77,05411	107,8758	138,6974	169,519	200,3407	225,8032	248,367	270,0311
7.0-8.0	34,22802	102,684	171,1401	239,5961	308,0521	376,5082	444,9642	513,4202	581,8763	643,0929
8.0-9.0	55,68246	167,0474	278,4123	389,7773	501,1422	612,5071	723,872	835,237	946,6019	1057,967
9.0-10.0	70,08598	210,2579	350,4299	490,6019	630,7738	770,9458	911,1178	1051,29	1191,462	1331,634
10.0-11.0	75,90291	227,7087	379,5145	531,3203	683,1262	834,932	986,7378	1138,544	1290,349	1442,155
11.0-12.0	75,82951	227,4885	379,1476	530,8066	682,4656	834,1247	985,7837	1137,443	1289,102	1440,761
12.0-13.0	72,86693	218,6008	364,3347	510,0685	655,8024	801,5362	947,2701	1093,004	1238,738	1384,472
13.0-14.0	69,34072	208,0222	346,7036	485,385	624,0665	762,7479	901,4293	1040,111	1178,792	1317,474
14.0-15.0	66,89421	200,6826	334,4711	468,2595	602,0479	735,8364	869,6248	1003,413	1137,202	1270,99
15.0-16.0	66,2087	198,6261	331,0435	463,4609	595,8783	728,2958	860,7132	993,1306	1125,548	1257,965
16.0-17.0	66,88128	200,6438	334,4064	468,1689	601,9315	735,694	869,4566	1003,219	1136,982	1270,744
17.0-18.0	67,99953	203,9986	339,9977	475,9967	611,9958	747,9948	883,9939	1019,993	1155,992	1291,991
18.0-19.0	68,8056	206,4168	344,028	481,6392	619,2504	756,8616	894,4728	1032,084	1169,695	1307,306
19.0-20.0	68,92468	206,774	344,6234	482,4727	620,3221	758,1715	896,0208	1033,87	1171,72	1309,569
20.0-21.0	68,28795	204,8639	341,4398	478,0157	614,5916	751,1675	887,7434	1024,319	1160,895	1297,471
21.0-22.0	66,99195	200,9759	334,9598	468,9437	602,9276	736,9115	870,8954	1004,879	1138,863	1272,847
22.0-23.0	65,19419	195,5826	325,971	456,3593	586,7477	717,1361	847,5245	977,9129	1108,301	1238,69

Figure 40: MPM load grid for August after optimization [kN].

This reduction in peak loads enables a larger portion of the grid to remain below the side-load limit. The resulting binary operability map is multiplied by the wave scatter probabilities, yielding the optimized workability distribution shown in Figure 41.

Hs/Tp	0.00-0.25	0.25-0.50	0.50-0.75	0.75-1.00	1.00-1.25	1.25-1.50	1.50-1.75	1.75-2.00	2.00-2.25	2.25-2.50
2.0-3.0	0	0,023522	0,070565	0	0	0	0	0	0	0
3.0-4.0	0	0,349462	2,483199	1,754032	0,124328	0	0	0	0	0
4.0-5.0	0	0,127688	2,446237	4,932796	3,75672	0,910618	0,147849	0,016801	0	0
5.0-6.0	0	0,443548	3,887769	4,438844	3,618952	2,580645	1,297043	0,430108	0,120968	0,036962
6.0-7.0	0	0,695565	5,927419	5,588038	3,313172	1,928763	1,381048	0,997984	0,581317	0,299059
7.0-8.0	0	0,62164	4,028898	5,16129	3,141801	1,327285	0,658602	0,500672	0,356183	0,292339
8.0-9.0	0,00672	0,863575	1,911962	1,703629	1,942204	1,081989	0	0	0	0
9.0-10.0	0	0,850134	2,308468	1,065188	0,806452	0	0	0	0	0
10.0-11.0	0	0,430108	2,174059	1,051747	0,510753	0	0	0	0	0
11.0-12.0	0	0,070565	0,554435	0,325941	0,181452	0	0	0	0	0
12.0-13.0	0	0,258737	0,688844	0,61828	0,252016	0	0	0	0	0
13.0-14.0	0	0,134409	0,31922	0,15457	0,181452	0	0	0	0	0
14.0-15.0	0	0,221774	0,591398	0,292339	0,231855	0	0	0	0	0
15.0-16.0	0	0,023522	0,063844	0,057124	0,070565	0	0	0	0	0
16.0-17.0	0	0,026882	0,225134	0,221774	0,053763	0	0	0	0	0
17.0-18.0	0	0,00672	0,020161	0,036962	0	0	0	0	0	0
18.0-19.0	0	0,070565	0,084005	0,080645	0,00336	0	0	0	0	0
19.0-20.0	0	0	0,016801	0,023522	0	0	0	0	0	0
20.0-21.0	0	0	0,00672	0	0	0	0	0	0	0
21.0-22.0	0	0	0,043683	0,033602	0	0	0	0	0	0
22.0-23.0	0	0	0,00672	0	0	0	0	0	0	0

Figure 41: Workability map for August after optimization.

The total expected workability for August improves from 79.92% in the nominal case to 93.76% after optimization. This gain is achieved purely through geometric rigging changes, without altering the crane, vessel, or payload parameters. The results confirm the effectiveness of the optimization method in minimizing dynamic side loads and increasing allowable weather windows.

Table 6: Total expected workability for August before and after optimization.

Configuration	Workability [%]	Change [p.p.]
Nominal	79.92	–
Optimized	93.76	+13.84

It is important to note that this workability improvement is based solely on the dynamic side load at the crane tip. Other operational constraints — such as vertical load limits, vessel motions, or clearance requirements — have not been considered in the optimization process. As such, the presented gain represents the theoretical maximum improvement in allowable conditions with respect to this specific limit state.

6.2 New Design Proposal

6.3 Monthly Optimization Summary

For completeness and to assess the robustness and consistency of the optimization approach, the same method was applied across all months of the year. This section summarizes the improvement in expected workability, the corresponding changes in rigging geometry, and the resulting shifts in the system's modal behavior.

Table 7: Workability before and after optimization per month.

Month	Nominal Workability [%]	Optimized Workability [%]
Jan	54.02	71.09
Feb	56.50	73.42
Mar	55.77	71.73
Apr	57.01	77.86
May	68.41	90.31
Jun	74.81	95.06
Jul	75.46	95.47
Aug	**79.92**	**93.76**
Sep	67.09	82.66
Oct	66.06	82.98
Nov	60.57	77.10
Dec	55.53	73.52

Workability Improvement Across all months, the optimization consistently improves the expected uptime by approximately 15–20%. August is used as the reference month throughout this thesis, but similar performance gains are observed year-round.

Table 8: Nominal and optimized rigging geometry per month. All values in meters.

Month	L_1	L_2	L_3	L_4
Nominal	**43.10**	**5.50**	**40.00**	**15.50**
Jan	57.01	6.06	31.51	11.58
Feb	57.44	6.26	31.43	11.05
Mar	56.62	7.01	31.24	11.39
Apr	56.56	9.76	30.23	10.09
May	56.56	9.76	30.23	10.08
Jun	56.74	5.43	30.21	14.26
Jul	56.56	9.76	30.23	10.09
Aug	**56.62**	**7.01**	**31.24**	**11.39**
Sep	57.61	6.39	31.56	10.58
Oct	56.08	5.88	31.40	12.85
Nov	57.44	6.26	31.43	11.05
Dec	56.56	9.76	30.23	10.09

Rigging Parameter Adjustments The optimized rigging layouts show similar trends each month: the upper sling L_1 is extended, the diagonal slings L_3 shortened, and the final leg L_4 adjusted to maintain the total geometry constraint. This recurring adjustment strategy confirms the optimizer's tendency to decouple the second mode from the dominant wave energy band.

Table 9: Second natural period before and after optimization.

Month	$\mathbf{T_2^{nom}}$ [s]	$\mathbf{T_2^{opt}}$ [s]
Jan	8.16	10.40
Feb	8.16	10.43
Mar	8.16	10.46
Apr	8.16	10.44
May	8.16	10.39
Jun	8.16	10.33
Jul	8.16	10.35
Aug	**8.16**	**10.47**
Sep	8.16	10.36
Oct	8.16	10.38
Nov	8.16	10.37
Dec	8.16	10.39

Modal Behavior After Optimization The most consistent dynamic effect of the optimization is the upward shift of the second natural period, moving it away from the critical 8-second region dominant in the North Atlantic wave climate. This decoupling strategy effectively reduces dynamic amplification.

6.4 New Design Proposal

The analysis of the monthly optimization outcomes reveals a clear trend in the parameter adaptations that lead to improved workability. Most notably, the upper sling length L_1 consistently increases across all months, typically reaching values above 56 m. This elongation is especially effective in shifting the second natural period away from the dominant wave energy region (around 8 seconds), which reduces resonance and improves dynamic performance.

The parameter changes are not arbitrary but follow a recognizable pattern:

- L_1 increases significantly to decouple the second mode from critical sea states.
- L_3 is consistently reduced, resulting in a steeper sling angle δ, which enhances lateral stiffness.
- L_4 and L_2 are adjusted to satisfy the total length and angle constraints while supporting the dynamic reshaping.

This consistency across months raises an important design consideration. Rather than implementing different rigging setups per month, a practical alternative is to adopt a single, optimized configuration that performs well year-round. The August-optimized configuration offers a compelling baseline, as it lies close to the annual average and achieves excellent performance under typical summer sea states.

The proposed new rigging design, based on the August configuration in Table 8, is therefore:

$$L_1 = 56.62 \text{ m}, \quad L_2 = 7.01 \text{ m}, \quad L_3 = 31.24 \text{ m}, \quad L_4 = 11.39 \text{ m}$$

This setup adheres to all constraints introduced earlier and results in a predicted workability of 93.76% for August, compared to 79.92% for the nominal rigging. A visual representation of the proposed rigging design is shown in Figure 42.

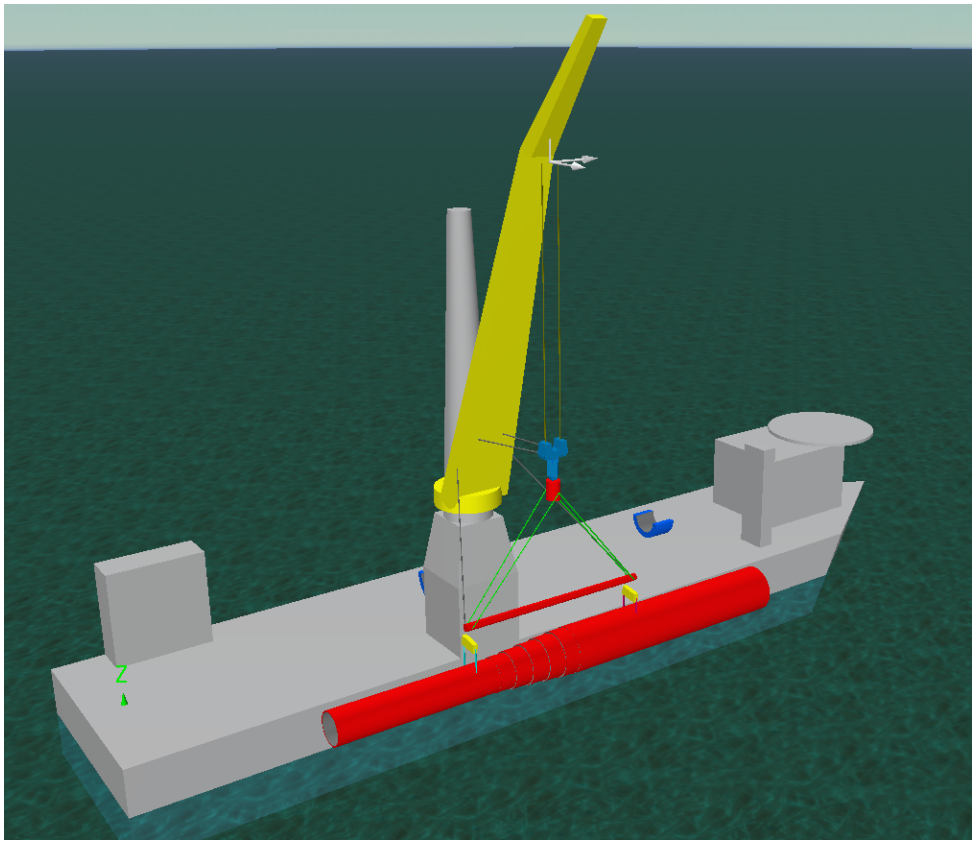


Figure 42: Visualization of the proposed new rigging configuration, based on the August-optimized geometry.

Although this research focused exclusively on side-load limitation as the critical constraint, it is important to note that other structural or operational limits may become relevant with the new design. Any implementation should therefore be supported by a comprehensive structural and procedural review beyond the scope of this study.

Future extensions could also explore season-dependent or forecast-driven reconfiguration strategies. The same evaluation framework can be directly applied to weather forecasts, enabling dynamic selection of the best rigging layout for upcoming sea states.

7 Discussion and Conclusion

7.1 Summary of Key Findings

This thesis aimed to identify and mitigate the dominant drivers of weather-related downtime during offshore monopile installation using the Bokalift 2 vessel. The focus was placed on the lift-off phase, where dynamic amplification in the rigging system was suspected to cause frequent exceedance of operational limits. The central research question was:

> *How can the rigging configuration of an offshore heavy lift vessel be optimized to reduce downtime during offshore monopile lift-off operations?*

To guide the investigation, the following sub-questions were addressed:

1. **Which operational constraint contributes most to weather-related downtime, and is suitable for optimization?**
   Exceedance analysis showed that the **Crane Sideload UC** was the most frequently violated limit during critical phases (especially LC3a and LC3b). It co-exceeded with tension and motion-related constraints, confirming the central role of excessive oscillation in downtime.
2. **Can a simplified dynamic model accurately capture the behavior of the crane–rigging–monopile system?**
   A linear Lagrangian model was developed and validated against OrcaFlex. It captured key dynamics and proved reliable for predicting sideload spectra and natural frequencies.
3. **Can this model be used to optimize the rigging geometry for improved workability?**
   Yes. The model enabled a frequency-domain framework that linked environmental conditions to sideload exceedances and enabled geometry-based tuning of system response.
4. **Does this optimization result in meaningful operational improvement?**
   Optimization of rigging lengths to shift resonance away from the dominant wave period (around 8 s) yielded **10–20 percentage points** improvement in monthly workability, with limited structural changes to the lifting hardware.

These conclusions are supported by the following key findings:

- **Crane Sideload UC** was identified as the dominant limiting criterion, often exceeded in lift-off conditions and strongly correlated with pendulum-induced dynamics.
- **Resonance around** $T_p = 8.16$ s — matching the second natural mode of the system — was the main cause of exceedances in critical load cases.
- A linearized **Lagrangian multi-body model** was developed and used to derive a frequency-domain transfer function from crane tip motion to sideload.
- A **grid-based workability framework** was implemented, combining spectral load prediction with environmental probability weighting.
- **Differential evolution optimization** allowed targeted reduction of sideload response by tuning cable lengths, leading to a measurable improvement in operational uptime.

Taken together, the results show that **modal tuning of the rigging layout** can reduce the operational impact of resonance and meaningfully improve offshore lifting workability.

7.2 Limitations and Critical Reflection

While this study presents a structured approach to identifying and mitigating dynamic overloads during offshore monopile installation, it is important to critically reflect on the assumptions, limitations, and implications of the methodology used.

Validation Discrepancies A comparison was performed between the simplified dynamic model and frequency-domain simulations from OrcaFlex. While both models show similar *trends* — notably the location and shape of resonant peaks — the predicted side load magnitudes differ consistently. The analytical model **underpredicts** the peak side loads compared to the OrcaFlex results. These differences are especially apparent in the frequency-domain transfer function $H_{\text{tip}\to\text{load}}$, which shows lower amplitude in the analytical model across most of the excitation range.

Despite this discrepancy in absolute values, the analytical model captures the correct *modal structure* and accurately identifies the dominant second mode near $T_p = 8.16\,\text{s}$. This alignment in resonance location — and the consistent shape of the response — supports the model's use for **optimization purposes**, where relative performance across different rigging configurations is more important than precise force prediction.

Threshold Definition The decision to reduce the sideload threshold to 684 kN (0.57·1200) was made to align the analytical model results with those of a more similar OrcaFlex simulation. While this adjustment introduces some uncertainty, it reflects a practical compromise given the consistent differences observed between the simplified and advanced models. These differences are primarily attributed to modeling assumptions, including damping distribution and cable stiffness as referred to in 4.9.2.

It is important to note that the objective of this thesis was not to evaluate OrcaFlex or to benchmark various OrcaFlex configurations against each other. Rather, OrcaFlex served as a reference tool to validate the dynamic response predicted by the analytical model. In that context, the threshold adjustment helps maintain a consistent and conservative limit when using the simplified model for parametric studies and optimization.

The approach used here prioritizes relative comparisons and trends over absolute accuracy. While the threshold shift is empirical, it enables meaningful insight into how rigging configurations influence system dynamics and operational feasibility. The results indicate that optimization of rigging parameters can significantly improve workability and should be considered in future projects.

Modeling Assumptions Several modeling simplifications were made to enable symbolic derivation, fast simulation, and tractable optimization. These include:

- **Planar dynamics:** Only side-swing motion in a vertical plane is considered. Real offshore lifting involves 3D behavior including yaw, out-of-plane pendulum effects, and crane slew corrections.
- **Small-angle approximation:** The Lagrangian formulation assumes small oscillations around equilibrium.
- **Idealized slings:** All cables are assumed to be inextensible, massless, and always under tension. This excludes effects such as elastic stretch, tension fluctuation, or line slack.
- **No feedback to vessel:** The model assumes that vessel motion is unaffected by the dynamics of the suspended rigging system. This simplification is justified by the large mass difference between the monopile and the vessel. However, small deviations were observed near resonance frequencies, indicating that inertial coupling may become non-negligible under specific conditions.
- **Simplified damping:** Rayleigh damping is used with a uniform 2% critical damping ratio for the first two modes. This approximation does not capture all physical sources of dissipation, such as vortex shedding, spreader friction, or nonlinear joint damping.
- **Deterministic inputs:** The excitation is modeled using predefined JONSWAP spectra and RAOs, assuming steady-state conditions. Human factors, crane control, and short-term variability are excluded.

Interpretation and Scope These assumptions limit the model's applicability to detailed design or certification-level calculations. However, for early-stage optimization and sensitivity exploration, the approach remains highly valuable. It enables fast evaluation of design changes and offers physical insight into the system's modal behavior, which would be difficult to extract from full 3D nonlinear models.

Nonetheless, caution is warranted in generalizing results. The optimization results should be interpreted as directionally correct rather than quantitatively precise. Future validation with measurement data or higher-fidelity simulation would be required to confirm load margins and refine the safety envelope.

7.3 Recommendations

7.3.1 Practical Recommendations for Boskalis

The findings of this study suggest that dynamic optimization of the rigging configuration can offer meaningful reductions in weather-related downtime during monopile lift-off operations. Based on these results, the following practical recommendations are proposed:

- **Pay closer attention to rigging modal alignment during project planning:** Although modal analysis is already included in engineering workflows, this study highlights the importance of evaluating rigging-induced resonance modes specifically in relation to local wave climates. Additional scrutiny may be warranted in the early design stages, especially when new rigging geometries are introduced.
- **Use rigging length tuning as a low-effort mitigation strategy:** Adjusting sling and cable lengths is relatively simple from both an engineering and logistical perspective, yet can offer meaningful gains in workability by shifting resonance frequencies away from critical sea states.
- **Explore the use of damping elements in rigging design:** The sensitivity of predicted side loads to damping assumptions suggests that introducing physical damping — whether passive (e.g., friction elements, viscoelastic inserts) or active — could be an effective way to reduce dynamic amplification. While not explored in this thesis, the concept shows promise and warrants further investigation in future projects.

7.3.2 Recommendations for Future Research

This thesis focused on a linear, planar rigging model and its application to a specific lift-off load case. Future research could broaden the impact and applicability of this work by exploring the following directions:

- **Extend to full 3D dynamic modeling:** Incorporating out-of-plane motion and 3D coupling would enable realistic simulation of skewed sea states and arbitrary crane headings.
- **Calibrate against offshore measurements:** Validation using motion sensors and load cells from offshore lifts would build confidence in the model's predictive value and refine its input assumptions.
- **Evaluate damping technologies:** Investigate the effectiveness of passive or active damping systems integrated into the rigging. These could provide a practical method for reducing resonance-driven overloads.
- **Integrate economic decision-making:** Assess the operational and financial impact of rigging design changes using cost–benefit or risk-based frameworks.
- **Incorporate additional UC constraints into the optimization:** This study focused on crane sideload limits. A natural extension is to simultaneously evaluate the other UC's for holistic workability assessment.

7.3.3 Further Model and Methodological Improvements

To enhance accuracy and robustness, several technical aspects of the current modeling approach could be improved in future work:

- **Refine damping formulation:** Rayleigh damping was used for its simplicity, but future implementations could explore experimentally derived damping to better reflect real offshore energy dissipation.
- **Include flexible rigging and structural elasticity:** Current rigid-body assumptions can be relaxed by incorporating elasticity in slings and spreader bars, which becomes increasingly relevant at higher loads or during transient events.
- **Expand beyond lift-off operations:** Although lift-off was the most critical phase identified, extending the framework to cover Empty spreader, Pre-tension, Storage lane and Upend hinge would improve overall operational planning.

References

Chopra, A. K. (2017). *Dynamics of structures: Theory and applications to earthquake engineering* (5th ed.). Pearson.

Clough, R. W., & Penzien, J. (1993). *Dynamics of structures* (2nd ed.). McGraw-Hill.

DNV. (2021). *Recommended practice dnv-rp-c205: Environmental conditions and environmental loads* (Tech. Rep.). DNV AS.

DNV. (2024). *Energy transition outlook 2024.* Retrieved from `https://www.dnv.com/energy-transition-outlook/download/` (Accessed May 2025)

DNV GL. (2019). *Recommended practice dnvgl-rp-c205: Environmental conditions and environmental loads* (Tech. Rep.). DNV GL AS.

Faltinsen, O. M. (1990). *Sea loads on ships and offshore structures.* Cambridge University Press.

Goldstein, H., Poole, C. P., & Safko, J. L. (2002). *Classical mechanics* (3rd ed.). Addison-Wesley.

Hand, L. N., & Finch, J. D. (1998). *Analytical mechanics.* Cambridge University Press.

Hasselmann, K., Barnett, T. P., Bouws, E., Carlson, H., Cartwright, D. E., Enke, K., ... Walden, H. (1973). Measurements of wind-wave growth and swell decay during the joint north sea wave project (jonswap). *Ergänzungsheft zur Deutschen Hydrographischen Zeitschrift, Reihe A*, *8*(12).

IEA. (2019). *Offshore wind outlook 2019.* Retrieved from `https://www.iea.org/reports/offshore-wind-outlook-2019` (Accessed May 2025)

Ishihara, K., & Kashiwagi, M. (2018). Bayesian estimation of directional wave-spectrum using vessel motions and wave-probes: Proposal and preliminary experimental validation. *Proceedings of the 37th International Conference on Ocean, Offshore and Arctic Engineering (OMAE).* Retrieved from `https://www.researchgate.net/publication/323024378_Bayesian_Estimation_of_Directional_Wave-Spectrum_Using_Vessel_Motions_and_Wave-Probes_Proposal_and_Preliminary_Experimental_Validation` (Accessed June 2025)

Journée, J., & Massie, W. (2008). *Offshore hydromechanics* (Second Edition ed.). Delft University of Technology. (Lecture notes, Offshore Engineering Series)

McLean, S. (2023). *How the feeder barge method is the key to unlocking increased offshore wind capacity in the usa.* Retrieved from `https://www.spinergie.com/blog/feeder-barge-method-key-to-unlocking-increased-offshore-wind-capacity-usa` (Accessed: 2025-07-11)

Noordzeeloket. (2020). *Routekaart windenergie op zee 2030.* Retrieved from `https://www.noordzeeloket.nl/functies-gebruik/windenergie/energietransitie-noordzee/` (Accessed May 2025)

Orcina Ltd. (2024). *Rayleigh damping in orcaflex.* `https://www.orcina.com/webhelp/OrcaFlex/Content/html/Rayleighdamping.htm`. (Accessed: 2025-07-11)

Storn, R., & Price, K. (1997). Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, *11*, 341-359. doi: 10.1023/A:1008202821328

Taylor, J. R. (2005). *Classical mechanics.* Sausalito, California: University Science Books.

The SciPy community. (2024a). *scipy.linalg.eigh documentation.* Retrieved 2025-07-12, from `https://docs.scipy.org/doc/scipy/reference/generated/scipy.linalg.eigh.html`

The SciPy community. (2024b). *scipy.optimize.differential_evolution documentation.* Retrieved 2025-07-12, from `https://docs.scipy.org/doc/scipy/reference/generated/scipy.optimize.differential_evolution.html`

U.S. Customs and Border Protection. (2024, December). *Informed compliance publication: What every member of the trade community should know about the jones act* (Tech. Rep.). U.S. Customs and Border Protection. Retrieved from `https://www.cbp.gov/sites/default/files/2024-12/Jones%20Act%20ICP_Complete_04DEC24.pdf` (Accessed: 2025-07-11)

WindEurope. (2020). *Offshore wind industry - key trends and statistics 2019.* Retrieved from `https://windeurope.org/about-wind/statistics/offshore/european-offshore-wind-industry-key-trends-statistics-2019/` (Accessed May 2025)

World Forum Offshore Wind. (2024, April). *Global offshore wind report 2023* (Tech. Rep.). World Forum Offshore Wind. Retrieved from `https://wfo-global.org/wp-content/uploads/2024/04/WFO-Report-2024Q1.pdf` (Accessed: 2025-07-12)

A Python Script

A.1 Workability Calculation Script

```
#


# 0. Imports and Plotting helpers
#


import numpy as np
import sympy as sp
import pandas as pd
import matplotlib.pyplot as plt
from matplotlib.ticker import FuncFormatter, ScalarFormatter
import scipy.linalg as LA
from scipy.interpolate import interp1d
from scipy.linalg import solve, eigh

def zoom_ylim(ax, y):
    y_pos = y[y > 0]
    if len(y_pos) == 0:
        return
    ymin = max(y_pos.min()*1e-2, 1e-4*y_pos.max())
    ymax = y_pos.max()*1.2
    ax.set_ylim(ymin, ymax)

def sci_formatter(x, pos):
    if x == 0: return "0"
    exponent = int(np.floor(np.log10(np.abs(x))))
    coeff = x / (10 ** exponent)
    return r"${:.1f} \times 10^{{{}}}$".format(coeff, exponent)

sci_fmt = FuncFormatter(sci_formatter)

#


# 1. Nominal parameters
#


m1, L1, I1, m2, L2, I2, m3, I3, w3, L3, m4, I4, L4, g = \
    sp.symbols('m1 L1 I1 m2 L2 I2 m3 I3 w3 L3 m4 I4 L4 g', real=True)

#Nominal parameters
params_const = {
    m1: 226_790.0,
    L1: 43.1,
    I1: 529_800.0,
    m2: 39_000.0,
    L2: 5.5,
    I2: 58_240.0,
    m3: 170_000.0,
    I3: 16_920_000.0,
    w3: 42.0,
    L3: 40.0,
    m4: 1_868_300.0,
    I4: 1_789_000_000.0,
    L4: 15.5,
    g : 9.81
}

params_syms = (m1, L1, I1, m2, L2, I2, m3, I3, w3, L3, m4, I4, L4, g)

def _vals(pdict):
    """Return a tuple of parameter values in the order params_syms."""
    return tuple(pdict[s] for s in params_syms)

#


# Unpack all numeric parameters into plain Python names
#
```

```
# _vals returns the tuple in the order (m1, L1, I1, m2, L2, I2, m3, I3, w3, L3, m4, I4,
    L4, g)
m1_, L1_, I1_, m2_, L2_, I2_, m3_, I3_, w3_, L3_, m4_, I4_, L4_, g_ = _vals(params_const)

#


# 2. Lagrange model     M, K (lambdified)
#


t     = sp.symbols('t')
R     = sp.Function('R')(t)
Rdot = sp.diff(R, t)
d     = sp.sqrt(L3**2 - (w3/2)**2)    # distance between m2 and m3
g_val = params_const[g]

#  s m a l l angle  substitution dict
 1 ,  2 ,  3 ,  4  = [sp.Function(f'  {i}')(t) for i in range(1, 5)]
sin_small = {sp.sin(  ):           for    in ( 1 ,  2 ,  3 ,  4 )}
cos_small = {sp.cos(  ): 1 -    **2/2 for    in ( 1 ,  2 ,  3 ,  4 )}
subs_small = {**sin_small, **cos_small}

# positions using full sin/cos
x1 = R + L1*sp.sin( 1 )
z1 = -L1*sp.cos( 1 )

x2 = x1 + L2*sp.sin( 2 )
z2 = z1 - L2*sp.cos( 2 )

x3 = x2 + d*sp.sin( 3 )
z3 = z2 - d*sp.cos( 3 )

x4 = x3 + L4*sp.sin( 4 )
z4 = z3 - L4*sp.cos( 4 )

    = ( 1 ,  2 ,  3 ,  4 )
 d  = tuple(sp.diff(q, t) for q in   )

# kinetic + potential energies in exact form
T_exact = (
    m1*(sp.diff(x1,t)**2 + sp.diff(z1,t)**2) + I1* d [0]**2
  + m2*(sp.diff(x2,t)**2 + sp.diff(z2,t)**2) + I2* d [1]**2
  + m3*(sp.diff(x3,t)**2 + sp.diff(z3,t)**2) + I3* d [2]**2
  + m4*(sp.diff(x4,t)**2 + sp.diff(z4,t)**2) + I4* d [2]**2
) / 2

V_exact = g*(
    m1*(z1 - (-L1))
  + m2*(z2 - (-L1 - L2))
  + m3*(z3 - (-L1 - L2 - d))
  + m4*(z4 - (-L1 - L2 - d - L4))
)

# apply  s m a l l angle  approximation
T = sp.simplify(T_exact.subs(subs_small))
V = sp.simplify(V_exact.subs(subs_small))

#  E u l e r Lagrange      linear M      + K     = 0
theta_zero = { 1 : 0,  2 : 0,  3 : 0,  4 : 0}
M_sym = sp.hessian(T,  d ).subs(theta_zero)
K_sym = sp.hessian(V,     ).subs(theta_zero)
#sp.pprint(M_sym, use_unicode=True)
#sp.pprint(K_sym, use_unicode=True)

# lambdify for speed
f_M = sp.lambdify(params_syms, sp.simplify(M_sym), 'numpy')
f_K = sp.lambdify(params_syms, sp.simplify(K_sym), 'numpy')

# numerical M, K
M = np.asarray(f_M(*_vals(params_const)), float)
K = np.asarray(f_K(*_vals(params_const)), float)

# validate d is positive
d_val = float(d.subs(params_const))
```

```
if d_val <= 0:
    raise ValueError(f"Invalid geometry: computed d = {d_val:.4g}. "
                     "Require L3   > (w3/2)   .")

#Build base-coupling M_b and total mass M_bb
M_bb_sym = m1 + m2 + m3 + m4
Mb1 = (m1 + m2 + m3 + m4)*L1
Mb2 = (m2 + m3 + m4)*L2
Mb3 = (m3 + m4)*d
Mb4 = (m4)*L4

M_b_sym = sp.Matrix([Mb1, Mb2, Mb3, Mb4])
M_bb_sym = m1 + m2 + m3 + m4

# lambdify them
f_Mb   = sp.lambdify(params_syms, M_b_sym,  'numpy')
f_Mbb  = sp.lambdify(params_syms, M_bb_sym, 'numpy')
#Evaluate numerically
Mb  = np.asarray(f_Mb(*_vals(params_const)), float).flatten()  # shape (4,)
Mbb = float(f_Mbb(*_vals(params_const)))

#


# 3. Rayleigh damping
#


# Modal analysis (after M, K defined)
 2 ,     = LA.eigh(K, M)
 n     = np.sqrt( 2 )
T_modes = 2*np.pi/ n

# Target 2% on modes 1 & 2
  _target   = 0.02
   ,     = LA.solve(
     [[1/ n [0],    n [0]],
      [1/ n [1],    n [1]]],
     2*  _target  *np.ones(2)
)

# Build damping matrix
C =    *M   +    *K

  _modal   = np.array([
     (  [:,i].conj().T @ (C @    [:,i])) / (2* n [i])
     for i in range(len( n ))
]).real


#


# 4. Excitation / output vectors (with damping + F_tip via   D  Alembert  )
#



def H_tip_load(omega):

    d_ = np.sqrt(L3_**2 - (w3_/2)**2)

    # Ensure omega is array-like
    w = np.atleast_1d(omega)
    H = np.zeros_like(w, dtype=complex)

    # Effective inertias (Ip) and coupling inertias (Iij)
    Ip1 = I1_ + (m1_ + m2_ + m3_ + m4_) * L1_**2
    Ip2 = I2_ + (m2_ + m3_ + m4_) * L2_**2
    Ip3 = I3_ + I4_ + (m3_ + m4_) * d_**2
    Ip4 = m4_ * L4_**2

    I12 = L1_ * L2_ * (m2_ + m3_ + m4_)
    I13 = L1_ * d_  * (m3_ + m4_)
    I14 = m4_ * L1_ * L4_
    I23 = L2_ * d_  * (m3_ + m4_)
```

```
    I24 = m4_ * L2_ * L4_
    I34 = m4_ * d_  * L4_

    # Gravitational stiffness terms
    K1 = g_ * L1_ * (m1_ + m2_ + m3_ + m4_)
    K2 = g_ * L2_ * (m2_ + m3_ + m4_)
    K3 = g_ * d_   * (m3_ + m4_)
    K4 = g_ * L4_ * m4_

    # Rayleigh damping matrix
    C_local =   *M +   *K  # 4 4

    for i, wi in enumerate(w):
        s = 1j * wi

        # Dynamic stiffness matrix D   = s   M + s C + K (4 4 )
        D   = s**2 * M + s * C_local + K

        # Assemble augmented dynamic system (5 5 ):
        D_dyn = np.zeros((5, 5), dtype=complex)
        D_dyn[:4, :4] = D
        D_dyn[4, :4] = s**2 * Mb
        D_dyn[4, 4]  = -1.0

        # Right-hand side vector from base acceleration input
        # RHS = -s   * [Mb; Mbb] * R(s), where R(s) = 1
        rhs = -s**2 * np.append(Mb, Mbb)

        # Solve system     output is: [ 1 ,  2 ,  3 ,  4 , F_tip]
        sol = np.linalg.solve(D_dyn, rhs)

        # Store only transfer function H = F_tip(s) / R(s)
        H[i] = sol[4]

    return H[0] if np.isscalar(omega) else H

#


# 5. Vessel motion RAO (waves     vessel CoG) + tip transform
#


raopath  = r"C:\Users\olla\OneDrive - Boskalis\Thesis Olivier Lagas\BL2RAOS.xlsx"
raosheet = "0"

df = pd.read_excel(raopath, sheet_name=raosheet, header=[0, 1])
df.columns = [" ".join(c).strip() for c in df.columns]
 _tab   = 2*np.pi / df[df.columns[0]].values

dofs = ["Surge", "Sway", "Heave", "Roll", "Pitch", "Yaw"]
def cpx(d):
    if d in ("Roll","Pitch","Yaw"):
        amp_col   = f"{d} ampl (deg/m)"
        phase_col = f"{d} Phase (deg)"
        # read in deg/m, convert to rad/m
        amp = np.deg2rad(df[amp_col].values)
    else:
        amp_col   = f"{d} ampl (m/m)"
        phase_col = f"{d} Phase (deg)"
        amp = df[amp_col].values

    phs = np.deg2rad(df[phase_col].values)   # phase always in  d e g rad

    return interp1d( _tab , amp * np.exp(-1j*phs),
                     bounds_error=False, fill_value=0.0)

# frequency grid
f_min, f_max = 1/60, 1.0    # Hz
   = np.linspace(2*np.pi*f_min, 2*np.pi*f_max, 900)
f =   /(2*np.pi)
T = 1/f
idxT = np.argsort(T)

vRAO = np.vstack([cpx(d)(  ) for d in dofs])

```

```
# CoG     boom-tip transform (vector P)
Rao_origin = np.array([95.44, 0, 14.69])
Cranetip_loc = np.array([73.75, -50, 131.09])
P = Cranetip_loc - Rao_origin
Tip = np.array([[1,0,0,       0,  P[2], -P[1]],
                [0,1,0,   -P[2],     0,  P[0]],
                [0,0,1,    P[1], -P[0],   0 ]])

tip_RAO = Tip @ vRAO
u_tip_x = tip_RAO[0]                          # horizontal tip motion

#


# 6. Wave spectrum
#


def jonswap(f, Hs, Tp):

    g  = 9.81
    fm = 1.0 / Tp

       = Tp/np.sqrt(Hs)
    if      <= 3.6:     = 5.0
    elif    <  5.0:     = np.exp(5.75 - 1.15*  )
    else:              = 1.0

       = np.where(f <= fm, 0.07, 0.09)

    #  P i e r s o n Moskowitz  base
    S_pm = (g**2)/(16 * np.pi**4) * f**(-5) * np.exp(-5/4 * (fm/f)**4)

    # JONSWAP hump exponent b
    b = np.exp(- (f - fm)**2 / (2 *   **2 * fm**2))

    # un-normalized JONSWAP
    S_raw = S_pm *   **b

    # normalize so that   S_raw   df = Hs  /16
    m0_raw  = np.trapz(S_raw, f)
     _corr   = (Hs**2/16) / m0_raw
    return   _corr   * S_raw        # [m  /Hz]

#


# 7. Crane Side-load Spectrum
#



Hs_demo, Tp_demo = 1   , 8          # demo sea-state
f_xlim  = (0.00, 0.3)                  # frequency [Hz]
Sf_ylim = (1, 5e8)                   # PSD [kN  /Hz]
T_xlim  = (1, 30.0)                  # period [s]
ST_ylim = (1, 5e8)                   # PSD [kN   s]

H_tip = np.vectorize(H_tip_load)(  )    # [N/m]
H_tot = u_tip_x * H_tip                 # [N]

S_f = jonswap(f, Hs_demo, Tp_demo)

S_Ff   = S_f * np.abs(H_tot)**2        # [N  /Hz]
S_Ff_k = S_Ff / 1e6                    # [kN  /Hz]
S_FT_k = S_Ff_k / (T**2)               # [kN   s]

#


# 8. Final Results Summary Single Jonswap wave
#


# convert        ordinary frequency f [Hz]
f =    / (2 * np.pi)
```

```
336
337 # Spectral moments &  z e r o crossing
338 m0 = np.trapz(S_Ff, f)
339 m2 = np.trapz((f**2) * S_Ff, f)
340 Tz = np.sqrt(m0 / m2)
341 sig = 2 * np.sqrt(m0)  # 2   significant load
342 m0_kN2 = m0 / 1e6     # [kN^2/Hz]
343 m2_kN2 = m2 / 1e6     # [Hz^2 kN ^2]
344
345 #  M o s t probable   m a x load  estimates
346 D_demo = 0.5 * 3600  # 0.5 h
347 D_full = 3.0 * 3600  # 3.0 h
348 def most_probable_max(D):
349     return sig * np.sqrt(0.5 * np.log(D / Tz))
350
351 Fmax_demo = most_probable_max(D_demo)
352 Fmax_full = most_probable_max(D_full)
353
354 # 4) Peak PSD in frequency domain
355 idx_pf       = np.argmax(S_Ff_k)
356 peak_pf_val  = S_Ff_k[idx_pf]
357 peak_pf_freq = f[idx_pf]
358
359 # 5) Spectrum at the 2nd mode frequency
360 T2 = T_modes[1]
361 f2 = 1.0 / T2
362 S2 = np.interp(f2, f, S_Ff_k)
363
364 # --- print everything ---
365 print("\n=== SIDE-LOAD RESULTS SUMMARY ===")
366 print(f"m0 (variance)               = {m0_kN2:.3e} kN /Hz")
367 print(f"m2 (2nd moment)             = {m2_kN2:.3e} Hz   kN  ")
368 print(f"Zero-crossing period Tz     = {Tz:.3f} s")
369 print(f"Significant load (2  )       = {sig/1e3:.3f} kN\n")
370
371 print(" M o s t probable  max loads:")
372 print(f"     Demo   (D={D_demo/3600:.1f} h) = {Fmax_demo/1e3:.3f} kN")
373 print(f"     Full   (D={D_full/3600:.1f} h) = {Fmax_full/1e3:.3f} kN\n")
374
375 print("Modal periods:")
376 for i, Tn in enumerate(T_modes, 1):
377     print(f"  Mode {i}: T = {Tn:5.2f} s  (f = {1/Tn:4.3f} Hz)")
378
379
380 #
381 # Here we move from single spectrum evaluation to grid evaluation
382 #
383
384 #
385 # 9. Build full library of side-load spectra over given Hs/Tp bins
386 #
387 H_tip = np.vectorize(H_tip_load)(  )    # [N/m per m]
388 H_tot = u_tip_x * H_tip                # [N]
389
390 Hs_edges = [
391     (0.0,0.25),(0.25,0.5),(0.5,0.75),(0.75,1.0),
392     (1.0,1.25),(1.25,1.5),(1.5,1.75),(1.75,2.0),
393     (2.0,2.25),(2.25,2.5),(2.5,2.75),(2.75,3.0),
394     (3.0,3.25),(3.25,3.5),(3.5,3.75),(3.75,4.0),
395     (4.0,4.25)
396 ]
397 Hs_vals = np.array([0.5*(h0+h1) for h0,h1 in Hs_edges])
398
399 Tp_edges = [(t, t+1.0) for t in np.arange(2.0, 23.0, 1)]
400 Tp_vals  = np.array([0.5*(t0+t1) for t0,t1 in Tp_edges])
401
402 n_Tp, n_Hs, n_   = len(Tp_vals), len(Hs_vals),   .size
403 S_Ff_grid = np.zeros((n_Tp, n_Hs, n_ ))
```

```
404
405 mask_Hs   = Hs_vals <= 2.5
406 Hs_vals   = Hs_vals[mask_Hs]
407 S_Ff_grid = S_Ff_grid[:, mask_Hs, :]
408
409 for i, Tp in enumerate(Tp_vals):
410     for j, Hs in enumerate(Hs_vals):
411         S_f = jonswap(f, Hs, Tp)
412         S_Ff_grid[i, j, :] = S_f * np.abs(H_tot)**2
413
414 Hs_labels = [f"{h0:.2f}-{h1:.2f}"
415              for (h0, h1), keep in zip(Hs_edges, mask_Hs) if keep]
416 Tp_labels = [f"{t0:.1f}-{t1:.1f}" for t0, t1 in Tp_edges]
417
418 np.save("Tp_vals.npy",    Tp_vals)
419 np.save("Hs_vals.npy",    Hs_vals)
420 np.save("omega.npy",        )
421 np.save("S_Ff_grid.npy",  S_Ff_grid)
422
423
424 #
425 # 10. Compute  m o s t probable  max load over duration D_hours
426 #
427
428 D_hours = 3.0
429 D       = D_hours * 3600
430
431 # Containers
432 n_Tp, n_Hs, _ = S_Ff_grid.shape
433 F_max_grid = np.zeros((n_Tp, n_Hs))
434
435 # Loop over grid
436 for i in range(n_Tp):
437     for j in range(n_Hs):
438         S_Ff = S_Ff_grid[i, j, :]
439
440         m0 = np.trapz(S_Ff, f)
441         m2 = np.trapz((f**2) * S_Ff, f)
442         Tz = np.sqrt(m0 / m2)
443
444         sig    = 2 * np.sqrt(m0)
445         Ncross = D / Tz
446         F_max  = sig * np.sqrt(0.5 * np.log(Ncross))
447
448         F_max_grid[i, j] = F_max / 1e3   #     kN
449
450 df_Fmax = pd.DataFrame(F_max_grid,
451                        index=Tp_labels,
452                        columns=Hs_labels)
453
454 df_Fmax.to_excel("F_max_grid.xlsx", sheet_name="F_max")
455 print("Wrote full matrix to F_max_grid.xlsx")
456
457 #
458 # 11. Unity check: mark workable (1) or not (0)
459 #
460
461 limit_kN = 0.57 * 1200
462
463 unity_grid = (F_max_grid <= limit_kN).astype(int)
464 df_unity = pd.DataFrame(
465     unity_grid,
466     index=Tp_labels,    # e.g. ["2.0-3.0", "3.0-4.0",    ]
467     columns=Hs_labels   # e.g. ["0.0-0.25", "0.25-0.5",    ]
468 )
469
470 df_unity.to_excel("Workability_unity_check.xlsx")
471 print("Wrote unity-check matrix to Workability_unity_check.xlsx\n")
```

```
472
473
474 #
475 # 12. Load Hs/Tp occurrence probabilities from Excel
476 #
477 prob_path  = r"C:\Users\olla\OneDrive - Boskalis\Thesis Olivier Lagas\
        USA_EastCoast_distributions_SFW01_IPformat_02011979.xlsx"
478 sheet_name = "August"
479 df_prob = pd.read_excel(prob_path, sheet_name=sheet_name, index_col=0)
480
481 #
482 # 13. Build a numeric midpoint probability map
483 #
484 import re
485 BIN_RE = re.compile(r"\s*([\d\.]+)\s*[-   ]\s*([\d\.]+)\s*")
486 def parse_bin(lbl):
487     m = BIN_RE.match(str(lbl).strip())
488     if not m:
489         raise ValueError
490     return float(m.group(1)), float(m.group(2))
491
492 prob_map = {}
493 for tp_lbl in df_prob.index:
494     try:
495         t0, t1 = parse_bin(tp_lbl)
496     except ValueError:
497         continue
498     tm = 0.5*(t0 + t1)
499     for hs_lbl in df_prob.columns:
500         try:
501             h0, h1 = parse_bin(hs_lbl)
502         except ValueError:
503             continue
504         hm = 0.5*(h0 + h1)
505         p = df_prob.at[tp_lbl, hs_lbl]
506         if isinstance(p, str) and p.strip().endswith('%'):
507             p = float(p.strip().rstrip('%'))/100
508         else:
509             p = float(p)
510         prob_map[(tm, hm)] = p
511
512 #
513 # 14. Compute weighted workability via numeric midpoint lookup
514 #
515 nTp, nHs = unity_grid.shape
516 weighted = np.zeros_like(unity_grid, dtype=float)
517
518 for i, tp_lbl in enumerate(Tp_labels):
519     t0, t1 = parse_bin(tp_lbl)
520     tm = 0.5*(t0+t1)
521     for j, hs_lbl in enumerate(Hs_labels):
522         h0, h1 = parse_bin(hs_lbl)
523         hm = 0.5*(h0+h1)
524         p = prob_map.get((tm, hm), 0.0)
525         weighted[i,j] = unity_grid[i,j] * p
526
527 workability = weighted.sum()
528
529 print(f"Overall workability: {workability*100:.2f}%")
530
531 df_weighted = pd.DataFrame(
532     weighted,
533     index=Tp_labels,
534     columns=Hs_labels
```

```
)
df_weighted.to_excel("Weighted_workability.xlsx")
print("Wrote weighted workability to Weighted_workability.xlsx")

df_workability_pct = df_weighted * 100.0
out_path = r"C:\Users\olla\OneDrive - Boskalis\Thesis Olivier Lagas\SideLoad_Results.xlsx
    "

with pd.ExcelWriter(out_path, engine="xlsxwriter") as writer:
    # 1) Max load
    df_Fmax.to_excel(writer,
                     sheet_name="MaxLoad_kN",
                     index=True)
    # 2) Unity check
    df_unity.to_excel(writer,
                      sheet_name="UnityCheck_01",
                      index=True)
    # 3) Workability per bin [%]
    df_workability_pct.to_excel(writer,
                                sheet_name="Workability_pct",
                                index=True)

print(f"Wrote all results to {out_path}")


#


# 15. Optimisation through Differential Evolution
#


import numpy as np
import pandas as pd
from scipy.optimize import differential_evolution
from scipy.linalg     import eigh

nominal_params      = {'L1':L1_, 'L2':L2_, 'L3':L3_, 'w3':w3_, 'L4':L4_}
ratio_nom           = (w3_/2) / L3_
nominal_delta       = 2*np.rad2deg(np.arcsin(np.clip(ratio_nom,0,1))) \
                      if 0<=ratio_nom<=1 else np.nan
nominal_ 2           =  2 .copy()
nominal_workability = workability

nom_Fmax     = pd.DataFrame(F_max_grid,    index=Tp_labels, columns=Hs_labels)
nom_unity    = pd.DataFrame(unity_grid,    index=Tp_labels, columns=Hs_labels)
nom_work_pct = pd.DataFrame(weighted*100.0, index=Tp_labels, columns=Hs_labels)

d_nom   = np.sqrt(L3_**2 - (w3_/2)**2)
L_total = L1_ + L2_ + d_nom + L4_

bnds = [(30.0,60.0), (5.0,10.0), (30.0,50.0), (42.0,42.0)]  # L1,L2,L3,w3

L4_min, L4_max       = 10.0, 20.0
min_delta_deg, max_delta_deg = 60.0, 90.0
sin_min = np.sin(np.deg2rad(min_delta_deg/2))
sin_max = np.sin(np.deg2rad(max_delta_deg/2))

def compute_workability(params):
    global L1_,L2_,L3_,w3_,L4_,M,K,C,Mb,Mbb

    L1_,L2_,L3_,w3_,L4_ = (params['L1'],params['L2'],
                           params['L3'],params['w3'],
                           params['L4'])
    params_const[L1],params_const[L2],params_const[L3],\
    params_const[w3],params_const[L4] = (L1_,L2_,L3_,w3_,L4_)
    _ = _vals(params_const)

    M   = np.asarray(f_M(*_vals(params_const)), float)
    K   = np.asarray(f_K(*_vals(params_const)), float)
    C   =   *M +   *K

    Mb  = np.asarray(f_Mb(*_vals(params_const)), float).flatten()
    Mbb = float(  f_Mbb(*_vals(params_const))   )

    H_tip = np.vectorize(H_tip_load)(  )
```

```
    H_tot = u_tip_x * H_tip

    D     = D_hours * 3600
    work = 0.0
    for i, Tp in enumerate(Tp_vals):
        for j, Hs in enumerate(Hs_vals):
            S_f   = jonswap(f, Hs, Tp)
            S_Ff  = S_f * np.abs(H_tot)**2
            m0    = np.trapz(S_Ff, f)
            m2    = np.trapz((f**2)*S_Ff, f)
            Tz    = np.sqrt(m0/m2)
            sig   = 2*np.sqrt(m0)
            Fmax  = sig * np.sqrt(0.5*np.log(D/Tz)) / 1e3
            if Fmax <= limit_kN:
                tm = 0.5 * sum(map(float, Tp_labels[i].split('-')))
                hm = 0.5 * sum(map(float, Hs_labels[j].split('-')))
                work += prob_map.get((tm,hm), 0.0)
    return work

w_nom_via_fun = compute_workability(nominal_params)
print(f"[CHECK] Nominale workability via functie: {w_nom_via_fun*100:.2f}% "\
      f"(origineel: {nominal_workability*100:.2f}%)")

# Objective for DE
def obj_de(x):
    L1_val,L2_val,L3_val,w3_val = x
    d_val  = np.sqrt(max(0.0, L3_val**2 - (w3_val/2)**2))
    L4_val = L_total - (L1_val+L2_val+d_val)
    ratio  = (w3_val/2) / L3_val
    if not (L4_min<=L4_val<=L4_max and sin_min<=ratio<=sin_max):
        return 1.0
    return -compute_workability({
        'L1':L1_val,'L2':L2_val,
        'L3':L3_val,'w3':w3_val,'L4':L4_val
    })

# --- Run DE with seed for reproducablity ---
print("Start differential_evolution...")
result = differential_evolution(
    obj_de, bnds,
    strategy='best1bin',
    maxiter=50, popsize=20,
    tol=1e-3, disp=True,
    polish=True,
    seed=42
)

x_opt        = result.x
best_w       = -result.fun
L1_val,L2_val,L3_val,w3_val = x_opt
_d_val       = np.sqrt(max(0.0, L3_val**2 - (w3_val/2)**2))
L4_val       = L_total - (L1_val+L2_val+_d_val)
ratio_opt    = (w3_val/2) / L3_val
best_delta   = 2*np.rad2deg(np.arcsin(np.clip(ratio_opt,0,1)))
best_params  = {
    'L1':L1_val,'L2':L2_val,
    'L3':L3_val,'w3':w3_val,'L4':L4_val
}

w_opt_via_fun = compute_workability(best_params)
print(f"[CHECK] Optimized workability via func: {w_opt_via_fun*100:.2f}% "\
      f"(DE-result: {best_w*100:.2f}%)")

df_params = pd.DataFrame({
    'Nominaal':  list(nominal_params.values()) + [nominal_delta],
    'Optimized': list(best_params.values())  + [best_delta]
}, index=['L1','L2','L3','w3','L4','delta (  )'])
print("\n=== Parameter Vergelijking ===")
print(df_params.to_string())

print(f"\n=== Workability ===")
print(f"Nominaal via origineel : {nominal_workability*100:.2f}%")
print(f"Nominaal via functie   : {w_nom_via_fun*100:.2f}%")
print(f"Optimized via DE       : {best_w*100:.2f}%")
print(f"Optimized via functie  : {w_opt_via_fun*100:.2f}%")

```

```
for sym,val in best_params.items():
    params_const[globals()[sym]] = val
_ = _vals(params_const)
 2_opt  ,  _opt  = eigh(
    np.asarray(f_K(*_vals(params_const)), float),
    np.asarray(f_M(*_vals(params_const)), float)
)
 n_opt  = np.sqrt( 2_opt  )
df_modes = pd.DataFrame({
    'Tn_nom (s)': 2*np.pi/np.sqrt(nominal_ 2),
    'Tn_opt (s)': 2*np.pi/ n_opt
}, index=[f"Mode {i+1}" for i in range(len( n_opt ))])
print("\n=== Modale Perioden ===")
print(df_modes.to_string())

Mb  = np.asarray(f_Mb(*_vals(params_const)), float).flatten()
Mbb = float(  f_Mbb(*_vals(params_const))   )
H_tip = np.vectorize(H_tip_load)(  )
H_tot = u_tip_x * H_tip

print("\nSchrijf geoptimaliseerde grids naar Excel...")
nT,nH,D = len(Tp_vals), len(Hs_vals), D_hours*3600
Fmax_opt, weighted_opt = np.zeros((nT,nH)), np.zeros((nT,nH))
for i,Tp in enumerate(Tp_vals):
    for j,Hs in enumerate(Hs_vals):
        S_f   = jonswap(f,Hs,Tp)
        S_Ff  = S_f * np.abs(H_tot)**2
        m0,m2 = np.trapz(S_Ff,f), np.trapz((f**2)*S_Ff,f)
        Tz    = np.sqrt(m0/m2)
        sig   = 2*np.sqrt(m0)
        Fmax_opt[i,j] = sig*np.sqrt(0.5*np.log(D/Tz))/1e3
        if Fmax_opt[i,j] <= limit_kN:
            tm = 0.5*sum(map(float,Tp_labels[i].split('-')))
            hm = 0.5*sum(map(float,Hs_labels[j].split('-')))
            weighted_opt[i,j] = prob_map.get((tm,hm),0.0)

df_Fmax_opt     = pd.DataFrame(Fmax_opt,    index=Tp_labels, columns=Hs_labels)
df_unity_opt    = pd.DataFrame((Fmax_opt<=limit_kN).astype(int),
                               index=Tp_labels, columns=Hs_labels)
df_work_pct_opt = pd.DataFrame(weighted_opt*100.0,
                               index=Tp_labels, columns=Hs_labels)

with pd.ExcelWriter(out_path, engine='openpyxl', mode='a') as writer:
    df_Fmax_opt    .to_excel(writer, sheet_name='Opt_F_max')
    df_unity_opt   .to_excel(writer, sheet_name='Opt_UnityCheck')
    df_work_pct_opt.to_excel(writer, sheet_name='Opt_Work_pct')
print(f"Geoptimaliseerde resultaten toegevoegd aan {out_path}")

from scipy.signal import find_peaks

# buiten de functie, vlak boven het plot/ e x p o r t blok
 v       = np.linspace(2*np.pi*0.01, 2*np.pi*1.0, 2000)
f_export =  v  / (2*np.pi)


#


# 16. Compare nominal vs optimized and export data
#



 v  = np.linspace(2*np.pi*0.01, 2*np.pi*1.0, 2000)
f_export =  v  / (2*np.pi)

def recalc_Htip(params):
    for sym, val in params.items():
        params_const[globals()[sym]] = val
    _ = _vals(params_const)

    global M, K, C, Mb, Mbb
    M   = np.asarray(f_M(*_vals(params_const)), float)
    K   = np.asarray(f_K(*_vals(params_const)), float)
    C   =   *M +   *K

```

```
    Mb  = np.asarray(f_Mb(*_vals(params_const)), float).flatten()
    Mbb = float(  f_Mbb(*_vals(params_const))   )

    H = np.abs(np.vectorize(H_tip_load)( v )) / 1e3  # kN/m
    return H

H_nom = recalc_Htip(nominal_params)
peaks_nom,_ = find_peaks(H_nom)
print("Nominal peaks [Hz]:", np.round(f_export[peaks_nom],3))

H_opt = recalc_Htip(best_params)
peaks_opt,_ = find_peaks(H_opt)
print("Optimized peaks [Hz]:", np.round(f_export[peaks_opt],3))

plt.figure(figsize=(8,4))
plt.semilogy(f_export, H_nom, label="Nominal")
plt.semilogy(f_export, H_opt, label="Optimized")
plt.xlabel("Frequency [Hz]")
plt.ylabel("Amplitude [kN/m]")
plt.title("H_tip_load: Nominal vs Optimized")
plt.legend()
plt.tight_layout()
plt.show()

df_Htip = pd.DataFrame({
    "Frequency_Hz"    : f_export,
    "Htip_nom_kN_per_m": H_nom,
    "Htip_opt_kN_per_m": H_opt
})
with pd.ExcelWriter(out_path, engine='openpyxl', mode='a') as writer:
    df_Htip.to_excel(writer, sheet_name="Htip_compare", index=False)
print(f"H_tip_load data ge xporteerd naar sheet 'Htip_compare' in {out_path}")
```

Listing 1: Workability Calculating Script