Jonas Bjørlo

Rig Heave Predictions Incorporated in Drilling Optimization

Master's thesis in Marine Technology
Supervisor: Dong Trong Nguyen
Co-supervisor: Karl Kristian Olsen
June 2023

Master's thesis

NTNU
Norwegian University of Science and Technology
Faculty of Engineering
Department of Marine Technology

NTNU

Norwegian University of
Science and Technology

Jonas Bjørlo

Rig Heave Predictions Incorporated in Drilling Optimization

NTNU
Norwegian University of
Science and Technology

MSC THESIS DESCRIPTION SHEET

Name of the candidate: Jonas Bjørlo

Field of study: Marine Control Engineering

Thesis title (Norwegian):   ---

Thesis title (English):   Rig Heave Predictions Incorporated in Drilling Optimization

Background
During drilling operations from a floating drilling installation, rig motions induce drilling instabilities such as vibrations, unstable weight-on-bit, issues for directional drilling and surge-swab induced pressure oscillations. To ensure an efficient drilling operation, the rig motions must be taken into account by proper motion compensation, as well as in other decision-making tools. A common strategy is to measure the rig motions with a motion reference unit (MRU) in the drilling control system and take control action based on the measurements. To improve performance, the rig motions can be estimated seconds ahead in time from a wave induced response prediction model of the vessel. Knowing the heave motion seconds ahead of time, a controller can therefore adjust drilling parameters based on the predicted future motion to avoid drilling instabilities. The described system requires two parts. The first part, a motion prediction system must be able to provide estimates of the future motion with acceptable accuracy. The second part is a drilling model that contains the context of the drilling operation, and that are able to control the drilling process.

Work description
1. Perform a background and literature review to provide information and relevant references on:
   - Theory behind the wave induced response of a semi-submersible. Response prediction methods for a semi-submersible vessel such as LSTM networks, the ACF method, and their performance when applied to real-time data from a rig.
   - The drilling process and related challenges during a drilling operation. The consequence of narrow pressure margins and substantial rig motions are the risks of fracturing the reservoir, blowouts, and other hazardous events. Outline the relevant challenges with emphasis on the ones related to heave.
   - Functionality of an auto-driller. Selection of drilling parameters such as weight-on-bit (WOB) and rotational speed (RPM) to avoid drilling instabilities and increase the quality of the drilling operation.
   - Theory behind model predictive control (MPC) for optimizing set points satisfying a set of constraints.
   - Relevant data acquisition systems on the rig, such as the marine control system and drilling control system, that are relevant for the implementation of an auto-driller controller.
   - Report the result of the literature review conducted during Fall 2022 into the master thesis document

   Write a list with abbreviations and definitions of terms, explaining relevant concepts related to the literature study and project assignment.

2. Based on the selected response prediction method, develop a controller (auto-driller) to select drilling parameters when knowing the rig motion seconds ahead of time.
   - The response is known seconds (one-two cycles of waves) ahead of time and can be incorporated in a MPC controller to select the best drilling parameters. The selected drilling parameters should suppress vibrations, unstable weight-on-bit and other drilling instabilities.

- An optimization problem should be established based on the known motion ahead of time, and an empirical model of the drilling process. Downhole measurements are used as input to the model.
- Case study: Weight-on-bit (WOB) topside/downhole comparison for the controller simulations with and without rig response predictions as input to the controller.
- Conclude master's thesis and propose further work.

Specifications

The scope of work may prove to be larger than initially anticipated. By the approval from the supervisor, described topics may be deleted or reduced in extent without consequences with regard to grading.

The candidate shall present personal contribution to the resolution of problems within the scope of work. Theories and conclusions should be based on mathematical derivations and logic reasoning identifying the various steps in the deduction.

The report shall be organized in a logical structure to give a clear exposition of background, results, assessments, and conclusions. The text should be brief and to the point, with a clear language. Rigorous mathematical deductions and illustrating figures are preferred over lengthy textual descriptions. The report shall have font size 11 pts., and it is not expected to be longer than 60-80 A4 pages, from introduction to conclusion, unless otherwise agreed upon. It shall be written in English (preferably US) and contain the following elements: Title page, abstract, acknowledgements, thesis specification, list of symbols and acronyms, table of contents, introduction with objective, background, and scope and delimitations, main body with problem formulations, derivations/developments and results, conclusions with recommendations for further work, references, and optional appendices. All figures, tables, and equations shall be numerated. The original contribution of the candidate and material taken from other sources shall be clearly identified. Work from other sources shall be properly acknowledged using quotations and a Harvard citation style (e.g. *natbib* Latex package). The work is expected to be conducted in an honest and ethical manner, without any sort of plagiarism and misconduct. Such practice is taken very seriously by the university and will have consequences. NTNU can use the results freely in research and teaching by proper referencing, unless otherwise agreed upon.

The thesis shall be submitted with a printed and electronic copy to the main supervisor, with the printed copy signed by the candidate. The final revised version of this thesis description must be included. The report must be submitted according to NTNU procedures. Computer code, pictures, videos, data series, and a PDF version of the report shall be included electronically with all submitted versions.

Start date: 11 January, 2023   **Due date:**   11 June, 2023

Supervisor: Dong Trong Nguyen
Co-advisor(s):   Karl Kristian Olsen

Trondheim, _____

Digitally signed by
Dong Trong Nguyen
Date: 2023.06.08
17:57:02 +02'00'

Dong Trong Nguyen
Supervisor

Abstract

The oil and gas industry are under constant development to meet the energy needs of the world's population. As resources gradually get harder to recover, it is important to improve the efficiency of the operations to stay competitive. An offshore drilling operation is complex, and the trend is to increasingly move into harsher, deep-water areas to explore for, and potentially produce hydrocarbons. For water-depths surpassing 120 meters, a floating drilling installation must be used to drill the well. A floating drilling installation is influenced by wave-induced forces that can cause significant heave, pitch, and roll motions. The rig motions can disturb the drilling operation significantly if not properly compensated for.

This thesis presents a Long-Short Term Memory (LSTM) artificial neural network for predicting the heave motions of a semi-submersible drilling unit seconds ahead of time using measurements from a Motion Reference Unit (MRU). The model is trained based on real heave data from a rig operating in the North Sea. A Model Predictive Controller (MPC) takes in the heave predictions and solves an optimization problem based on a rate of penetration (ROP) model to steer and stabilize the ROP to the desired value for floating drilling installations subject to wave-induced motions. The coefficients of the ROP model are estimated in real time through using a least squares parameter estimation technique.

The system is implemented using Python libraries that are available to the public. The controller performance is evaluated through two different simulation scenarios. First, through observing parameter recommendations in open loop based on data from a real well that has been drilled in the North Sea. Then, through connecting the controller to a drilling simulator to assess the closed loop performance. Simulation results indicate that the MPC can compensate for the predicted wave-induced motions by adjusting the drilling parameters in real-time. Employing a LSTM neural network to predict heave motions seconds ahead of time is a promising strategy based on the simulation results presented in this thesis. Accurate heave predictions are a prerequisite for the proposed control strategy to work.

Sammendrag

Olje- og gassindustrien er under konstant utvikling for å møte energibehovene til verdens befolkning. Ettersom ressursene gradvis blir vanskeligere å utvinne, er det viktig å øke effektiviteten for å holde seg konkurransedyktig. En offshore boreoperasjon er kompleks og trenden er i økende grad å bevege seg inn i mer tøffe, dype vannområder for å lete etter, og potensielt produsere hydrokarboner. For vanndybder over 120 meter må det brukes en flytende boreinstallasjon for å bore brønnen. Flytende borerigger påvirkes av bølgeinduserte krefter som kan forårsake betydelig hiv, stamp og rull bevegelser. Riggbevegelsene kan forstyrre boreoperasjonen betydelig hvis det ikke blir korrekt kompensert for.

Denne oppgaven presenterer et kunstig nevralt nettverk med lang korttidshukommelse (LSTM) for å forutsi hiv-bevegelsene til en halvt nedsenkbar boreenhet sekunder på forhånd ved bruk av målinger fra en bevegelsessensor (MRU). Modellen er trent basert på reelle hivdata fra en rigg som opererer i Nordsjøen. En prediktiv regulator (MPC) tar inn hivprediksjonene og løser et optimaliseringsproblem basert på en modell av penetrasjonshastigheten (ROP) for å styre og stabilisere ROP til ønsket verdi for flytende boreinstallasjoner under påvirkning av bølgeinduserte bevegelser. Koeffisientene i ROP modellen estimeres i sanntid ved å bruke en minste kvadraters parameterestimeringsteknikk.

Systemet er implementert ved hjelp av Python-biblioteker som er tilgjengelige for allmennheten. Kontrollerens ytelse evalueres gjennom to forskjellige simuleringsscenarier. For det første scenarioet observeres parameteranbefalinger i åpen sløyfe basert på data fra en ekte brønn som er boret i Nordsjøen. Deretter, for å vurdere lukket-sløyfe-ytelsen til regulatoren, kobles den til en boresimulator for å studere utviklingen i ROP. Simuleringsresultater indikerer at MPC regulatoren kan bruke hivprediksjoner til å justere boreparametere for å undertrykke effekten av de bølgeinduserte bevegelsene. Å bruke et LSTM kunstig nevralt nettverk til å predikere hiv-bevegelser flere sekunder frem i tid, er en lovende strategi basert på de resultatene som er presentert i denne oppgaven. Nøyaktige hiv-prediksjoner er en forutsetning for at den foreslåtte reguleringsstrategien skal fungere.

Preface

This thesis concludes the course *TMR4930 - Marine Technology, Master's Thesis*, and therefore the master's degree at the Norwegian University of Science and Technology (NTNU). The work has been carried out during the spring semester in 2023 and is a continuation of the specialization project *TMR4510 - Marine Control Systems, Specialization Project* carried out during the autumn semester in 2022. The literature review and background theory in Chapter 2 and 3 are reproduced from the work carried out in the specialization project.

AkerBP and Odfjell Drilling have been generous and provided both drilling data and MRU data from a drilling operation in the North Sea, and this is highly appreciated. The codebase that have been developed in this thesis is entirely written in Python, using libraries including the machine learning platform TENSORFLOW and convex optimization platform CVXOPT that are open to the public.

I want to thank my supervisor Dong Trong Nguyen for providing valuable feedback during the last year. I also want to give a special thanks to Karl Kristian Olsen for being my co-supervisor. Your experience and knowledge within the field of drilling automation have been of high value.

Jonas Bjørlo
Trondheim, June 2023

Table of Contents

List of Tables

List of Figures

Nomenclature

List of Abbreviations

BHA   Bottomhole Assembly

ECD   Equivalent Circulating Density

DRS   Drum Rotation Speed

GD   Gradient Descent

ID   Inner Diameter

LSTM   Long Short-Term Memory

LWD   Logging While Drilling

MPC   Model Predictive Control

MRU   Motion Reference Unit

MWD   Measurement While Drilling

OD   Outer Diameter

PDC   Polycrystalline Diamond Compact

RPM   Revolutions per Minute

ROP   Rate of Penetration

SGD   Stochastic Gradient Descent

SPP   Stand Pipe Pressure

UCS   Unconfined Compressive Strength

WOB   Weight on Bit

1

Introduction

The demand for oil and gas is high and will continue to be a very important source for energy in many years to come. Offshore exploration and production activities are increasingly moving into waters where environmental conditions are harsher. Some prospects have a complex geology to drill, in addition to the harsh environment encountered on the surface. Strategies that can improve the efficiency and safety of harsh environment drilling operations are important to be able to recover the resources cost-efficiently and without any incidents. The drilling operation is simpler to execute if the wave-induced response of the floating drilling installation is kept to its minimum. The rig motions are a disturbance to the drilling process and are also challenging the integrity of the wellhead and the marine riser.

Semi-submersible drilling units, like the one shown in Figure 1.1, have been the preferred choice for drilling in harsh environments for many years. A very low ratio between the waterplane area and mass gives great seakeeping capabilities over the frequencies where the incident waves contain the most energy. Even with the perfect design, the rig motions will be substantial under many sea states in the North Sea. The next step in improving operations on harsh environment semi-submersible drilling units is to use the predicted future motions in the control systems onboard to be able to drill more efficiently. Through recent advances in drilling technology, the availability of downhole properties measured in real-time have increased. This motivates the development of more sophisticated controllers to use in the drilling operation.

The thesis is done in collaboration with Aker BP, which have provided the drilling data used in the thesis. Aker BP have multiple harsh environment semi-submersible drilling units on contract to drill prospects in areas ranging from the North Sea, Norwegian Sea, including the Barents Sea. The response data from a semi-submersible drilling unit, measured by a Motion Reference Unit (MRU), have been provided by Odfjell Drilling.

Figure 1.1: Deepsea Nordkapp (Odfjell Drilling, 2022)

1.1   Motivation

During drilling operations from a floating drilling installation, rig motions induce drilling instabilities such as vibrations, unstable weight on bit (WOB), issues for directional drilling and surge-swab induced pressure oscillations. To ensure an efficient drilling operation, the rig motions must be considered by the drilling control system, as well as in other decision-making tools. A common strategy is to measure the rig motions with an MRU in the drilling control system and take control action based on the measurements.

To improve performance, the rig motions can be estimated seconds ahead in time from a wave induced response prediction model of the vessel. Knowing the heave motion seconds ahead of time, a controller can therefore adjust drilling parameters based on the predicted future motion to avoid drilling instabilities. The described system requires two parts. The first part, a motion prediction system must be able to provide estimates of the future motion with acceptable accuracy. The second part is a drilling model that contains the context of the drilling operation, and that can control the drilling process.

1.2   Literature Review

The literature review carried out in the specialization project (Bjørlo, 2022) identified that there had not been done any previous efforts in combining rig heave predictions seconds ahead of time with rate of penetration (ROP) optimization. This section presents relevant publications on wave-induced response predictions and drilling optimization considered relevant for the thesis. The findings related to heave predictions are reproduced from the specialization project (Bjørlo, 2022).

Kommedal (2021) formulated an auto-driller based on a Model Predictive Controller (MPC) for use in drilling operations from a fixed drilling installation. Like this thesis, his work was carried out in collaboration with Aker BP. The concept of linearizing a suitable ROP model and formulating it as a quadratic programming (QP) problem will be adapted in this thesis. It has been the desire to extend on this work in order for the controller

to function for floating drilling installations as well, through the development of a heave prediction module and including the heave predictions in the controller formulation. The MPC in this thesis will also differ by exploring the use of another ROP model, which is the Hareland model (Hareland and Rampersad, 1994). Other efforts related to MPC formulations for use in offshore drilling includes work done by Sui et al. (2013). A nonlinear MPC was formulated based on the Bourgoyne and Young (1974) ROP model to efficiently adjust the WOB and rotational speed (RPM) to achieve the desired ROP.

Relevant work on drilling optimization related to heave motions includes the study by Pastusek et al. (2016). The conclusion from that study was that the auto-driller control system had a significant effect on the dynamic response of the drilling system when not properly tuned. Also, that the heave motion gave large WOB variations overlapping with the torsional period of the system inducing vibrations, stick-slip and other instabilities.

To drive the development of new drilling automation solutions, Gravdal et al. (2021) shared learnings from the OpenLab Drilling Simulator at NORCE. It is a high-fidelity drilling simulator open to the public. With an open application programmable interface (API), different real-time drilling optimization objectives can be tested. Such as selection of drilling parameters to achieve a specific rate of penetration (ROP) or pressure control in managed pressure drilling. The OpenLab simulation environment will be used for verification of the implemented system in this thesis.

Guo et al. (2021) presented an approach for predicting the heave and pitch motions of a semi submersible by using neural networks. A Long Short-Term Memory (LSTM) based machine learning model was trained on motion and wave measurements. The training and test data were obtained from a model test in a wave basin. The model was able to predict up to 46.5s into the future with an accuracy of close to 90%. The conclusion was that the proposed LSTM model had a strong ability to predict vessel wave-excited motions, and as a further step it was proposed that the model could be used to predict motions based on only the motion itself.

Nielsen et al. (2018) formulated a prediction method by obtaining the sample autocorrelation function (ACF) for a recent time window of the response. The prediction procedure assumes that the response is characterized by a stationary process and has memory in its behavior. This approach does not require information about the characteristics of the incident wave system. The response was efficiently predicted 8-9 wave periods ahead of time using data from a model test. Takami et al. (2021) proposed to use the instantaneous autocorrelation function to address the problem of the assumption of the non-stationarities of the waves, which will be encountered in irregular waves. The conclusion was that this approach had better prediction accuracy than direct use of the sample autocorrelation function.

In the specialization project the two methods were implemented and compared to each other (Bjørlo, 2022). For this thesis it was decided to go further with the heave prediction method using a LSTM artificial neural network. This was done due to the capabilities of a neural network to learn nonlinear patterns in the wave-induced response, but at the cost of requiring offline training.

1.3   Objective and Scope

The objective of this master's thesis is to formulate a heave prediction model based on MRU measurements only and use the outputs in a MPC to optimize the drilling operation.

A heave prediction module based on a LSTM artificial neural network with a single input feature is formulated. Accuracy of the predictions for various time horizons is assessed. Real heave response data from a semi-submersible operating in the North Sea will be used as input for the module to be implemented.

The MPC should be formulated with a suitable ROP model, such as the Hareland model (Hareland and Rampersad, 1994), together with the predicted heave motions to adjust the WOB and RPM to achieve the desired ROP. A parameter estimation technique should be implemented to capture the relevant drilling dynamics in the ROP model coefficients.

Verification of the system is done through two different operating modes. Advisory mode simulations are based on drilling data from a real well in the North Sea, allowing actual data to be used in the algorithms. Then, simulations in the OpenLab environment are carried out to study the closed-loop performance of the MPC.

1.4   Outline of Thesis

The thesis consists of eight chapters including this introduction. Chapter 2 presents relevant background theory on the response behaviour for a semi-submersible subject to incident waves. A method for predicting future responses with a LSTM artificial neural network is outlined.

Chapter 3 describes the fundamentals of offshore drilling. Different ROP models are presented, and the choice of model serves as the starting point for the controller to be implemented. At the end, the OpenLab environment is described, which will be used for verification of the controller.

Chapter 4 covers the synthesis of the model predictive controller based on the selected ROP model and the predicted heave motion that were presented in the two previous chapters. A least-squares method for estimating coefficients in the ROP model, in real-time, is outlined.

Chapter 5 covers the implementation specific details. First, handling of the drilling data and MRU data are outlined. Then, the implementation specific details of the artificial neural network for heave prediction and the model predictive controller are outlined. In the end, the complete system is summarized.

Chapter 6 presents the results from two different simulation cases, advisory mode and closed-loop mode. The accuracy of the heave predictions is assessed through two simulation cases carried out with different training data.

Chapter 7 discusses the performance of the heave prediction module and the controller. Challenges related to verification of drilling automation solutions in simulations are outlined. The observed strengths and weaknesses with the selected methods are also discussed.

Chapter 8 concludes the thesis and proposes further work actions.

2

Wave-Induced Response Prediction

This chapter will give a brief overview of the behavior of a semi-submersible structure subject to waves. In addition, information on the capabilities of neural networks to predict the response are presented. The content is based on work carried out in the specialization project (Bjørlo, 2022), but with an extension to the section about neural networks.

2.1 Physics of Fluid Motions

To calculate the response of a structure, the pressure acting on the body must be solved for and integrated over the body. The Navier-Stokes equations are the governing equations for fluid motions. The Navier-Stokes equations, given by (2.1), states that the local acceleration and convective acceleration balance the pressure forces, shear forces and gravitational forces of the fluid.

$$\frac{\partial u}{\partial t} + u \cdot \nabla u = -\frac{1}{\rho}\nabla p + g + \nu\nabla^2 u \tag{2.1}$$

where u is the fluid velocity, p is the pressure and ν is the shear rate. But they are very demanding to solve, despite all the work done on improving the numerical solvers. To lower the complexity and get simpler relations that describe the fluid motion, potential flow theory can be used. In most conditions, potential theory is sufficient to compute an accurate prediction of the loads on a large-volume structure like a semi-submersible (DNV, 2010), which is the marine structure of interest in this project.

2.1.1 Potential Flow Theory

A requirement for a potential flow is that the fluid is inviscid ($\nu = 0$), which means that there are no shear forces acting on the fluid. The flow is also assumed to be irrotational ($\nabla \times u = 0$), that means there are no vortices in the flow. By limiting the flow to be irrotational, the fluid velocity u can be expressed with a velocity potential φ such that $u = \nabla\varphi$. The problem is then to find the velocity potential φ that satisfies the Laplace

equation (2.2) in the fluid domain Ω with boundary conditions.

$$\nabla^2 \varphi = 0 \tag{2.2}$$

Figure 2.1 shows the setup of a nonlinear boundary value problem that can be solved to get the velocity potential φ for a semi-submersible in waves.
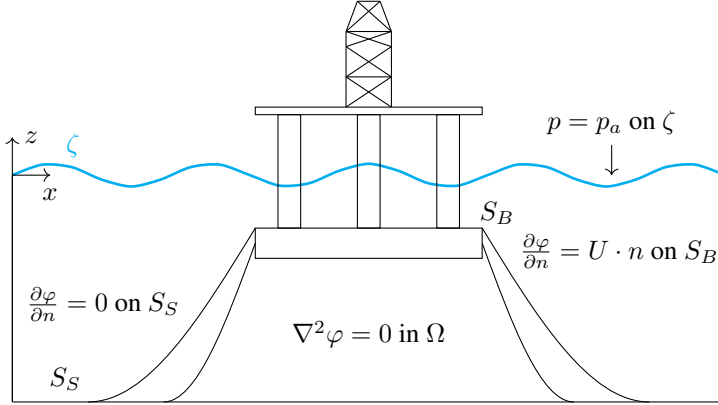
Figure 2.1: Boundary Value Problem Setup

The boundary conditions that must be considered are the kinematic and dynamic boundary conditions, and will be briefly discussed in the next two sections. The indicated boundary conditions in the figure are nonlinear and are required to be solved timestep by timestep. When linearizing the boundary conditions, there exists frequency domain solutions for the problem. The benefit of solving it in frequency domain is that there is only necessary to perform one computation per wave frequency ω. The expressions for the linearized boundary conditions will therefore also be briefly outlined.

Kinematic boundary condition

The kinematic boundary condition states that no fluid can enter through the body boundary S_B or seabed S_S. In terms of the body velocity U, the solid boundary normal vector n and surface S, the kinematic boundary condition is given by (2.3).

$$\frac{\partial \varphi}{\partial n} = U \cdot n \quad \text{on } S \tag{2.3}$$

A kinematic boundary condition must also be applied for fluid particles on the free surface since a fluid particle on the free surface is assumed to stay on the free surface.

$$\frac{\partial \zeta}{\partial t} + \frac{\partial \varphi}{\partial x}\frac{\partial \zeta}{\partial x} + \frac{\partial \varphi}{\partial y}\frac{\partial \zeta}{\partial y} - \frac{\partial \varphi}{\partial z} = 0 \quad \text{on } z = \zeta(x, y, t) \tag{2.4}$$

The kinetic free surface boundary condition is nonlinear due to that it contains the free surface ζ, which is unknown before the problem is solved.

Dynamic free-surface condition

The dynamic free-surface condition states that on the free surface $\zeta(x, y, t)$, the water pressure p is equal to the atmospheric pressure p_a. The Bernoulli equation relates the pressure p to the fluid velocity, given by (2.5).

$$p + \rho g z + \rho \frac{\partial \varphi}{\partial t} + \frac{\rho}{2} \nabla \varphi \cdot \nabla \varphi = C \tag{2.5}$$

where ρ is the fluid density, z is the vertical height and C is an arbitrary constant. By selecting a constant $C = p_a/\rho$ the dynamic free surface condition is obtained

$$g\zeta + \frac{\partial \varphi}{\partial t} + \frac{1}{2} \left(\left(\frac{\partial \varphi}{\partial x} \right)^2 + \left(\frac{\partial \varphi}{\partial y} \right)^2 + \left(\frac{\partial \varphi}{\partial z} \right)^2 \right) = 0 \quad \text{on } z = \zeta(x, y, t) \tag{2.6}$$

The dynamic free surface boundary condition is also nonlinear due to that the free surface ζ is not known before the problem is solved.

Linearized free-surface conditions

The free-surface conditions are linearized by doing a transformation from the instantaneous free-surface at $z = \zeta(x, y, t)$ to the mean free surface at $z = 0$. The linearized kinematic free surface condition is then given by (2.7), while the linearized dynamic free surface condition is given by (2.8).

$$\frac{\partial \zeta}{\partial y} = \frac{\partial \varphi}{\partial z} \quad \text{on } z = 0 \tag{2.7}$$

$$g\zeta + \frac{\partial \varphi}{\partial t} = 0 \quad \text{on } z = 0 \tag{2.8}$$

2.1.2   Linear Wave Theory

The most general solution to the linear problem is when there are no body present in the fluid domain, and that the free surface is of infinite horizontal extent. The solution to the problem gives a velocity potential φ and free surface elevation ζ according to (2.9) and (2.10), reproduced from (Faltinsen, 1990).

$$\varphi(x, z, t) = \frac{g\zeta_a}{\omega} \frac{\cosh k(z + h)}{\cosh kh} \cos(\omega t - kx) \tag{2.9}$$

$$\zeta(x, t) = \zeta_a \sin(\omega t - kx) \tag{2.10}$$

where k is the wave number and ω is the circular wave frequency. Once the wave velocity potential φ has been solved for, the dynamic pressure term p_D and velocity components u and w can be found in both space and time with (2.11) - (2.13).

$$p_D(x, z, t) = \rho g \zeta_a \frac{\cosh k(z + h)}{\cosh kh} \sin(\omega t - kx) \tag{2.11}$$

$$u(x, z, t) = \omega \zeta_a \frac{\cosh k(z + h)}{\sinh kh} \sin(\omega t - kx) \tag{2.12}$$

$$w(x, z, t) = \omega \zeta_a \frac{\sinh k(z + h)}{\sinh kh} \cos(\omega t - kx) \tag{2.13}$$

For deep water depths, the solution is found by taking the limit when $kh \longrightarrow \infty$.

2.1.3 Irregular Waves

The linear wave equation and physical quantities presented previously applies for a single wave component with amplitude ζ_a and frequency ω. Irregular waves can be described as the sum of N different regular wave components of different amplitude A_j and frequency ω_j shifted with a uniformly distributed phase angle ϵ_j

$$\zeta = \sum_{j=1}^{N} A_j \sin(\omega_j t - k_j x + \epsilon_j) \tag{2.14}$$

The next question is to determine which amplitudes and frequencies that are relevant. To describe waves in the North Sea, the JONSWAP wave spectrum $S(\omega)$ can be used.

$$S(\omega) = 155 \frac{H_s^2}{T_1^4 \omega^5} \exp\left(\frac{-944}{T_1^4 \omega^4}\right) (3.3)^Y, \quad Y = \exp\left(-\left(\frac{0.191\omega T_1 - 1}{\sqrt{2}\,\sigma}\right)^2\right) \tag{2.15}$$

where T_1 is a characteristic period of the spectra, H_s is the significant wave height and σ is a shape factor. Figure 2.2 shows the JONSWAP wave spectrum $S(\omega)$ along the vertical axis, the regular wave components derived from the spectrum in black propagating along the time axis, and the resulting irregular wave realization in red. This figure clearly demonstrates the link between the frequency domain and time domain representation of waves.
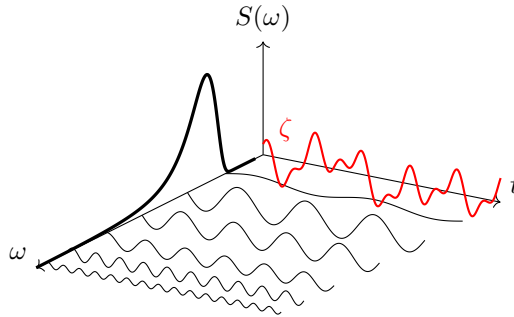
Figure 2.2: Irregular Wave Realisation

2.2 Response Computation

The response in irregular seas is the superposition of the responses due to the regular wave components. Based on linear theory, it will be adequate to study the response in regular waves. The steady state response of a structure will be (Faltinsen, 1990)

$$\eta = \sum_{j=1}^{N} A_j |H(\omega_j)| \sin(\omega_j t + \delta(\omega_j) + \epsilon_j) \tag{2.16}$$

where $|H(\omega_j)|$ is the transfer function between the incident wave amplitude and the response amplitude, also known as a Response Amplitude Operator (RAO). The transfer function can be calculated in a software like WAMIT or WADAM. The result implies that the response will oscillate at the same frequency as the incident wave frequency ω_j, only shifted with a phase δ. It should be mentioned that this assumption only is valid for linear conditions, the structure will be observed to oscillate at other frequencies due to nonlinear effects.

Figure 2.3 shows a typical RAO in heave for a semi-submersible drilling rig. The heave RAO indicates that incident waves with high frequency ($0 < T < 7$ s) will result in minor heave motions. For swell, that have a low frequency, the motion amplification factor is significant. But the energy content in long waves is low, such that the resulting heave motion will be limited. This is a major benefit that drilling rigs with a semi-submersible hull have, since the heave motion is disturbing the drilling process and should be kept to a minimum.
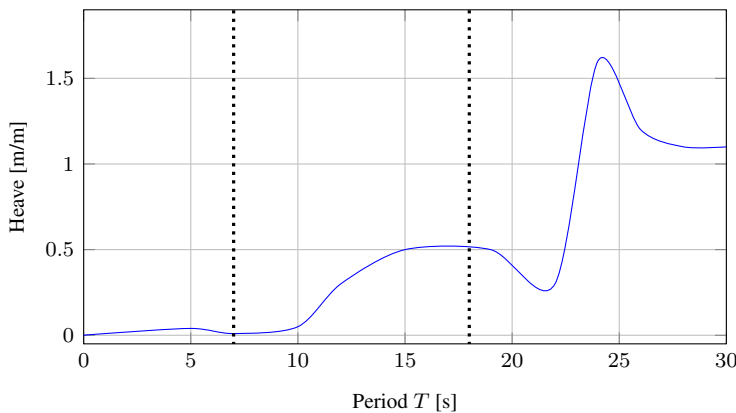
Figure 2.3: Typical RAO for a semi-submersible

A strategy that uses RAOs require input about the exciting wave system to estimate the response. In addition, the RAOs must be known beforehand, or estimated during operation. The scope of this thesis is to employ a strategy that can accurately predict the heave motion to improve the drilling process, and a method using neural networks will be presented.

2.3 Response Prediction Using Neural Networks

In the specialization project (Bjørlo, 2022), two different methods for heave prediction using the autocorrelation function (ACF) and a LSTM neural network were implemented and compared against each other. The heave prediction method utilizing neural networks will be further elaborated.

2.3.1 Fundamentals of Neural Networks

An artificial neural network (ANN) is a data processing system that is inspired by the human brain. An ANN is made up of multiple processing elements, named neurons, that are connected to each other with an associated weight. Application areas for ANN include pattern classification, function approximation, optimization, and time-series prediction, where the last is of interest in this thesis.

A neural network can be characterized based on three properties (Fausett, 1993), summarized below. The content is adapted from da Silva and Spatti (2016).

- **Architecture**: An ANN can be divided in three categories of layers as shown in Figure 2.4. The input layer receives a specific number of features, or measurements, from the environment of interest. The hidden layer consists of neurons performing the internal processing that extracts patterns from the system. The output layer also consists of neurons and produce the final network outputs in a desired format. The main architectures are mesh networks, single-layer feedforward networks, multilayer feedforward networks and recurrent networks. The recurrent network architecture will be further elaborated, as it has properties suited for response prediction.

- **Training Procedure**: The training procedure is responsible for determining the weights and biases in the neural network to get the desired behavior. The different training procedures are reinforcement learning, supervised learning, unsupervised learning. The training can also be divided into online and offline (batch) learning. In offline learning, the weight adjustments are performed after the full training set has been made available. In online learning, the adjustments can be made when each individual training sample are available. It is therefore more suited if the dynamics of the system changes rapidly.

- **Activation Function**: Each neuron contains an internal state called the activation level. The activation level is a function of the inputs that have been received. The activation function decides the output from the neuron, given the weighted inputs that have been received. A suitable activation function, denoted $\phi(z)$, can be the sigmoid function $\sigma(z)$ or the $\tanh(z)$ function. The two activation function candidates are shown in Figure 2.5.

Input Layer   Hidden Layer   Output Layer
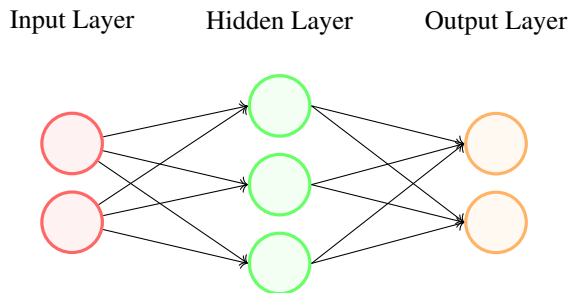
Figure 2.4: Artificial Neural Network Architecture
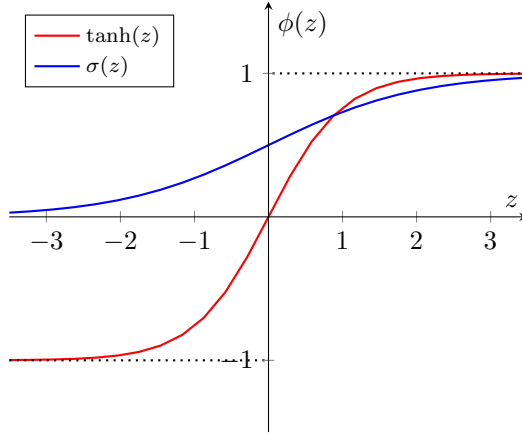
Figure 2.5: Activation Functions (σ and $\tanh$)

2.3.2   Recurrent Neural Network (RNN)

A Recurrent Neural Network (RNN) has properties that make it suitable for learning patterns associated with sequential data. A time-series of the heave response is an example of sequential data, and is the reason behind selecting a RNN for predicting the response of a semi-submersible. In an RNN, the neurons send feedback signals allowing previous outputs being used as inputs. The memory behavior of an RNN is what differs from other neural network architectures where the input and output often are assumed to be independent.

Schmidt (2019) gave a thorough mathematical explanation of RNNs. Denote the input and hidden state at time t as $\mathbf{X}_t \in \mathbb{R}^{n \times h}$ and $\mathbf{H}_t \in \mathbb{R}^{n \times h}$ where n is the number of samples, d is the number of inputs and h is the number of hidden units. Also, denote the input-to-hidden-state weight matrix $\mathbf{W}_{xh} \in \mathbb{R}^{d \times h}$, hidden-state-to-hidden-state weight matrix $\mathbf{W}_{hh} \in \mathbb{R}^{h \times h}$ and a bias vector $\mathbf{b}_h \in \mathbb{R}^{1 \times h}$. Let ϕ be an activation function deciding the output from a neuron in the network. The relations for the hidden state $\mathbf{H}_t$ and output state $\mathbf{O}_t$ are then given by

$$\mathbf{H}_t = \phi_h(\mathbf{X}_t \mathbf{W}_{xh} + \mathbf{X}_{t-1} \mathbf{W}_{hh} + \mathbf{b}_h) \tag{2.17}$$

$$\mathbf{O}_t = \phi_o(\mathbf{H}_t \mathbf{W}_{ho} + \mathbf{b}_o) \tag{2.18}$$

This can be compared to the expressions for regular feed-forward neural networks, given by (2.19)-(2.20). The difference between the two architectures being the ability to take into account previous inputs $\mathbf{X}_{0:t-1}$.

$$\mathbf{H} = \phi_h(\mathbf{W}\mathbf{W}_{xh} + \mathbf{b}_h) \tag{2.19}$$

$$\mathbf{O} = \phi_o(\mathbf{H}\mathbf{W}_{ho} + \mathbf{b}_o) \tag{2.20}$$

2.3.3 Supervised Learning

The goal of the learning process in neural networks is to fit the weights $\mathbf{W}$ and biases $\mathbf{b}$ to minimize a cost function, given the predicted network output $\hat{y}$ and the corresponding label y. The cost function is a function of network weights and biases of all the neurons in all the layers. Salehinejad et al. (2018) outlined the fundamentals of training RNNs which will be presented in this section. For the neural network to be trained, a quadratic loss function $\mathcal{L}$ will be considered

$$\mathcal{L}(\hat{y}^{(j)}, y^{(j)}) = \tfrac{1}{2}||y^{(j)} - \hat{y}^{(j)}||^2 \tag{2.21}$$

such that the scope of the training will be to minimize the quadratic loss function denoted $J(\mathbf{W}, \mathbf{b})$, also known as the Mean Squared Error (MSE), over the entire training set consisting of m samples.

$$J(\mathbf{W}, \mathbf{b}) = \frac{1}{2m} \sum_{j=1}^{m} ||y^{(j)} - \hat{y}^{(j)}||^2 \tag{2.22}$$

The Gradient Descent (GD) optimization method adjusts the weights of the model by moving in the opposite direction of the minimum of the error function derivatives. The gradient is computed for the whole dataset, and provides a single update

$$\theta_{t+1} = \theta_t - \frac{\lambda}{m} \sum_{j=1}^{m} \frac{\partial \mathcal{L}_j}{\partial \theta} \tag{2.23}$$

where λ is the learning rate and θ is a set of parameters. The GD optimization requires to run through all the m training samples to update the parameters. This is a time-consuming process when the training data set is big. To speed up the learning process, a Stochastic Gradient Descent (SGD) method that estimate the gradient on a subset of the data, is used. The ADAM optimizer, summarized in Algorithm 1, is an extended version of a SGD method where estimation of the moments of the gradients is included to adapt the learning rate.

The RNN is processing sequential data, which are a structure through time. The computation of the gradient must be extended through time to train it, through Back Propagation Though Time (BPTT). The strategy of BPTT is to propagate the network back in time, keeping track of the error between the predicted and actual output. Then, the network is rolled out again with updated weights. The network parameters are defined as $\theta = [\mathbf{W}_{hh}, \mathbf{W}_{xh}, \mathbf{W}_{ho}, \mathbf{b}_x, \mathbf{b}_h, \mathbf{b}_o]$. The gradients are written as

$$\frac{\partial \mathcal{L}}{\partial \theta} = \sum_{t=1}^{T} \frac{\partial \mathcal{L}_t}{\partial \theta} \tag{2.24}$$

where the loss function gradient, by the chain rule, is given by

$$\frac{\partial \mathcal{L}_t}{\partial \theta} = \sum_{j=1}^{t} \left(\frac{\partial \mathcal{L}_t}{\partial \mathbf{H}_t} \cdot \frac{\partial \mathbf{H}_t}{\partial \mathbf{H}_j} \cdot \frac{\partial \mathbf{H}_j^+}{\partial \theta} \right) \tag{2.25}$$

where $\frac{\partial \mathbf{H}_j^+}{\partial \theta}$ is the partial derivative. The error can be transported in time from timestep t back to timestep j with the relation

$$\frac{\partial \mathbf{H}_t}{\partial \mathbf{H}_j} = \prod_{i=j+1}^{t} \frac{\partial \mathbf{H}_i}{\partial \mathbf{H}_{i-1}} \tag{2.26}$$

Algorithm 1: ADAM (Kingma and Ba, 2015)

Require: α: Stepsize
Require: $\beta_1, \beta_2 \in [0, 1)$: Exponential decay rates
Require: $f(\theta)$: Stochastic objective function, parameters θ
Require: θ_0: Initial parameter vector
$m_0 \leftarrow 0$
$v_0 \leftarrow 0$
$t \leftarrow 0$
while θ_t not converged **do**
  $\quad t \leftarrow t + 1$
  $\quad g_t \leftarrow \nabla_\theta f_t(\theta_{t-1})$ (Get gradients w.r.t stochastic objective)
  $\quad m_t \leftarrow \beta_1 m_{t-1} + (1 - \beta_1)g_t$ (Update biased first moment estimate)
  $\quad v_t \leftarrow \beta_2 v_{t-1} + (1 - \beta_2)g_t^2$ (Update biased second raw moment estimate)
  $\quad \hat{m}_t \leftarrow m_t/(1 - \beta_1^t)$ (Compute bias-corrected first moment estimate)
  $\quad \hat{v}_t \leftarrow v_t/(1 - \beta_2^t)$ (Compute bias-corrected second raw moment estimate)
  $\quad \theta_t \leftarrow \theta_{t-1} - \alpha \hat{m}_t/(\sqrt{\hat{v}_t} + \epsilon)$ (Update parameters)
end
return θ_t (Resulting parameters)

Using the ADAM optimizer to fit the network parameters θ, where the gradients are computed with Back Propagation Through Time (BPTT) concludes the supervised learning strategy for the neural network to be implemented.

Gradient magnitudes can shrink exponentially when propagating back with time, such that long term dependencies in the data are ignored, namely the vanishing gradient problem. This is a potential issue when training regular RNN. The LSTM unit solves the problem of learning long term dependencies in the data and will be presented in the next section.

2.3.4   Long Short-Term Memory Unit

A LSTM unit is designed to handle the vanishing gradient problem that occurs for RNNs and is therefore well suited to make predictions based on time-series data. Guo et al. (2021) proposed that a LSTM neural network had the potential to predict the motion of a marine vessel based on only the measured motion itself. This approach will be further investigated, and a LSTM neural network will be trained based on heave data from an MRU installed on a semi-submersible.

The LSTM unit structure was first presented by Hochreiter and Schmidhuber (1997) and is visualized in Figure 2.6. The unit consists of an input gate, an output gate and a

forget gate. Given an input sequence $\mathbf{X}_t$, the expressions for the respective gates are given by Schmidt (2019)

$$\mathbf{I}_t = \sigma(\mathbf{X}_t \mathbf{W}_{xo} + \mathbf{H}_{t-1} \mathbf{W}_{ho} + \mathbf{b}_o) \tag{2.27}$$

$$\mathbf{O}_t = \sigma(\mathbf{X}_t \mathbf{W}_{xi} + \mathbf{H}_{t-1} \mathbf{W}_{hi} + \mathbf{b}_i) \tag{2.28}$$

$$\mathbf{F}_t = \sigma(\mathbf{X}_t \mathbf{W}_{xf} + \mathbf{H}_{t-1} \mathbf{W}_{hf} + \mathbf{b}_f) \tag{2.29}$$

where σ is the sigmoid function. The state weight matrices $\mathbf{W}_{xo}, \mathbf{W}_{xi}, \mathbf{W}_{xf} \in \mathbb{R}^{d \times h}$, hidden state matrices $\mathbf{W}_{ho}, \mathbf{W}_{hi}, \mathbf{W}_{hf} \in \mathbb{R}^{h \times h}$ and the bias terms $\mathbf{b}_o, \mathbf{b}_i, \mathbf{b}_f \in \mathbb{R}^{1 \times h}$ are fitted during the training of the model.

Introducing a candidate memory cell $\tilde{\mathbf{C}}_t$, with corresponding weights $\mathbf{W}_{xc} \in \mathbb{R}^{d \times h}$, $\mathbf{W}_{hc} \in \mathbb{R}^{h \times h}$ and bias terms $\mathbf{b}_c \in \mathbb{R}^{1 \times h}$.

$$\tilde{\mathbf{C}}_t = \tanh(\mathbf{X}_t \mathbf{W}_{xc} + \mathbf{H}_{t-1} \mathbf{W}_{hc} + \mathbf{b}_c) \tag{2.30}$$

The new memory $\mathbf{C}_t$ is computed from the previous memory cell content $\mathbf{C}_{t-1}$ and the candidate memory cell $\tilde{\mathbf{C}}_t$, based on how much of the memory to be preserved, given by

$$\mathbf{C}_t = \mathbf{F}_t \odot \mathbf{C}_{t-1} + \mathbf{I}_t \odot \tilde{\mathbf{C}}_t \tag{2.31}$$

where the operator $\odot$ represents the element-wise product. The hidden states $\mathbf{H}_t$ are then given by

$$\mathbf{H}_t = \mathbf{O}_t \odot \tanh(\mathbf{C}_t) \tag{2.32}$$
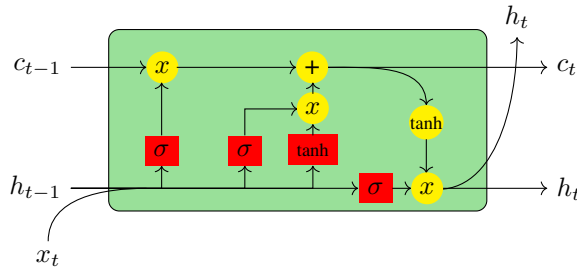
Figure 2.6: LSTM Cell

The LSTM unit maintains a memory such that important features that are detected at an early stage in an input sequence, can be captured and remembered over a long distance. The cell is therefore efficient at capturing long-term features in the input. Patterns developing over multiple timesteps like in a time series can be efficiently predicted with a model consisting of LSTM layers. The reader with an interest in a more comprehensive comparison between LSTM and RNN should check out the paper by Chung et al. (2014).

The TENSORFLOW and KERAS libraries are open source Python libraries for creating and training artifical neural networks (Chollet et al., 2015; Abadi et al., 2015). The LSTM cell is available in the Keras library, which can be utilized when constructing the neural network. The implementation will be discussed in a dedicated section.

When constructing and tuning a neural network, the network must be properly tuned to get good performance. The number of LSTM units will determine if the model is going to underfit or overfit the data. The number of training epochs and size of batch are also important tuning parameters, which will be covered in the next section.

2.3.5   Hyperparameters

A challenge when designing neural networks is to determine the best parameters for the network to solve the problem of interest. The hyperparameters determine the learning capabilities of the network and are tuned for the specific challenge. Below are the parameters to be determined listed, with a desciption of how the parameters effect the performance of the network. Here, batch training is considered, weights and biases are updated after all the inputs are passed to the network. This is opposed to the incremental training approach, where the weights are updated each time an input is passed to the network. The content in this section is based on considerations by Brownlee (2023).

Sequence length

The sequence length is the number of entries in the input. For instance, if a model should make a prediction based on the evolution of the last 30 samples, the sequence length should be 30 samples. It should be long enough to capture the relevant sequential patterns in the system. But at the same time, not so long such that non-relevant dynamics are included in the prediction model. To get an early estimate of the relevant frequencies in the data that should be covered, the Fast Fourier Transform can be applied to the data prior to training. Given an input sequence $x_n = [x_0, x_1, \ldots, x_{N-1}]$, the Fast Fourier Transform will generate a new sequence $X_k = [X_0, X_1, \ldots, X_{N-1}]$ in the frequency domain representing the frequencies that are present in the input.

$$X_k = \sum_{n=0}^{N-1} x_n e^{-i2\pi kn/N} \tag{2.33}$$

where $k = 0, 1, \ldots, N - 1$. Another useful property to calculate is the autocorrelation function R, which indicates similarities between the time-series and a delayed version of itself as a function of the delay τ.

$$R(\tau) = \frac{1}{N - \tau} \sum_{n=0}^{N-1} x(n)x(n + \tau) \tag{2.34}$$

Number of hidden layers and nodes

The optimal number of hidden layers will depend on the complexity of the prediction model, which is determined by knowledge of the dynamics in the model or monitoring how the learning progresses for different setups. Increasing the number of layers means that the model have more parameters available, thus it can learn more complex patterns. The same applies to increasing the number of nodes. For models with multiple features, adding more layers can be necessary to be able to learn the patterns. But for models with a

single feature, adding multiple hidden layers and nodes can lead to overfitting the data. A more comprehensive explanation on the selection of hidden layers and nodes is presented in the paper by Stathakis (2008).

Number of epochs

The number of epochs determines how many times the entire training data set is worked backwards and forwards through the network. The selection will govern how many times the weights and biases in the network are updated during training and can be any value in the range $1 - \infty$. Afaq and Rao (2020) studied the selection of an optimal number of epochs, and concluded that it should be based on the evolution of the training loss and validation loss. The training loss is a metric for how well the model is fitting the training data set. The validation loss is a metric for how well the model fits new data from the validation data set. Monitoring the learning curves during training gives an indication whether the model is a good fit, or if it is underfitting or overfitting the training data.

Batch size

The number of batches make up an epoch and determines the number of input samples that must be processed before updating the network weights. If the entire training data set is run through before updating, then the batch size is equal to the number of samples. For that approach the estimation of the gradient will be accurate, but at the cost of high computational load and significant memory requirements. The other alternative is selecting batch sizes that are smaller than the number of samples, which means that there will be multiple propagations with weight updates in the network. Selecting smaller batch sizes will lead to fluctuations in the gradient, since it is estimated based on a subset of the data, but with the benefit of having more frequent updates to the weights in the network.

Learning rate

The learning rate α determines how much to update the weights in the network based on the estimated error gradient. Selecting a learning rate that is too low can result in an inefficient training process, due to that the altering of the weights are limited. On the other hand, having a too large learning rate can lead to a too aggressive selection of weights resulting in a suboptimal prediction model that does not capture the most relevant dynamics.

The ADAM optimizer included moments that are used to alter the learning rate. It is done by increasing the learning rate when the error cost gradient is heading in the same direction for a long time. The decay rate for the moment estimates, are a parameter that can be selected to optimize the learning rate.

3

Rotary Drilling

Drilling efficiently thousands of meters deep into the ground is a complex operation, requiring highly specialized tools. High pressures and temperatures put the drilling equipment under substantial mechanical and thermal stress. This chapter will give an overview of the topside equipment and downhole tools required in a drilling operation. Drilling related rock mechanics seen in context with hydraulics and drillstring dynamics in order to avoid drilling incidents are discussed next. ROP models that provide a mathematical description of the drilling process are outlined. Relevant drilling automation systems including the auto-driller and advisory systems are covered at the end. The last two sections are based on considerations from the specialization project (Bjørlo, 2022).

3.1 Drilling Rig Equipment

There are many different types of rig setups, depending on factors such as the geographic location, type of well to be drilled and whether the site is located onshore or offshore. The rig equipment described here will try to match the typical setup for a modern harsh environment semi-submersible rig operating in the North Sea. The typical setup includes a top drive system accompanied by an active heave compensated drawworks system (Saipem, 2022; Odfjell Drilling, 2022).

Top Drive System

The top drive system provides the necessary torque to the drillstring to be able to drill the formation efficiently. Some top drive systems are able to provide up to 105,000 ft-lb, or 142 kNm of continuous torque to the drillstring (National Oilwell Varco, 2018). The top drive is suspended from a hook on the travelling block in the derrick, and is free to move vertically up or down. The ability to rotate and move vertically simultaneously is where the top drive differs from the more traditional rotary table. Top drive systems are able to run three joint stands of drillpipe, where rig setups with traditional rotary table are only able to run single joints of drillpipe. A joint of drillpipe is 30 ft, which means that a three
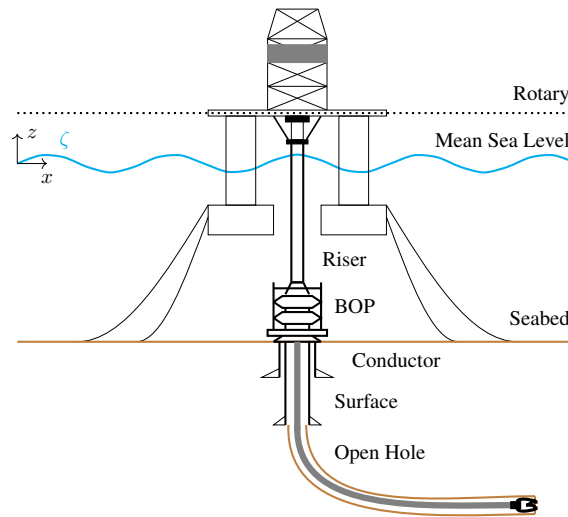
Figure 3.1: Drilling operation from a semi-submersible

joint stand is 90 ft. When tripping in or out of the hole, that is moving the drillstring in or out, making or breaking the connection at each stand is much more efficient then doing it at each joint of drillpipe. When having to trip out, for instance when changing the drillbit, a lot of time will be saved.

The rotational speed (RPM) of the top drive is an important controllable drilling parameter that the driller controls from the chair in the drillers cabin. The RPM is optimized during an operation to avoid vibrations and get the desired ROP.

Dual Active Heave Drawworks

During drilling from a floating drilling installation, the drillstring will follow the vertical motions of the rig if there are no heave compensation active. A heave compensation system ensures that the drill string movement is decoupled from the rig motions, such that the tension on the drill string is constant while drilling. With constant tension in the drillstring, the WOB is kept constant, which is important to avoid instabilities such as vibrations. There are two different categories of heave compensation systems, namely passive and active heave compensation (Woodacre et al., 2015).

The passive heave compensation system uses a pneumatic or hydraulic dampening system to minimize the effect from the rig heave on the drillstring. The passive system has no active controller, nor requires any input power to function. A passive system can typically not reach more than 80% heave decoupling (Hatleskog and Dunnigan, 2006). To reach a higher heave decoupling ratio, which is important in harsh environments, an active heave compensation system must be installed.

An active heave compensation system utilizes rig heave measurements from one or

multiple MRU sensors on the rig floor. Based on the measured heave motion, the draw-works is rotated to keep the drillstring at a fixed position independent of the wave-induced motions of the rig. The active heave compensation system requires a controller, as well as input power. To ensure high hoisting capacity, operational redundancy and safety, the concept of installing two drawworks were developed, known as Dual Active Heave Draw-works (Fivelstad et al., 2014). If a lock-up occurs in one of the drawworks, there will still be active heave compensation due to redundancy. The drawworks position also reflects the amount of weight on bit (WOB) that is being applied. Which again is governing the Rate of Penetration (ROP) being achieved.

Mud Pumps

The mud pumps are used to circulate drilling fluid from topside through the drillstring, into the wellbore, and back up through the wellbore annulus. Due to high frictional losses in the drillpipe and in the annulus when circulating at high flow rates, the pumps must be able to handle high pressures. Being able to circulate at high flow rates is important for cleaning the hole efficiently to avoid accumulation of cuttings in the wellbore. The mud pumps are rated at maximum flow, maximum pressure, and horsepower. The mud pumps are either duplex or triplex positive-displacement pumps, with triplex being used on the modern rigs (Mitchell and Miska, 2011). For a triplex pump, the flow rate is calculated from the stroke per minute of the pumps, and the known displaced volume at each stroke.

Slips

The slips are shaped like a wedge and are used to keep the drillstring suspended in the rig floor. The slips are used when making or breaking a drillstring connection. When the drillstring is suspended in slips, the heave compensation system is not active. For a floating drilling installation, this means that the drillstring will follow the vertical rig motions. The vertical motion of the drillstring will induce swab and surge pressure oscillations that can exhaust the formation downhole.

There exist solutions for a heave compensated rig floor, such as the Heave Compen-sated Floor (HCF) by Huisman. This will eliminate the heave motion of the drillstring during connections, increase the weather window for installation of completions and en-able managed pressure drilling (MPD) from floaters (Huisman, 2019). But for most rigs, the rig floor will not be heave compensated, which means that the drillstring will not be compensated when staying in slips during connections.

Marine Drilling Riser System

The marine riser connects the drilling vessel to the subsea BOP. At the top of the riser, a diverter is responsible for redirecting potential hydrocarbon flows coming from the well when the BOP has not yet been installed, which is typically during drilling the conductor section of the well (API, 1993). Below the diverter, a telescopic joint is installed to make sure the axial load on the riser is kept constant during operation.

Marine riser joints are connected to make up the riser. The riser joints are rated based on diameter, thickness, and material grading (API, 1993). The inside diameter of the riser

must be large enough to run the planned casing through it, and the riser must be strong enough to handle current, rig motions, tension, and internal pressure. The choke and kill lines are placed on the outside of the riser main tube and provides circulation between the well and the vessel when the BOP is closed.

Subsea BOP Stack

The BOP makes it possible to close of the well in case of a well-control incident. A typical well-control incident is a kick, which is an unwanted inflow of hydrocarbons into the wellbore. During a drilling operation from a semi-submersible the BOP is placed at the seabed, such that the well can be shut in and the rig can disconnect safely.

A subsea BOP stack consists of a lower marine riser package (LMRP) and a lower BOP stack. A flex joint connects the marine riser and the LMRP, to allow lateral movement of the rig (Rick von Flatern, 2016). The LMRP typically consists of an annular preventer and a control system. An annular preventer can provide sealing around circular objects such as the drillpipe or BHA. The sealing element is made of hard rubber making it possible to hoist or lower the drillstring, including the tool joints, through the annular preventer.

The lower BOP stack consists of ram-type preventers, the choke and kill lines. The ram-type preventers are designed for providing sealing around a specific diameter of a pipe, but in return has a higher maximum operating pressure than the annular preventer (Mitchell and Miska, 2011). Ram-type preventers that provide sealing around a pipe are named pipe rams, while the ram-types that can seal around an open hole are named blind rams. The choke line is used to redirect flow from the wellbore to the choke manifold. The kill line is used for pumping high density mud into the wellbore from the wellhead.

The integrity of the BOP stack is essential for the safety of a drilling operation in case of an emergency. Rig motions in all degrees of freedom puts a challenge to the integrity of the equipment. Monitoring fatigue and frequent testing of the equipment are carried out during the whole operation.

3.2 Drillstring Components

The drillstring is made up from drillpipe and a bottomhole assembly (BHA). In addition, comes components such as drilling jars, reamers, stabilizers and accelerators. The function of the drillstring is summarized in the list below (Mitchell and Miska, 2011).

- Transmit rotary motion from the topdrive to the drill bit

- Transport drilling fluid to the working face of the bit

- Provide the necessary WOB for the drilling action

- Control the borehole direction

The relevant drillstring components are briefly described in the order from the top to the bottom of the string.

Tool	Properties
Directional	Azimuth, Inclination, Toolface
Drilling Dynamics	WOB, RPM, Torque
PWD	Annular Pressure
GR	Formation Gamma Radiation
Neutron	Formation Porosity
Sonic	Seismic Velocity
Resistivity	Formation Fluid Saturation
Caliper	Borehole Size/Shape

Table 3.1: Common M/LWD Tools

Drillpipe

The upper part of the drillstring is made up of drillpipe. Three joint stands of drillpipe are made up before making a connection to minimize the amount of time in slips. Drillpipes are classified based on the American Petroleum Institute (API) standard (API RP 7G, 1998). The classification is based on pipe dimensions, length, thickness and steel grading. The drillpipe must be able to withstand the high tensions and bending moments that occurs during drilling.

For conventional drillpipe, mud pulse telemetry is used to transfer data from downhole tools to the topside and to transmit control signals down to operate the tools. Due to the limited bandwidth, it is possible to achieve with mud pulse telemetry, wired drill pipe has been developed. The data transfer happens through a data cable protected in a conduit on the inside of the pipe joint. The wired pipe system has demonstrated data transmission rates of up to 2,000,000 bits/sec (Jellison et al., 2003).

Measurements/Logging While Drilling (M/LWD)

The terms Measurement While Drilling (MWD) and Logging While Drilling (LWD) are used interchangeably for the downhole measurement tools used during drilling. Table 3.1 shows some of the most common M/LWD tools used, and the properties that are derived from the tool measurements (Bonner et al., 1993). The selection of tools to run will vary for each section to be drilled and the purpose of the well, based on the data acquisition requirements.

For drilling optimization, mainly the first three tools are of interest. The directional tool is mostly relevant for steering, and the pressure is closely monitored to analyze the well stability. For the section of the well analyzed in this thesis, the DrillDOC tool from Halliburton (2019) was used to measure the downhole drilling dynamics. The downhole measurements of WOB, RPM and torque will provide a much clearer view of the drilling situation than relying solely on topside measurements. Having more accurate measurements are valuable when optimizing the drilling parameters.

Rotary Steerable System (RSS)

The Rotary Steerable System (RSS) is an enabler for efficient directional drilling. The RSS eliminates the need to alternate between sliding with the mud motor and rotating the string to steer. Continuous rotation of the string is favorable for avoiding drilling incidents and have better hole cleaning. The system provides more accurate well placement and higher drilling performance. The RSS is controlled from the surface by issuing commands downhole by downlinking. There exist two categories of RSS, push the bit and point the bit (Sugiura, 2008).

- **Push the bit**: Steering is achieved through external pads on the tool that are adjusted to generate the desired side force against the formation.

- **Point the bit**: An internal drive shaft is bended to tilt the bit to the desired angle.

Drillbit

The purpose of the bit is to conduct the drilling action at the end of the drillstring. Different designs may achieve the drilling action either by scraping, grinding, gouging, or chipping the rock (Mitchell and Miska, 2011). The bits are divided in two categories, fixed-cutter, and roller-cone bits.

- **Fixed-cutter bits**: Fixed-cutter bits have no moving parts, and removes rock by scraping against the formation. The Polycrystalline Diamond Compact (PDC) bit is an example of a fixed-cutter bit where the cutters are made up of synthetic diamond.

- **Roller-cone bits**: Roller-cone bits have rotating cones with teeth that crush the rock when weight is applied to it. Then, the rotational motion of the teeth lifts the drilled cuttings of bottom before the mud flow out from the jets transports the cuttings further up. The cone teeth are selected based on the properties of the formation to be drilled, such as the rock strength.

3.3 Drilling Geomechanics

To plan a well and avoiding drilling problems, knowledge about geomechanics is essential. The wellbore stability analysis determines which mud weight to be used in the drilling operation. The properties of the formation to be drilled determines operational limits such as minimum and maximum ECD limit. The rock strength governs the achievable ROP, and is taken into consideration in the drilling parameter selection. A geological prognosis is made during the planning phase and will be updated as live measurements are available during operations. Geomechanical considerations are inspired by the content in Aadnoy (2011).

Overburden Pressure

The overburden stress σ_v exerted on a formation at depth D is the cumulative weight of the overlying formations. It is computed by integrating the bulk density $\rho_b(z)$ over the

(a) PPFG and WBS

(b) Unconfined Compressive Strength (UCS)
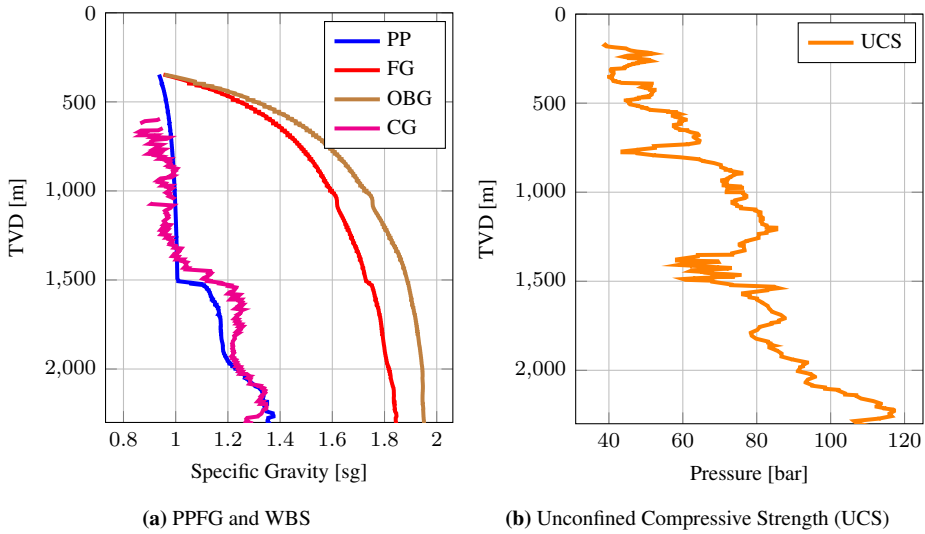
Figure 3.2: Geomechanical Log Curves

depth.

$$\sigma_v = \int_0^D \rho_b(z)g \; dz \tag{3.1}$$

Typical values for the bulk density are between 1.8 g/cm^3 to 2.2 g/cm^3. As the depth increases, the bulk density increases because of compaction of the grains. Figure 3.2a shows a sample overburden pressure gradient colored in brown.

Pore Pressure

The pore pressure is the pressure exerted by the fluids that are contained in the formations. For shallow formations that are normally pressurized, the pore pressure will be equal to the normal pore pressure. The normal pore pressure equals the hydrostatic head at the depth of interest. The pore pressure will often exceed the normal pore pressure due to overpressure in case the fluid has been trapped inside the rock.

Eaton's method 1975 can be used for estimating the pore pressure PP with the velocity data from the Sonic Tool that were presented in the section about MWD Tools.

$$PP = OBG - (OBG - PP_n)\left(\frac{\Delta t_n}{\Delta t}\right)^n \tag{3.2}$$

where OBG is the overburden gradient, PP_n is the normal pore pressure, Δt is the compressional transit time, Δt_n is the compressional transit time for a normally pressurized formation and n is the Eaton exponent.

The pore pressure is a very important parameter to evaluate and study before and during drilling. It serves as a lower limit for the pressure in the well. If the pressure falls

below the pore pressure, it means that hydrocarbons are free to flow into the well, which is called a kick. If this is to happen, it is a well control scenario and the hydrocarbons must be safely circulated out before continuing to drill. Figure 3.2a shows a sample pore pressure gradient colored in blue.

Fracture Pressure

The fracture pressure is the pressure where the formation will start to fracture. Matthew and Kelly (1967) proposed an expression used to estimate the fracture pressure FP

$$FP = k(OBG - PP) + PP \tag{3.3}$$

where k is the effective stress ratio. If the formation has been fractured, mud from the wellbore can start to migrate into the formation. Mud losses into the formation can lead to a well control scenario or a wellbore stability issue. Figure 3.2a shows a sample fracture pressure gradient colored in red.

Unconfined Compressive Strength

The Unconfined Compressive Strength (UCS) is the highest axial stress that a cylindrical shaped rock can withstand before breaking down. It is therefore also referred to as uniaxial compressive strength. The rock strength has a direct impact on the ROP during drilling. Hard stringers, which is areas with abnormally high UCS, are more challenging to drill and must be carefully handled with contingency procedures. Figure 3.2b shows a sample unconfined compressive strength (UCS) log curve.

Collapse Pressure

The collapse pressure is the minimum pressure required to prevent the well from collapsing. The wellbore stability can be calculated with the Mohr-Coloumb criterion when the pore pressure and rock strength have been estimated. Figure 3.2a shows the collapse pressure gradient colored in yellow.

3.4 Drilling Process Modelling

The drilling process consists of drilling a section, running a casing, and cementing it in place. The first step, drilling a section, is the point of interest for the models that are presented.

3.4.1 Wellbore Trajectory

The wellbore trajectory is determined by taking measurements of the azimuth, inclination, and measured depth (MD) at multiple stations along the well path. There is a dedicated MWD tool for directional surveying as described earlier. The directional surveying is typically done for each stand (30 m), but high curvature sections may require more frequent samples (Mitchell and Miska, 2011).

There exist multiple methods to represent the wellbore trajectory based on the survey stations. The minimum curvature method is the preferred method for representing a wellbore trajectory. The North (Δx), East (Δy) and vertical (Δz) distance between two survey stations are given by (Mitchell and Miska, 2011)

$$\Delta x = (\sin I_1 \cos \alpha_1 + \sin I_2 \cos \alpha_2)RF \tag{3.4}$$

$$\Delta y = (\sin I_1 \sin \alpha_1 + \sin I_2 \sin \alpha_2)RF \tag{3.5}$$

$$\Delta z = (\cos I_1 + \cos I_2)RF \tag{3.6}$$

where I_1, I_2 are the inclination angles and α_1, α_2 are the azimuth angles of the two survey stations respectively. RF is the ratio factor, defined by

$$RF = \frac{\Delta s}{\beta} \tan \frac{\beta}{2} \tag{3.7}$$

where Δs is the well path segment connecting the two stations, defined as the difference in the measured depth ΔMD. The dogleg β is defined by

$$\beta = \cos \Big(\cos(I_2 - I_1) - \sin I_1 \sin I_2 (1 - \cos(\alpha_2 - \alpha_1)) \Big) \tag{3.8}$$

Table 3.2 shows a survey program and Figure 3.3 the corresponding trajectory when using the minimum curvature method to calculate the North, East and vertical coordinates.

#	MD [m]	Incl. [^{o}]	Azi. [^{o}]
1	0	0	0
2	374	0	0
3	430	0	0
4	470	2.13	60
5	510	3.13	60
6	540	6.13	60
7	570	10	60
⋮	⋮	⋮	⋮
200	5127	88	48
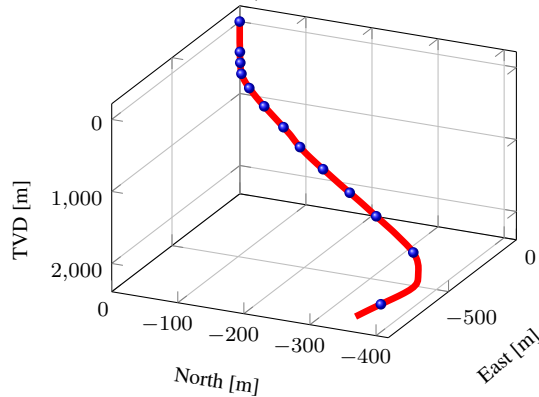
Table 3.2: Survey Program

Figure 3.3: Trajectory

3.4.2 Rate Of Penetration (ROP) Modelling

The drilling speed, often referred to as the rate of penetration (ROP), is an important parameter to optimize during the drilling operation. There are multiple factors that govern the ROP. The rock formation to be drilled, hole diameter, hole cleaning and hydraulics, WOB, RPM and bit type Mitchell and Miska (2011). Some of the factors are controllable, such as the WOB and RPM. But factors such as the rock formation to be drilled is not controllable and must therefore be carefully handled by optimizing the controllable parameters.

ROP Model	Bingham (1964)	Bourgoyne and Young (1974)	Warren (1986)	Hareland and Rampersad (1994)
Operational Parameters				
Weight-On-Bit (W)	✓	✓	✓	✓
Rotational Speed (N_r)	✓	✓	✓	✓
Flow Rate (q)		✓	✓	
Well Depth (h)		✓		
ECD (γ_c)		✓		
Bit Wear (H)		✓		✓
Rock Properties				
Pore Pressure (ρ_p)		✓		
UCS (σ_c)				✓
Bit Properties				
Bit Diameter (d_b)	✓	✓	✓	✓
Bit Nozzle Diameter (d_n)		✓	✓	
PDC Cutters Design				
Number of Cutters (N_c)				✓
Cutter Diameter				✓
Cutter Siderake Angle				✓
Cutter Backrake Angle				✓
Drilling Fluid				
Mud Density (ρ_f, γ_f)		✓	✓	
Viscosity (μ)		✓	✓	

Table 3.3: Required Drilling Parameters per ROP Model, adapted from Soares et al. (2016)

Already during the planning phase some choices that govern the ROP are made, for instance when selecting the drilling fluid, BHA, and bit to use. How the planning phase selections are optimized is not part of the scope of this thesis and will therefore not be further outlined. But some parameters in the models that are presented originates from the mentioned selections.

The controllable parameters to be optimized during operation, are the RPM, WOB and flow rate. In this thesis, the controller is limited to only provide RPM and WOB control inputs. Three different models for *ROP* expressed in form of the controllable parameters are briefly outlined below. Some of the symbols in the equations have been modified to have a consistent set of symbols. The Hareland model will be utilized in the MPC controller to be implemented, due to that it has the most extensive list of parameters to model PDC bits. Therefore, the most emphasis will be put on the Hareland model. The other models are briefly outlined to indicate strengths and weaknesses in the different models, and that accurate ROP modelling is complex involving nonlinearities and multiple parameters.

Bingham Model

Bingham (1964) developed an empirical model that can be applied to both roller-cone and fixed-cutter bits.

$$ROP = a \left(\frac{WOB}{D_b} \right)^b RPM \tag{3.9}$$

where *ROP* is the rate of penetration (ft/hr), *WOB* is the weight-on-bit (klb), *RPM* is the rotational speed (rev/min), d_b is the bit diameter (in), and a and b are dimensionless constants describing the drillability of the formation.

Bourgoyne and Young Model

The Bourgoyne and Young (1974) model is the most comprehensive model in terms of the number of factors taken into account, given by

$$ROP = \text{Exp}\left(a_1 + \sum_{j=2}^{8} a_j x_j\right) \tag{3.10}$$

where a_1 is the formation strength parameter, a_2 is the normal compaction trend exponent, a_3 is the under-compaction exponent, a_4 is the pressure differential exponent, a_5 is a bit weight exponent, a_6 is the rotational speed exponent, a_7 is a tooth wear exponent and a_8 is the hydraulic exponent.

Warren Model

Warren (1987) developed a model for predicting the ROP for roller cone bits in low borehole pressure conditions. The starting point for the model is based on a perfect cleaning model

$$ROP = \left(\frac{a\sigma_c^2 d_b^3}{N_r^b W^2} + \frac{c}{N_r d_b}\right)^{-1} \tag{3.11}$$

where d_b is the bit diameter, σ_c is the rock strength and a, b and c are empirical coefficients. Cuttings removal was incorporated into the perfect cleaning model (3.11) to give the imperfect cleaning model and final expression for the ROP, given by

$$ROP = \left(\frac{a\sigma_c^2 d_b^3}{N_r W^2} + \frac{b}{N_r d_b} + \frac{c d_b \gamma_f \mu}{F_{jm}}\right)^{-1} \tag{3.12}$$

where γ_f is the mud density, μ is the mud viscosity and F_{jm} is the impact force from the jets.

Hareland and Rampersad Model

The Hareland and Rampersad (1994) model considers the rock-bit interaction through deriving the conservation of mass due to the penetration of the cutters into the formation. When applying weight to the bit, the depth of each cutter penetrating the rock is a function of the rock strength, number of cutters, the applied mechanical weight, and the lower projected contact area of each cutter. The projected area at the front of the cutter is given

by

$$A_v = \cos\alpha\sin\theta\left[\left(\frac{d_c}{2}\right)^2\cos^{-1}\left(1 - \frac{4WOB}{\cos\theta\pi N_c d_c^2\sigma_c}\right)\right.$$
$$\left. - \left(\frac{2WOB}{\cos\theta\pi N_c\sigma_c} - \frac{4WOB^2}{(\cos\theta N_c d_c\sigma_c)^2}\right)^{0.5}\left(\frac{WOB}{\cos\theta\pi N_c\sigma_c}\right)\right] \tag{3.13}$$

where α is the cutter siderake angle, θ is the cutter backrake angle, d_c is the cutter diameter, N_c is the number of cutters and σ_c is the unconfined compressive strength. The ROP is equivalent to the amount of rock that the front of the cutters remove, given by

$$ROP = \frac{14.14\ N_s\ RPM\ A_v}{D_b} \tag{3.14}$$

where N_s is the number of cutters and D_b is the diameter of the bit. An empirical correction factor *COR* is introduced to handle phenomena that the theoretical model is not able to handle.

$$COR = \frac{a}{(RPM^b\ WOB^c)} \tag{3.15}$$

where a is an empirical lithology correction factor, b is a bit rotation correction factor and c is a mechanical weight-on-bit correction factor. The correction factor *COR* is multiplied with the *ROP* expression in (3.14) to yield the full expression for PDC bits.

Soares et al. (2016) studied the selection of appropriate bounds on the a, b and c coefficients in the Hareland model in order to fit field data. The results from the study are reproduced in Table 3.4. The upper bound of the a coefficient was selected in the same order of magnitude as the highest observed UCS to balance the division of the rock strength. Lower bounds for RPM coefficient b and WOB coefficient c were set at 0.5, which proved to limit the model errors, even though it limits the relation between RPM and ROP to at most a square root factor. Upper bounds for b at 1.5 and 1 for c resulted in the lowest model errors when fitting the model to field data.

Coefficient	Lower Bound	Upper Bound
a	0.001	10000
b	0.5	1.5
c	0.5	1

Table 3.4: Recommended Bounds on Hareland Model Coefficients

3.5 Drilling Issues

This section briefly presents drilling problems related to vibrations. There also exists other drilling issues such as hole cleaning problems and stuck pipe incidents, but these will not be further elaborated due to not being relevant for the controller to be implemented.

3.5.1   Vibrations

Excessive amounts of vibrations lower the drilling performance. In addition, it causes a lot of stress to the downhole sensors and equipment. Vibration modes such as stick-slip and bit bounce are briefly outlined.

Stick-slip

Stick-slip is a torsional vibration mode where the drillstring torque up and spins free at regular intervals. For this mode, the downhole RPM can be multiple times higher than the surface RPM. The problem typically arises for deep wells and when encountering hard formations. The RPM fluctuations will reduce the quality of the drilling operation in the form of lower rate of penetration (ROP). If the stick-slip severity is too high over a long period of time, there will also be extra wear on the bit and other equipment in the bottom hole assembly (BHA). Sensors will have lower accuracy and the steering capabilities of the BHA will be reduced. The main indication that a stick-slip phenomena is occurring is the observation of cyclic torque fluctuations.

A strategy for reducing stick-slip is to manually decrease the WOB setpoint and increase the RPM, but this will often lead to poor drilling rates. Different practices for preventing stick-slip were reviewed in (Johannessen and Myrvold, 2010), and a model based controller was proposed to control the top drive speed in order to solve the stick-slip problems. (Kyllingstad and Nessjøen, 2009) proposed to use a PI speed controller to dampen the torsional vibrations at the stick-slip frequency. This was done by assuming that the drill string is a transmission line for torsional waves and measuring the stick-slip frequency. By automatically tuning the controller gains based on the stick-slip frequency, the top drive speed controller provides torsional damping.

The heave motion of the rig can provoke stick-slip during drilling. Even with heave compensation systems that are performing well, the WOB variation observed at the surface can reach up to ± 10000 lb (Pastusek et al., 2016). This equates to around 2.4 metric tonnes. The periods of these oscillations will be around the heave motion period. The torque variations at the bit, due to the WOB variations are estimated to represent 3400 ft-lb of torque variations. These WOB and torque variations are by itself enough to induce stick-slip.

Bit bounce

Bit bounce is an axial vibration mode that is linked to the longitudinal motion of the drillstring. The result of bit bounce is that there will be large WOB variations and possibly accompanied by standpipe pressure (SPP) variations. This vibration mode can also be directly linked with the heave motion of a floating drilling rig.

3.6   Drilling Automation

Some of the instabilities that can occur during a drilling operation has been mentioned. This section will cover some software tools that are used to keep the instabilities at a low

level. Decision support systems monitors the operation and provides advice on which drilling parameters to use. Auto-drillers are the low-level controllers that drive the drilling process to its desired setpoints.

3.6.1 Auto-driller

An auto-driller should enable drilling at maximum rate of penetration (ROP) while maintaining a stable drilling operation. It must be able to deal with the frequent changes in rock properties that were described in the section about geomechanics, while at the same time avoid the instabilities mentioned in the section about drillstring dynamics. Badgwell et al. (2018) studied the design and tuning strategies of auto-drillers. The article mentioned that a typical auto-driller consists of a PID-controller that is tuned directly based on the input desired ROP, and will therefore not change gains when the formation is varying, or the controller is unstable. To improve the performance, an enhanced WOB controller was presented. Figure 3.4 shows a typical setup for an auto-driller.
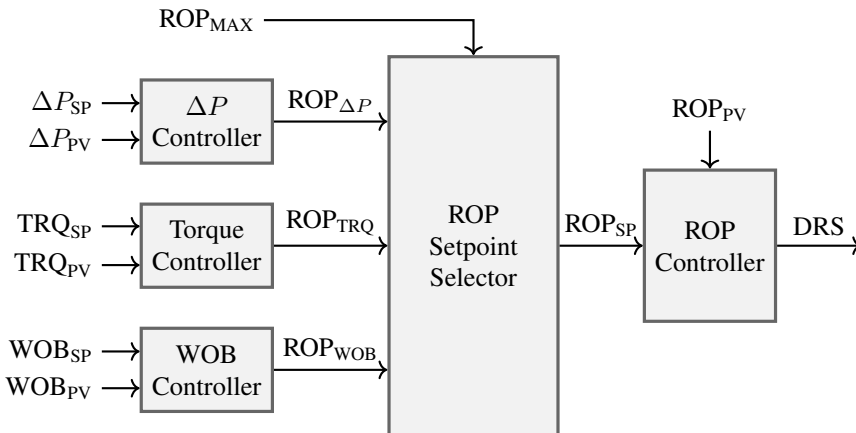

Figure 3.4: Auto-Driller Setup, adapted from (Badgwell et al., 2018)

The WOB, torque, differential pressure ΔP and ROP can be controlled by adjusting the Drum Rotation Speed (DRS). PID controllers compute a desired ROP based on setpoints for differential pressure ΔP_{SP}, torque TRQ_{SP} or weight on bit WOB_{SP}. The output is collected and compared to a maximum limit ROP_{MAX} in a setpoint selector. The most conservative value is chosen from the desired ROP setpoints. If the WOB_{SP} setpoint is used, then the auto-driller is in WOB mode. When the ROP_{MAX} value is used, it is said to operate in ROP mode. The last controller is the ROP controller which adjusts the Drum Rotation Speed to achieve the desired ROP. It should be noted that there exist many different auto-driller designs, and that the one that was presented here is reproduced from (Badgwell et al., 2018).

3.6.2   Decision Support System

The goal of a decision support system in the context of a drilling operation is to provide recommendations for controllable drilling parameters to help improve the drilling process (Payette et al., 2015). It is therefore often referred to as an advisory system. An advice from a decision support system can be to provide a maximum ROP limit, for instance based on hole cleaning monitoring. The ROP limit can then be used as a input for the maximum ROP limit in the auto-driller.

Therefore, a decision support system can be tied in with an auto-driller to achieve automatic, and ultimately autonomous drilling. The determination of setpoints for WOB, torque ROP or differential pressure is equally important as the low-level auto-driller controller to achieve good performance.

3.6.3   OpenLab Drilling Simulator

The OpenLab Drilling Simulator can simulate transient hydraulics, temperature, torque and drag, and cuttings transport (Gravdal et al., 2019). The Python library OPENLAB can be used to set up new wellbore configurations and configure drilling parameters, as well as reading the simulation results. The OpenLab environment is useful when developing and testing new automation solutions, as it provides an interface to efficiently run closed-loop drilling simulations. The OpenLab environment will be utilized to validate the performance of the MPC controller to be implemented in this thesis.

The first step when setting up simulations in OpenLab is to input wellbore configuration parameters. The required hole section information includes the wellbore diameter to be drilled, casing setting depths and dimensions. The well path is configured with a survey program in the same format as shown in Table 3.2. Drilling mud properties including density and viscosity are required, and extra attention should be paid to the mud details if the drilling hydraulics are of interest. BHA properties that must be configured are lengths, weights, and dimensions of the components and this will have impact on the drillstring dynamics calculations. Geological properties that must be defined are the pore pressure and fracture gradient, as well as the rock strength, that were presented in section 3.3. The last set of input is the rig information covering the pump, drawworks and top-drive properties.

Setpoint	Unit
DesiredROP	[m/s]
DesiredWOB	[kg]
SurfaceRPM	[rps]
TopOfStringVelocity	[m/s]
FlowRateIn	[m^3/s]
BopChokeOpening	[%]
ChokeOpening	[%]
WOBAutoDriller	[-]

Table 3.5: OpenLab Setpoints

Result	Unit
HookLoad	[kg]
SurfaceTorque	[Nm]
SurfaceRPM	[rps]
WOB	[kg]
InstantaneousROP	[m/s]
SPP	[bar]
TD	[m]
BitDepth	[m]

Table 3.6: OpenLab Results

Drilling parameters are configured through a set of sequences with setpoints and can

be changed at each timestep in the simulation through the interface. Setpoints that can be configured are shown in Table 3.5. Regular drilling is assumed, which means that the choke opening is set to 100 % during the entire drilling simulation. The output of the controller to be implemented is the WOB and RPM required to reach a specific ROP.

Relevant simulation results that are available each timestep are shown in Table 3.6. The simulation engine considers the inertia of the top drive system and the drawworks, and the actual WOB and RPM that are applied is available. The InstantaneousROP, measuring the achieved drilling performance, can be extracted, and passed on to the controller.

4

Model Predictive Control

This chapter covers the application of Model Predictive Control (MPC) for optimizing the drilling process. First, the essentials of MPC are outlined. The next section covers the formulation of a state-space model for ROP based on work by Sui et al. (2013). Then, the proposed incorporation of the heave predictions and the Hareland model into the MPC are outlined. Linearizing an ROP model and formulating it as a quadratic programming (QP) problem are based on the proposed framework by Kommedal (2021). The last section covers a parameter estimation technique for estimating the coefficients in the ROP model.

4.1 Model Predictive Control (MPC)

The idea of Model Predictive Control (MPC) is to predict the future behavior of a system over a specified prediction horizon and based on the prediction compute an optimal control input. The formulation is done in terms of minimizing a dynamic objective function J on the form (Qin and Badgwell, 2003)

$$J = \sum_{j=1}^{N_p} ||\mathbf{e_y^{k+j}}||^2_{Q_j} + \sum_{j=0}^{M-1} ||\Delta\mathbf{u^{k+j}}||^2_{S_j} + \sum_{j=0}^{M-1} ||\mathbf{e_u^{k+j}}||^2_{R_j} \qquad (4.1)$$

subject to the model constraints

$$\mathbf{x^{k+j}} = f(\mathbf{x^{k+j-1}}, \mathbf{u^{k+j-1}}), \quad \forall j = 1, N_p \qquad (4.2)$$

$$\mathbf{y^{k+j}} = g(\mathbf{x^{k+j}}) + b_j, \qquad \forall j = 1, N_p \qquad (4.3)$$

and subject to the inequality constraints

$$\underline{\mathbf{y}} \leq \quad \mathbf{y^{k+j}} \quad \leq \overline{\mathbf{y}}, \quad \forall j = 1, N_p \qquad (4.4)$$

$$\underline{\mathbf{u}} \leq \quad \mathbf{u^{k+j}} \quad \leq \overline{\mathbf{u}}, \quad \forall j = 0, M-1 \qquad (4.5)$$

$$\underline{\Delta\mathbf{u}} \leq \quad \Delta\mathbf{u^{k+j}} \leq \overline{\Delta\mathbf{u}}, \quad \forall j = 0, M-1 \qquad (4.6)$$

The first term in the objective function penalize the deviation from the desired response, denoted $\mathbf{e_y^{k+j}}$, over the prediction horizon N_p to control the plant output to the desired reference value. The second term penalize rapid changes in the input, denoted $\Delta\mathbf{u^{k+j}}$, over the control horizon M. The third term control input behaviour by introducing input penalties $\mathbf{e_u^{k+j}}$ over the control horizon M. The time-dependent weight matrices Q_j, S_j and R_j are controlling the contribution from each of the terms in the objective function. The model constraints $\mathbf{x^{k+j}}$ and $\mathbf{y^{k+j}}$ represent the predicted states and outputs over the prediction horizon N_p. The inequality constraints on $\mathbf{y^{k+j}}$, $\mathbf{u^{k+j}}$, $\Delta\mathbf{y^{k+j}}$ adds upper and lower bound to the outputs, inputs and the rate of change on the input. The solution to the problem is a set of M input adjustments

$$\mathbf{u_M} = [\mathbf{u^k}, \mathbf{u^{k+1}}, \ldots, \mathbf{u^{k+M-1}}] \tag{4.7}$$

The first input $\mathbf{u^k}$ is appended to the system, and the process is repeated. Figure 4.1 shows a visualization of the concept of MPC, which is that the future output is driven towards the reference trajectory through selecting the optimal input sequence based on the predicted future output.
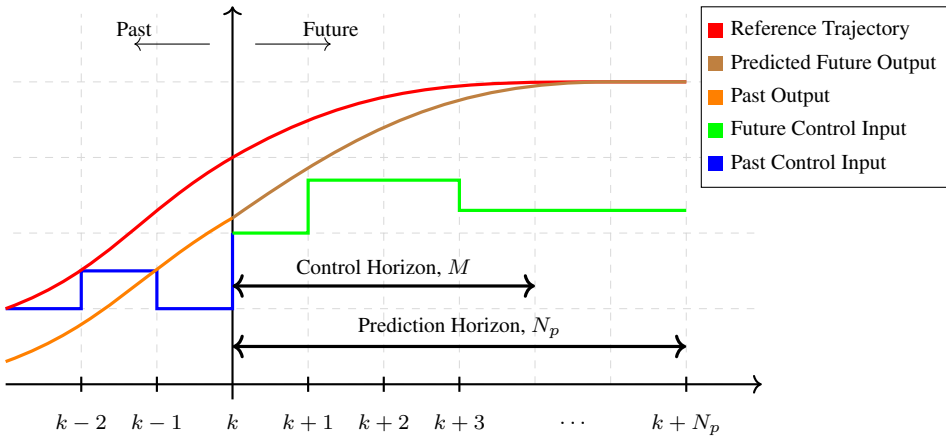

Figure 4.1: Model Predictive Control (MPC) Scheme

The reasoning behind developing and employing a MPC controller for drilling a well are briefly outlined in the list below.

- **Multivariable Optimization**: The drilling process is highly nonlinear and multi-variable, as indicated by the ROP models presented in section 3.4.2. The MPC can find the optimal adjustment to multiple inputs to get the desired output, which is useful for drilling applications where RPM and WOB should be controlled. This is capabilities that controllers such as a regular PI controller does not possess.

- **Constraint Handling**: To drill a well safely and efficient, constraints on the inputs and outputs are defined in operational procedures. A constraint may for instance be that the maximum weight applied to the bit should be 30 tons, to not damage the

bit. For MPC, the constraints on inputs and outputs can be formulated as shown in (4.4)-(4.6).

- **Real-Time Data Availability**: Through the development of wired pipe, vast amounts of information about the drilling process are made available. This has not been the case before and is one of the reasons that more sophisticated control strategies than traditional PI-control have not been deployed.

- **Predictive Ability**: In the advancement towards autonomous drilling, the MPC has the ability to take into account events before they occur. The proactive control action can handle varying formation properties, or as studied in this thesis, mitigate the effects from the wave induced motion of the rig. All while preserving the safety of the operation thorugh constraints on the inputs and outputs.

4.2   State-Space ROP Formulation

The first and fundamental step in designing a robust MPC controller, is to select an appropriate model. The Hareland and Rampersad (1994) model was selected, as it accurately can predict the evolution of ROP when drilling with PDC bits.

Sui et al. (2013) formulated the ROP dynamics in state-space form for the Bourgoyne and Young ROP model. The formulation is adopted and have been modified to apply for the Hareland model. The ROP is the instantaneous slope of the measured depth h, which can be mathematically defined by the differential

$$ROP(t) = \frac{dh}{dt} \tag{4.8}$$

The evolution of the depth h from time t to $t + 1$ are developed using the forward Euler method, which gives the expression

$$h[k + 1] = h[k] + \Delta t ROP[k] \tag{4.9}$$

where Δt is the sampling time. The error $e(t)$ is defined as the difference between the real ROP and the estimated ROP.

$$e[k] = ROP[k] - \widehat{ROP}[k] \tag{4.10}$$

Combining the expressions from (4.9) and (4.10) yields

$$h[k + 1] = h[k] + \Delta t(\widehat{ROP}[k] + e[k]) \tag{4.11}$$

Gathering $h[k]$ and $e[k]$ into the state vector $\mathbf{x}[k]$, which will be denoted $\mathbf{x^k}$ for simplification of notation. The WOB $w[k]$ and the RPM $r[k]$ are gathered in the input vector $\mathbf{u}[k]$, denoted $\mathbf{u^k}$. The vector $\boldsymbol{\alpha^k}$ is defined to express the time dependent factors in the ROP model.

$$\mathbf{x^k} = \begin{bmatrix} h[k] \\ e[k] \end{bmatrix}, \quad \mathbf{u^k} = \begin{bmatrix} w[k] \\ r[k] \end{bmatrix} \tag{4.12}$$

$$\boldsymbol{\alpha^k} = \begin{bmatrix} a[k] & b[k] & c[k] & \sigma_c[k] \end{bmatrix} \tag{4.13}$$

where $a[k]$, $b[k]$, $c[k]$, $\sigma_c[k]$ are the time dependent coefficients that goes into computing $\widehat{ROP}[k]$. The ROP dynamics can then be expressed in terms of the state space model given below

$$\mathbf{x^{k+1}} = f(\mathbf{x^k}, \mathbf{u^k}, \boldsymbol{\alpha^k}) \tag{4.14}$$

$$\mathbf{y^k} = g(\mathbf{x^k}, \mathbf{u^k}, \boldsymbol{\alpha^k}) \tag{4.15}$$

4.2.1 Linearized State-Space Model

Formulating the MPC controller in terms of the nonlinear state-space model of the ROP dynamics requires to solve a constrained nonlinear optimization problem online at each sampling instance. Linearizing the state space model gives a model that can be applied to reduce the computational effort required in the optimization.

A function $f(x)$ can be approximated around a small perturbation $\delta x = x - x^p$ of the operating point x_{op} by taking the Taylor expansion and keeping only the terms that are linear in δx (Balchen et al., 2016).

$$f(x) = f(x_{op} + \delta x) \approx f(x_{op}) + \left.\frac{\partial f}{\partial x}\right|_{x=x_{op}} \delta x \tag{4.16}$$

This approach is applied to the state space model on the form presented in (4.14)-(4.15). The resulting linearized version of the state space model is given by $\mathbf{x_L^{k+1}}$ and $\mathbf{y_L^k}$.

$$\mathbf{x_L^{k+1}} = f(\mathbf{x^k}, \mathbf{y^k}, \boldsymbol{\alpha^k}) \approx f(\mathbf{x_{op}}, \mathbf{u_{op}}) + A_t\delta\mathbf{x} + B_t\delta\mathbf{u} \tag{4.17}$$

$$\mathbf{y_L^k} = g(\mathbf{x^k}, \mathbf{y^k}, \boldsymbol{\alpha^k}) \approx g(\mathbf{x_{op}}, \mathbf{u_{op}}) + C_t\delta\mathbf{x} + D_t\delta\mathbf{u} \tag{4.18}$$

where the linearized state space matrices A_t, B_t, C_t and D_t are given by

$$A_t = \left\{\left.\frac{\partial f_i}{\partial \mathbf{x}_j}\right|_p\right\}, B_t = \left\{\left.\frac{\partial f_i}{\partial \mathbf{u}_j}\right|_p\right\}, C_t = \left\{\left.\frac{\partial g_i}{\partial \mathbf{x}_j}\right|_p\right\}, D_t = \left\{\left.\frac{\partial g_i}{\partial \mathbf{u}_j}\right|_p\right\} \tag{4.19}$$

where $|_p$ means that the differential is evaluated at the operating point. Carrying out this operation on the ROP dynamics based on the Hareland model then yields the linearized state space matrices

$$A_t = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \tag{4.20}$$

$$B_t = \left.\begin{bmatrix} \Delta t\widehat{ROP}J_{u,11} & \Delta t\widehat{ROP}J_{u,12} \\ 0 & 0 \end{bmatrix}\right|_p \tag{4.21}$$

$$C_t = \begin{bmatrix} 0 & 1 \end{bmatrix} \tag{4.22}$$

$$D_t = \left.\begin{bmatrix} \widehat{ROP}J_{u,11} & \widehat{ROP}J_{u,12} \end{bmatrix}\right|_p \tag{4.23}$$

where

$$
\begin{aligned}
J_{u,11} = &\left[707 N_c N_r^{1-b} a \cos(\alpha) \sin(\theta) \left[\sqrt{2} \left(4 N_c \sigma_c \cos(\theta) \sqrt{-\frac{W(-\kappa + 2W)}{\kappa^2}} \right)^{-1} \right. \right. \\
&\left. - \frac{\sqrt{-\frac{2W(-\pi\kappa + 2W)}{N_c \sigma_c \cos(\theta)\kappa}}}{(N_c \sigma_c \pi^2 \cos(\theta))} + \frac{\sqrt{2} W(-\kappa\pi + 4W)}{2 N_c^2 \sigma_c^2 \pi^2 \cos(\theta)^2 \kappa \sqrt{-\frac{W(-\kappa\pi + 2W)}{N_c \kappa \sigma_c \cos(\theta)}}} \right] \right] (50 D_b W^c)^{-1} \\
&- \left[707 N_c N_r^{1-b} ac \cos(\alpha) \sin(\theta) \left[\frac{d_c^2}{4} \left(\pi - \cos^{-1}\left(\frac{-\kappa + 4W}{\kappa} \right) \right) \right. \right. \\
&\left. \left. - \sqrt{2} W \sqrt{-\frac{W(-\kappa\pi + 2W)}{\kappa N_c \sigma_c \cos(\theta)}} (N_c \sigma_c \pi^2 \cos(\theta))^{-1} \right] \right] (50 D_b W^{c+1})^{-1}
\end{aligned}
\tag{4.24}
$$

and

$$
\begin{aligned}
J_{u,12} = 707 a \cos(\alpha) \sin(\theta) &\left(\frac{4\sqrt{2} W}{\pi} \sqrt{-\frac{W(-\kappa\pi + 2W)}{N_c \sigma_c \cos(\theta)\kappa}} \right. \\
&\left. - \kappa\pi \left(\pi - \cos^{-1}\left(\frac{-\kappa + 4W}{\kappa} \right) \right) \right) \frac{b-1}{200 D_b N_r^b W^c \sigma_c \pi \cos(\theta)}
\end{aligned}
\tag{4.25}
$$

where the constant κ is introduced to simplify the expressions, given by

$$
\kappa = N_c \sigma_c \cos(\theta) d_c^2
\tag{4.26}
$$

The terms $J_{u,11}$ and $J_{u,12}$ are evaluated at the operating point, which is the last input $\mathbf{u^{k-1}}$. The bit specific parameters α, θ, N_c, D_b and d_c are static for a BHA run and are given by the specifications from the bit manufacturer. The strategy for finding the coefficients a, b and c in the model are based on fitting the fully nonlinear model to field data and will be further outlined in the next section.

The model is further extended to consider the effects that the wave induced motions of the rig have on the ROP. The desired control action is to gradually provide more weight to the bit when the rig is moving upwards, and the opposite when the rig moves downwards, as shown in Figure 4.2.

Proposing a linear relationship between the heave motions of the rig η_3, the WOB and the resulting change in ROP as given in (4.27). The idea is that during upwards heave motions, the effectiveness of the applied WOB on the ROP is reduced by a factor of $C_\eta \eta_3$, requiring increased WOB to maintain the ROP. During downwards heave motions, the effectiveness of the applied WOB is increased by a factor of $C_\eta \eta_3$, such that slacking off on the weight will be required to keep the ROP.

$$
\delta ROP = -C_\eta \eta_3 \delta WOB
\tag{4.27}
$$

where C_η is a heave correction factor given by (4.28). The heave correction factor considers that the elasticity of the drillstring will dampen the effect of the heave motion when
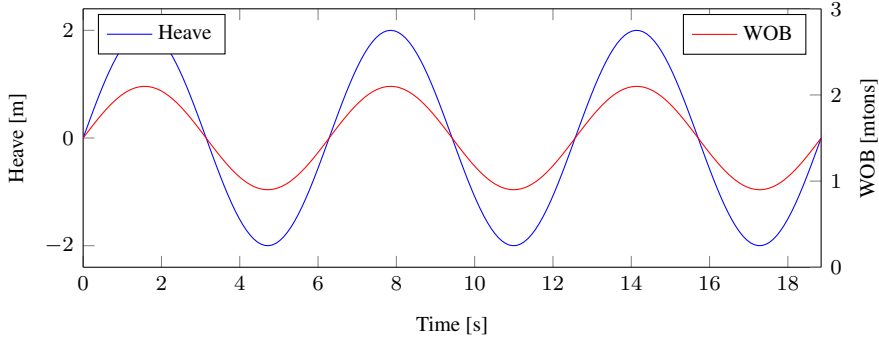
Figure 4.2: Rig Heave and WOB Correction

the length of the drillstring increases.

$$C_\eta = \frac{\xi}{EL} \tag{4.28}$$

where E (MPa) is the elasticity of the drillpipe and L (m) is the length of the drillstring. The empirical coefficient ξ is introduced to account for unmodelled effects such as the type and efficiency of the installed drawworks.

The heave compensated and extended *ROP* dynamics model are then differentiated to yield the additional terms to be added to the original B_t and D_t state space matrices, giving $B_{t,\eta}$ and $D_{t,\eta}$

$$B_{t,\eta} = B_t + \begin{bmatrix} -\Delta t \widehat{ROP} C_\eta \eta_3 & 0 \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} \Delta t \widehat{ROP}(J_{u,11} - C_\eta \eta_3) & \Delta t \widehat{ROP} J_{u,12} \\ 0 & 0 \end{bmatrix} \tag{4.29}$$

$$D_{t,\eta} = D_t + \begin{bmatrix} -\widehat{ROP} C_\eta \eta_3 & 0 \end{bmatrix} = \begin{bmatrix} \widehat{ROP}(J_{u,11} - C_\eta \eta_3) & \widehat{ROP} J_{u,12} \end{bmatrix} \tag{4.30}$$

4.3   Quadratic Programming (QP) Formulation

The strategy for solving the MPC optimization problem numerically is to formulate it on a quadratic programming (QP) format, for which there exists extensive and highly efficient solvers. The general formulation of a QP problem involves minimizing a quadratic objective function with linear constraints on the form given below

$$\begin{aligned} \min_{\mathbf{x} \in \mathbb{R}^n} \quad & \frac{1}{2}\mathbf{x}^\mathsf{T} H \mathbf{x} + f^\mathsf{T} \mathbf{x} \\ s.t. \quad & A\mathbf{x} \leq b \\ & A_{eq}\mathbf{x} = b_{eq} \\ & lb \leq \mathbf{x} \leq ub \end{aligned} \tag{4.31}$$

Defining the evolution of the inputs and outputs over the prediction horizon by $\mathbf{u_k} = \{\mathbf{u^k}, \mathbf{u^{k+1}}, \ldots, \mathbf{u^{k+N_P-1}}\} \in \mathbb{R}^{N_p \cdot n_y}$ and $\mathbf{y_k} = \{\mathbf{y^k}, \mathbf{y^{k+1}}, \ldots, \mathbf{y^{k+N_P-1}}\} \in \mathbb{R}^{N_p \cdot n_y}$.

The reference trajectory for the outputs are denoted by $\mathbf{r_y} \in \mathbb{R}^{N_p \cdot n_y}$. The corresponding errors in the outputs are denoted $\mathbf{e_k^y} = \mathbf{y_k} - \mathbf{r_y} \in \mathbb{R}^{N_p \cdot n_y}$. Input references are denoted $\mathbf{r_u} \in \mathbb{R}^{N_p \cdot n_u}$, with the corresponding input error denoted as $\mathbf{e_k^u} = \mathbf{u_k} - \mathbf{r_u} \in \mathbb{R}^{N_p \cdot n_u}$ The unconstrained control objective can be formulated by keeping the terms penalizing the deviation from desired outputs and inputs given by the first and third term from the formulation in (4.1) to obtain the cost function $J_{y,u}$ given below.

$$J_{y,u} = ||\mathbf{e_k^y}||_{\mathbf{Q}}^2 + ||\mathbf{e_k^u}||_{\mathbf{R}}^2 = \mathbf{e_k^y}^\mathsf{T} \mathbf{Q} \mathbf{e_k^y} + \mathbf{e_k^u}^\mathsf{T} \mathbf{R} \mathbf{e_k^u} \tag{4.32}$$

The development of the tracking error $\mathbf{e_k^y}$ must be derived in order to find $J_{y,u}$. Zhakatayev et al. (2017) formulated the development of the tracking error based on a linearized state space model, which have been adopted in this thesis, reflected in (4.35) and the corresponding matrix form of the development.

Further, Kommedal (2021) formulated the QP problem for an MPC auto-driller to include constraints on the maximum ROP and rate of change in the inputs, which have been adopted in this thesis and are outlined in (4.35)-(4.48).

First, the state-space matrices are rewritten to simplify the notation and gathering constant terms in ϕ in state equation and γ in output equation.

$$\begin{aligned}
\mathbf{x_L^{k+1}} &= f(\mathbf{x_{op}}, \mathbf{u_{op}}) + A_t(\mathbf{x^k} - \mathbf{x_{op}}) + B_{t,\eta}(\mathbf{u^k} - \mathbf{u_{op}}) \\
&= A_t\mathbf{x^k} + B_{t,\eta}\mathbf{u^k} + \phi \\[6pt]
\mathbf{y_L^k} &= g(\mathbf{x_{op}}, \mathbf{u_{op}}) + C_t(\mathbf{x^k} - \mathbf{x_{op}}) + D_{t,\eta}(\mathbf{u^k} - \mathbf{u_{op}}) \\
&= C_t\mathbf{x^k} + D_{t,\eta}\mathbf{u^k} + \gamma
\end{aligned} \tag{4.33}$$

The evolution of the estimated output error $\mathbf{e_k^y} = \{\mathbf{e_y^k}, \mathbf{e_y^{k+1}}, \dots, \mathbf{e_y^{k+N_p-1}}\}$ over the prediction horizon are developed based on the state and output equations in (4.33).

$$\begin{aligned}
\mathbf{e_y^k} &= C_t\mathbf{x^k} + D_{t,\eta}\mathbf{u^k} + \gamma - \mathbf{r^k} \\
\mathbf{e_y^{k+1}} &= C_t\mathbf{x^{k+1}} + D_{t,\eta}\mathbf{u^{k+1}} + \gamma - \mathbf{r^k} \\
&= C_tA_t\mathbf{x^k} + C_tB_{t,\eta}\mathbf{u^k} + D_{t,\eta}\mathbf{u^{k+1}} + C_t\phi + \gamma - \mathbf{r^{k+1}} \\
\mathbf{e_y^{k+2}} &= C_t\mathbf{x^{k+2}} + D_{t,\eta}\mathbf{u^{k+2}} + \gamma - \mathbf{r^{k+2}} \\
&= C_tA_t^2\mathbf{x^k} + C_tA_tB_{t,\eta}\mathbf{u^k} + C_tB_{t,\eta}\mathbf{u^{k+1}} + D_{t,\eta}\mathbf{u^{k+2}} \\
&\quad + C_tA_t\phi + C_t\phi + \gamma - \mathbf{r^{k+2}}
\end{aligned} \tag{4.34}$$

The terms are gathered and structured in matrix form as shown below

$$
\begin{bmatrix}
\mathbf{e_y^k} \\
\mathbf{e_y^{k+1}} \\
\mathbf{e_y^{k+2}} \\
\vdots \\
\mathbf{e_y^{k+N_p-1}}
\end{bmatrix}
=
\begin{bmatrix}
C_t \\
C_t A_t \\
C_t A_t^2 \\
\vdots \\
C_t A_t^{N_p-1}
\end{bmatrix}
\mathbf{x^k}
+
\begin{bmatrix}
D_{t,\eta} & 0 & 0 & \cdots \\
C_t B_{t,\eta} & D_{t,\eta} & 0 & \cdots \\
C_t A_t B_{t,\eta} & C_t B_{t,\eta} & D_{t,\eta} & \cdots \\
\vdots & \vdots & \vdots & \ddots \\
C_t A_t^{N_p-2} B_{t,\eta} & C_t A_t^{N_p-3} B_{t,\eta} & C_t A_t^{N_p-4} B_{t,\eta} & \cdots
\end{bmatrix}
$$

$$
\cdot
\begin{bmatrix}
\mathbf{u^k} \\
\mathbf{u^{k+1}} \\
\mathbf{u^{k+2}} \\
\vdots \\
\mathbf{u^{k+N_p-1}}
\end{bmatrix}
+
\begin{bmatrix}
0 \\
C_t \\
C_t(I + A_t) \\
\vdots \\
C_t(I + \sum_{i=1}^{N_p-2} A_t^i)
\end{bmatrix}
\phi +
\begin{bmatrix}
I \\
I \\
I \\
\vdots \\
I
\end{bmatrix}
\gamma -
\begin{bmatrix}
\mathbf{r_y^k} \\
\mathbf{r_y^{k+1}} \\
\mathbf{r_y^{k+2}} \\
\vdots \\
\mathbf{r_y^{k+N_p-1}}
\end{bmatrix}
$$

For ease of notation, the matrices $P, H, K, I_{N_p} \in \mathbb{R}^{N_p}$ are introduced, which yields the compact form for the ROP tracking error $\mathbf{e_k}$

$$\mathbf{e_k} = P\mathbf{x^k} + H\mathbf{u_k} + K\phi + I_{N_p}\gamma - \mathbf{r_k} \tag{4.35}$$

Constant terms are gathered to form the M matrix, which gives the final expression for the ROP tracking error

$$\mathbf{e_k} = P\mathbf{x^k} + H\mathbf{u_k} + M \tag{4.36}$$

Inserting the expression into the cost function and simplifying to get the $\tilde{H}$ and f^T matrices.

$$
\begin{aligned}
J &= \frac{1}{2}\mathbf{u_k^\mathsf{T}}(H^\mathsf{T}QH + R)\mathbf{u_k} + ((\mathbf{x^k})^\mathsf{T}P^\mathsf{T}QH + M^\mathsf{T}QH - \mathbf{u_k}R)\mathbf{u_k} \\
&= \frac{1}{2}\mathbf{u_k^\mathsf{T}}\tilde{H}\mathbf{u_k} + f^\mathsf{T}\mathbf{u_k}
\end{aligned}
\tag{4.37}
$$

Constructing the $\tilde{H}$ and f^T matrices at each timestep and proceeding to use a QP solver to get the optimal inputs $\boldsymbol{u}$ concludes the unconstrained control strategy.

$$\tilde{H} = H^\mathsf{T}QH + R \tag{4.38}$$

$$f^\mathsf{T} = (\mathbf{x^k})^\mathsf{T}P^\mathsf{T}QH + M^\mathsf{T}QH - \mathbf{u_k}R \tag{4.39}$$

The unconstrained objective is now augmented to include constraints. A slack variable $\mathbf{s}$ is introduced, and the objective function is augmented to include a linear and quadratic slack term with corresponding weight variables w and W.

$$J = \frac{1}{2}\mathbf{x^\mathsf{T}}H\mathbf{x} + f^\mathsf{T}\mathbf{x} + \mathbf{s^\mathsf{T}}W\mathbf{s} + \mathrm{w^\mathsf{T}}\mathbf{s} \tag{4.40}$$

Constraints on the rate of change in the inputs $\Delta\mathbf{u_k}$ are added to avoid significant jumps in the WOB and RPM. Denoting the upper and lower limit to the rate of change in the input as $\underline{\Delta\mathbf{u}}$ and $\overline{\Delta\mathbf{u}}$. At sample time n into the future, the control input development is given by the cumulative sum of input changes $\Delta\mathbf{u^k}, \Delta\mathbf{u^{k+1}}, \ldots, \Delta\mathbf{u^{k+n}}$ and the initial input $\mathbf{u^{k-1}}$.

$$\mathbf{u^{k+n}} = \mathbf{u^{k-1}} + \sum_{i=1}^{n} \Delta\mathbf{u^{k+i}} \tag{4.41}$$

The development of the input $\mathbf{u^k}$ over the entire prediction horizon is represented in matrix form below.

$$
\begin{bmatrix} \mathbf{u^k} \\ \mathbf{u^{k+1}} \\ \vdots \\ \mathbf{u^{k+N_p-1}} \end{bmatrix} = \begin{bmatrix} \mathbf{u^{k-1}} \\ \mathbf{u^{k-1}} \\ \vdots \\ \mathbf{u^{k-1}} \end{bmatrix} + \begin{bmatrix} I & 0 & \dots & 0 \\ I & I & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ I & I & \dots & I \end{bmatrix} \begin{bmatrix} \Delta\mathbf{u_1} \\ \Delta\mathbf{u_2} \\ \vdots \\ \Delta\mathbf{u_{N_p}} \end{bmatrix} \tag{4.42}
$$

Introducing D as the lower triangular identity matrix, such that the input development can be written in compact form by

$$
\mathbf{u^k} = \mathbf{u^{k-1}} + D\Delta\mathbf{u_k} \tag{4.43}
$$

Solving for $\Delta\mathbf{u_k}$ and adding lower bound $\underline{\Delta\mathbf{u}}$ and upper bound $\overline{\Delta\mathbf{u}}$ to the rate of change in the input

$$
\underline{\Delta\mathbf{u}} + D^{-1}\mathbf{u^{k-1}} \leq D^{-1}\mathbf{u_k} \leq \overline{\Delta\mathbf{u}} + D^{-1}\mathbf{u^{k-1}} \tag{4.44}
$$

Defining $A_h = \begin{bmatrix} D^{-1} & -D^{-1} \end{bmatrix}^{\mathsf{T}}$ and $b_h = \begin{bmatrix} \overline{\Delta\mathbf{u}} + D^{-1}\mathbf{u^{k-1}} & -\overline{\Delta\mathbf{u}} - D^{-1}\mathbf{u^{k-1}} \end{bmatrix}^{\mathsf{T}}$ such that the inequality can be written as

$$
A_h\mathbf{u_k} \leq b_h \tag{4.45}
$$

The ROP development is obtained by removing the reference output from the tracking error expression in (4.35).

$$
\mathbf{y_k} = P\mathbf{x^k} + H\mathbf{u_k} + K\phi + I_{N_p}\gamma \tag{4.46}
$$

Forcing $\mathbf{y_k} \leq \overline{\mathbf{y}} + \mathbf{s}$ such that the corresponding inequality is then found by solving for $H\mathbf{u_k}$, further introducing b_y to gather all terms.

$$
H\mathbf{u_k} \leq \overline{\mathbf{y}} - P\mathbf{x^k} - K\phi - I_{N_p}\gamma + \mathbf{s} = b_y + \mathbf{s} \tag{4.47}
$$

The objective function (4.40) is reorganized on matrix form, which gives the final system.

$$
\min_{\begin{bmatrix} \mathbf{u_k} \\ \mathbf{s} \end{bmatrix}} \quad \frac{1}{2} \begin{bmatrix} \mathbf{u_k} \\ \mathbf{s} \end{bmatrix}^{\mathsf{T}} \begin{bmatrix} \tilde{H} & 0 \\ 0 & W \end{bmatrix} \begin{bmatrix} \mathbf{u_k} \\ \mathbf{s} \end{bmatrix} + \begin{bmatrix} f \\ \mathrm{w} \end{bmatrix}^{\mathsf{T}} \begin{bmatrix} \mathbf{u_k} \\ \mathbf{s} \end{bmatrix}
$$
$$
s.t. \quad \begin{bmatrix} A_h & 0 \\ H & -I \\ -I & 0 \\ I & 0 \\ 0 & -I \end{bmatrix} \begin{bmatrix} \mathbf{u_k} \\ \mathbf{s} \end{bmatrix} \leq \begin{bmatrix} b_h \\ b_y \\ lb \\ ub \\ 0 \end{bmatrix} \tag{4.48}
$$

where the inequality represent the two inequalities in (4.45) and (4.47), the lower and upper bounds on the inputs and a lower bound on the slack to force it to be strictly positive.

4.4   Parameter Estimation for ROP Model

The ROP models contain empirical coefficients that must be estimated based on field data in to have an accurate model for control. The coefficients have upper and lower bounds that must be considered. The strategy is to solve a bound-constrained minimization problem on the form below

$$\min_{x \in \mathbb{R}^n} f(x)$$
$$s.t \quad lb \le x \le ub$$
(4.49)

where $f : \mathbb{R}^n \mapsto \mathbb{R}^1$ is the objective function, $lb \in \mathbb{R}^n$ are the lower bounds and $ub \in \mathbb{R}^n$ are the upper bounds on the decision variables x.

The strategy will involve using a least squares trust region approach to solve the optimization problem. The main idea is to compensate the bounds by scaling the variables and have been adopted from Branch et al. (1999). The method is outlined below, based on the implementation that is available through the SciPy library (Virtanen et al., 2020). Defining a vector $v = [v_1, \ldots, v_i]$, $i = 1, \ldots, n$, where each component represent the distances to the bounds, only if it is finite

$$v_i = \begin{cases} ub_i - x_i, & \text{if } g_i < 0, ub_i < \infty \\ x_i - lb_i, & \text{if } g_i > 0, lb_i > -\infty \\ 1, & \text{otherwise} \end{cases}$$
(4.50)

where g is the gradient of a cost function. Defining the a scaling matrix $D = diag\{v^{0.5}\}$. First-order optimality conditions can be stated as

$$D^2 g(x) = 0$$
(4.51)

Next, let the Newton step satisfy a new optimization problem, given below

$$(D^2 H + diag\{g\} J_v)p = -D^2 g$$
(4.52)

where H is the Hessian matrix, J_v is the Jacobian matrix of v (-1, 1 or 0), such that all elements of matrix $C = diag\{g\} J_v$ are non-negative. Based on a change of variables $x = D\hat{x}$, the new expression is given by

$$\hat{B}\hat{p} = -\hat{g}$$
(4.53)

where $\hat{B} = \hat{J}^T \hat{J}, \hat{J} = JD$ and $\hat{g} = Dg$. A trust-region problem based on the Newton step in the new variables is formulated below.

$$\min\{\tfrac{1}{2}\hat{p}^T \hat{B}p + \hat{g}^T \hat{p} \ : \ ||\hat{p}|| \le \Delta\}$$
(4.54)

In the original space, given by $B = H + D^{-1}CD^{-1}$, the equivalent trust-region problem is given by

$$\min\{\tfrac{1}{2}p^T Bp + g^T p \ : \ ||D^{-1}p|| \le \Delta\}$$
(4.55)

The residual function ρ to be used for fitting the drilling coefficients is chosen as the deviation between the estimated ROP and the field data, given by

$$\rho = \widehat{ROP} - y \tag{4.56}$$

The time horizon of drilling a section can span over multiple days. Therefore it is favorable to fit the model parameters only based on the most recent data points. This will ensure that the relevant drilling dynamics are properly captured in the coefficients to be estimated. Multiple effects such as the bit wear and differences in the lithology can affect the ROP coefficient over short time differences. Sui et al. (2013) presented using a sliding data window in order to gather only the most recent drilling data. The sliding data window of size L is defined as the set Φ, given by

$$\Phi(t) = \{ROP(t - L), \dots, ROP(t - 1)\} \tag{4.57}$$

Figure 4.3 visualizes a sliding data window Φ for the values of ROP to be used in the parameter estimation algorithm.
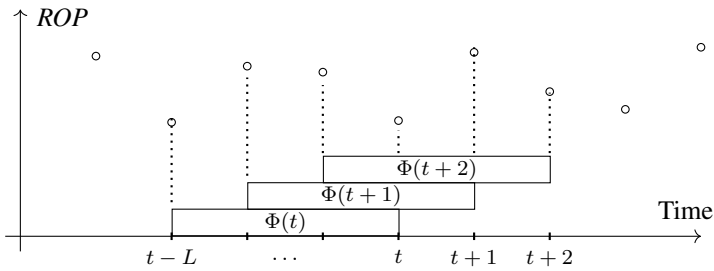

Figure 4.3: Sliding Data Window Φ

5

System Implementation

This chapter presents the details behind the implementation of the system. The first section covers the processing of the raw MRU and drilling data. The next section presents the neural network setup, training, tuning, and exporting. The neural network implementation is outlined first, and then the model predictive controller setup using predicted heave motion is presented. The last section presents the complete system. All names that are written in the style `foo_foo` refer to functions, classes, and other objects in the source code. Details for accessing the source code for the work carried out in this thesis are available in Appendix B.

The system has been programmed using the Python language. There are multiple powerful and well documented open source scientific and computing Python libraries, available for download through The Python Package Index (PyPI). The packages that are used in this project are summarised below.

The NUMPY library is utilized to have efficient multidimensional array computing for the project. The SCIPY package provides linear algebra and non-linear optimization utilities. The PANDAS package was used for handling the input drilling data and MRU data. The CVXOPT library has an efficient quadratic programming (QP) solver used for solving the QP formulation of the MPC. The TENSORFLOW machine learning platform is used together with the KERAS library to construct the heave prediction model for the project. Setting up and running drilling simulations are done through the OPENLAB client.

5.1 Data Handling

The MRU data for this project is from a semi-submersible rig operating in the North Sea and have been provided by Odfjell Drilling. The drilling data originates from a well in the North Sea and have been provided by AkerBP.
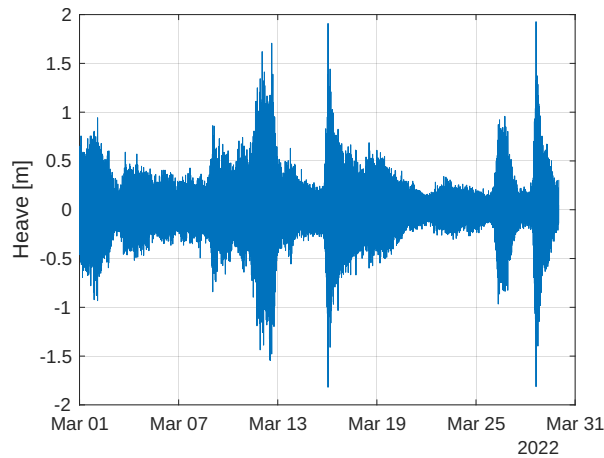
Figure 5.1: Raw MRU Input Data

5.1.1 MRU Data

The MRU data were given as an export from a database that aggregates and stores time-series from different sensors on the rig. The format of the export was a comma-separated value file (*.csv*) with measurements for a period of a month. The file had two columns, the average of three MRU measurements from the drillfloor, together with a timestamp. Figure 5.1 shows the MRU input data. The raw data did not have a constant sampling rate, due to compression of the measurements for minimizing storage footprint. The first step is to resample it. The sampling rate was selected at 2 Hz and linear interpolation was used for computing the new values. A function `load_heave` is implemented to handle the raw MRU input data and resamples it to 2Hz. Figure 5.2 shows a comparison between the raw input and the resampled MRU data for a 250 seconds data window.
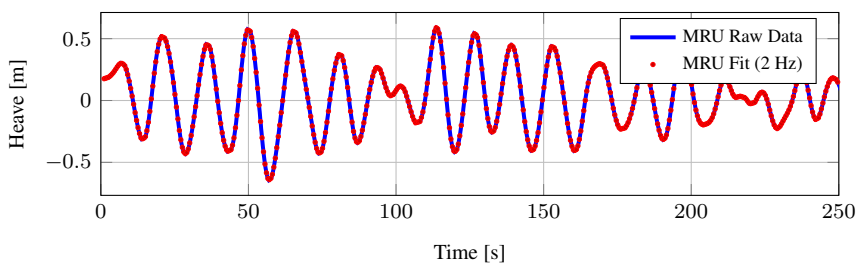

Figure 5.2: Resampled MRU data

The input data are split into three separate sets. That is, a training, validation, and test set. A division of 70%-20%-10% between the training, validation and test set was selected. The purpose of the mentioned sets are briefly outlined below.

- **Training Set (70%)**: In the training process, the weights and biases in the neu-

ral network are fitted based on the training set. Each sample of the training set is again divided into input and output (label) arrays as shown in Figure 5.3, with inputs in blue and labels in green color. In the figure, a sample in the training set consists of 210 values corresponding to 105 seconds, where two thirds of the sample are selected as inputs, and one third are chosen as labels. The setup indicates that 70 seconds of MRU measurements are aggregated and used to predict the heave response for the next 35 seconds.

- **Validation Set (20%)**: When a model is being fitted during the training procedure, predictions based on the validation set are evaluated in order to quantify the performance of the model. The measure of the performance is used to avoid over-/underfitting of the data. The training process can be terminated early to avoid overfitting of the data if the prediction error of the validation set increase with extended training.

- **Test Set (10%)**: The test set is used for providing a final metric of the performance of the fitted model after the training has finished.
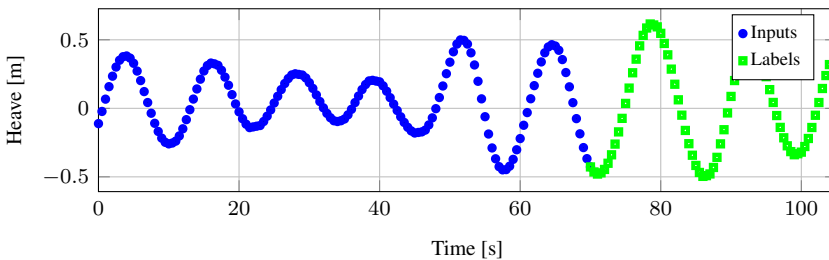

Figure 5.3: Inputs and Labels in the Training Data Set

The function `split_dataset(s,testsize,trainsize)` handles the partitioning of the data frame `s` into the training, validation and test sets according to the specified parameters for the fraction division (0-1) of the test set `testsize` and the training set `trainsize`.

The sets that have been partitioned are stored in a `Window` object. To construct the window object, the parameters `input_width`, `label_width` and `shift` are required. The input width specifies the number of values that are aggregated before making a prediction. The label width determines how many predictions are to be made. The shift parameter specifies the relative placement of the labels versus the inputs, or how far into the future the predictions are to be made. In Figure 5.3 the shift was set to the number of labels, such that there are no gaps between the inputs and labels. The member function `split_window()` does the input and label partitioning. A last parameter `label_columns` facilitates having multiple features if desired, but the implementation in this thesis covers one feature, which are the MRU measurements `MRU_VALUE`.

The `make_dataset` member function handles the sequential data, through the Keras `timeseries_dataset_from_array` utility method.

5.1.2   Drilling Data Handling

The drilling data were given as an export from a data acquisition system. The format was tabular in the form of an Excel file (*.xlsx*) for a BHA run from a real well in the North Sea. The RPM and WOB measurements are from downhole tools, delivered through wired pipe telemetry. The flow rate and ROP are measured topside and is therefore aggregated from the drilling control system. The first step was gathering the relevant data by removing the logs for operations such as tripping in/out of the hole. The next step carried out was to remove all erroneous measurements such as negative WOB values from the time series.

Figure 6.6 shows the raw input drilling data for a duration of around 2 hours and 30 minutes. The two intervals that are relevant to sort out, are straight before 04:20 and before 05:40. Here it can be seen that the ROP, RPM, WOB and flow rate goes to zero for a long period of time. Also, the block position is at the lowest point during the operation before it is increased up to 40 meters again. This indicates that a stand has been drilled, which means that there is made a new connection to have more length of pipe available to drill. The mentioned intervals are removed from the time series as only the drilling part is of further interest.

5.2   Neural Network Architecture and Training

Construction and training of the network is done in the `create_model` procedure. The Sequential model is used, since there is time-series data involved. After the input layer, there is a LSTM layer consisting of 50 units. A dense layer is added, consisting of 70 neurons. The last step involves reshaping the output to $y \in \mathbb{R}^{70 \times 1}$. Figure 5.5 shows the architecture that have been implemented and will be used for the simulation scenarios later. The number of layers and nodes to select in the model is not an exact science. The network only has a single feature, and a therefore a single layer of LSTM was selected. The constructed neural network is going to take in input values $x \in \mathbb{R}^{140 \times 1}$, and produce the predicted output $y \in \mathbb{R}^{70 \times 1}$ values into the future. The reasoning behind selecting 70 nodes in the dense layer is to match the number of outputs. The `compile_and_fit` procedure specifies a callback function to terminate the training procedure by monitoring the validation loss. The training procedure terminates when the mean absolute error (MAE) of the predictions in the validation set increase from an epoch to another.

$$MAE = \frac{\sum_{i=1}^{n} |y_i - x_i|}{n} = \frac{\sum_{i=1}^{n} |e_i|}{n} \qquad (5.1)$$

where n is the number of samples and e_i is the error between the predicted value y_i and the true value x_i. The optimizer used is the ADAM method, that were presented in algorithm 1.

5.3   Controller Architecture

The `setup` function handles all the static configuration variables required for the controller and drilling simulations. The static configuration includes the Hareland model bit
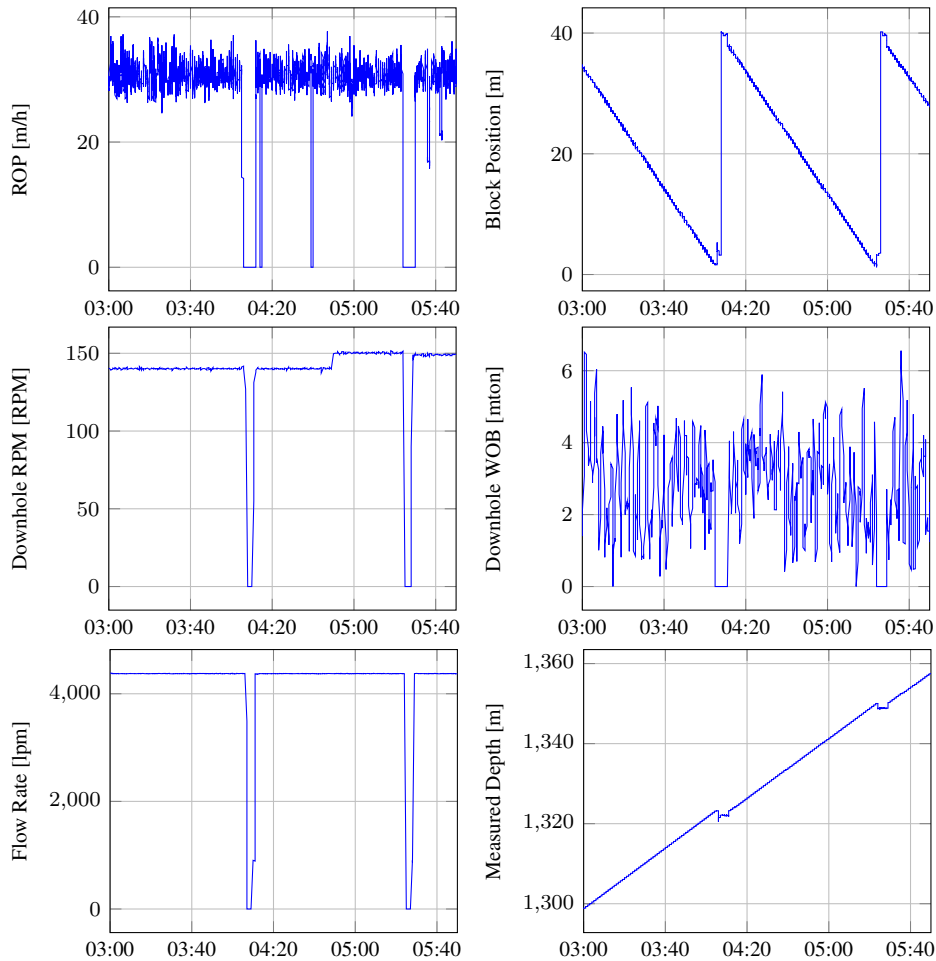
Figure 5.4: Drilling Data

parameters, the length of the sliding data window L, and initial guesses for the drilling coefficients a,b,c to be used in the parameter fitting. Number of inputs, outputs and states are defined as well as prediction horizon. Weighting matrices Q, R, S are defined. Upper and lower limits for WOB, RPM are defined in addition to maximum rate of change to the MVs. For advisory mode, the drilling response is read through an input file, coming in as an export from a M/LWD data acquisition system. For closed loop simulations, the Open-Lab simulation is configured to constantly be stored and used as input. The two different scenarios will be further outlined later.

A function `hareland` contains the full definition of the Hareland model, as given in (3.14). Formulas within the field of drilling engineering often employ imperial units instead of metric which applies to the Hareland model as well. The input to the function is the RPM [rpm], WOB W [klbs] and the drilling coefficients a,b,c. The function returns the
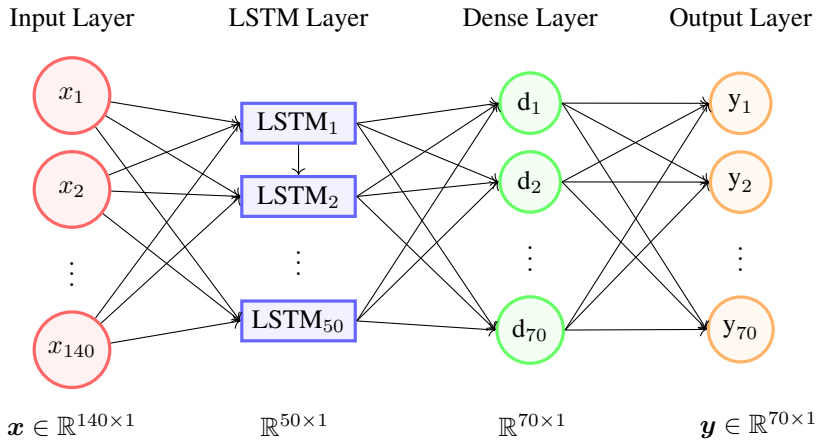
Input Layer   LSTM Layer   Dense Layer   Output Layer

Figure 5.5: Model Architecture

resulting ROP [ft/hr]. `get_linearized_model` computes the linearized state space matrices A_t, B_t, C_t and D_t for the Hareland model. The parameters are the sampling period dt, the estimated ROP, drilling coefficients a, b, c and the input operating points in which the linearization happens around (W, N_r). The terms J11_ev and J12_ev are evaluated based on the analytical expressions (4.24), (4.25) and are split into separate functions `J11` and `J12`.

The function `run_mpc` wraps all logic for the MPC at each timestep. The drilling data and heave data is partitioned with `partition_drilling_data` and `partition_responsedata`. This is done based on the sliding window approach. Then, there is a startup procedure used for drilling as shown in Figure 5.6, which must be executed in order to run the MPC in OpenLab. The actions are not required for running it in the advisory mode. Circulation is established by increasing the flowrate up to the desired value. Then, the drillstring rotation is started by gradually increasing the RPM. First, when bottom have been tagged, the MPC algorithm will start to act on the system.

The QP problem is formulated and solved for optimal inputs through `get_opt_drillparams`. The optimal control input is either applied to the system if running closed loop simulations, or just stored in memory and presented to the user when running in advisory mode.

The function `run_openlab` opens the connection to OpenLab, that is a cloud hosted application requiring credentials to access. The step duration, configuration name and simulation name must be applied in addition to credentials. The well configuration can be accessed through the web client. A parameter simulation_time specifies how many timesteps the simulation is going to run for.
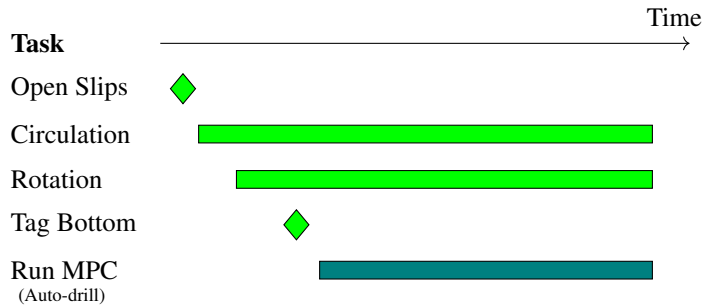
Figure 5.6: Startup of Drilling Procedure

5.4 Final System

An overview of the components in the system that have been implemented are shown in Figure 5.7. The MRU measurements are passed to a LSTM artificial neural network that make predictions about the future heave response of the rig. The future predicted heave response is passed to an MPC. The MPC solves an optimization problem based on a linearized Hareland ROP model. The MPC is constrained with upper and lower limits in the input variables, and on the maximum ROP. The solution is an optimal set of WOB (w_{opt}) and RPM (r_{opt}).

The dashed rectangles that are colored red and blue represent the two different operating modes of the system. Red box shows the advisory mode of the system, where M/LWD downhole sensor data from a real well is passed to the MPC. The optimal WOB and RPM will for this mode be based on actual drilling data from a well in the North Sea. Blue box shows the closed loop simulation mode, where the OpenLab drilling digital twin is used. The optimal WOB and RPM is applied to the drilling digital twin and the corresponding response of the system is passed back to the MPC.
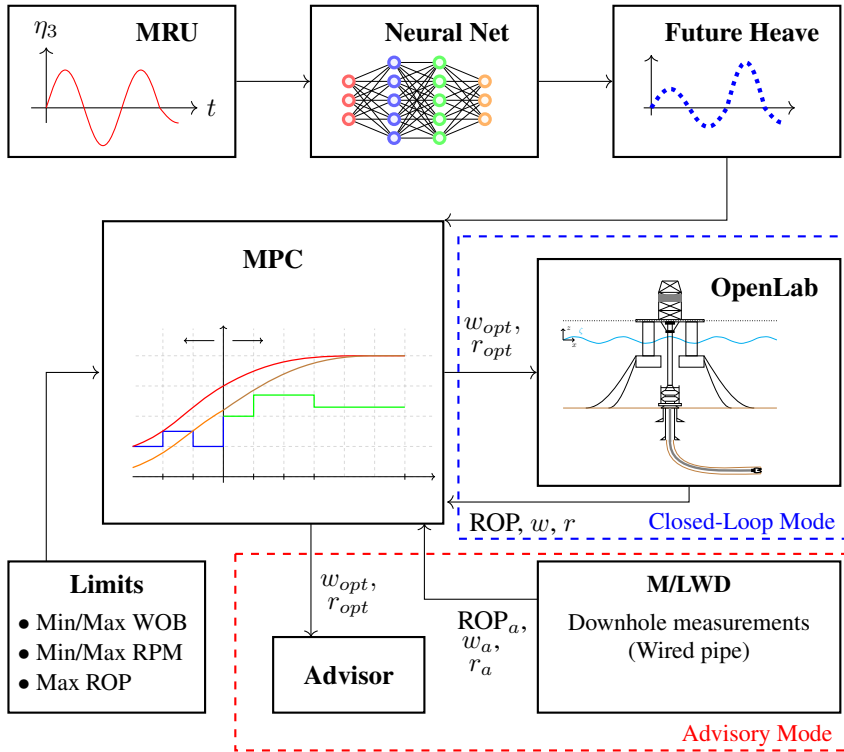
Figure 5.7: System Overview

6

Results

This chapter presents the simulation results from the implemented system. First, the performance of the heave prediction system is assessed. Then, the performance of the MPC controller utilizing the heave predictions to optimize the drilling parameters are presented. The MPC controller was run in two different operating modes. Advisory mode is based on data from a real well. Closed loop simulations are run against the OpenLab environment.

6.1 Heave Prediction

Two different simulation cases are evaluated for the heave prediction module. The procedure consists of using a training dataset, as shown in Figure 6.1 for the first case, and training the neural network based on the selected data. Only 500 seconds of MRU data were considered for the first simulation. Batch size was set to 10 and the number of epochs was set to 20. Since the validation set is defined as 20% of the total input, the validation loss was not monitored during the training. Three random inputs that are outside the training set are passed to the model that have been trained, and the accuracy of the prediction is evaluated. Figure 6.2 shows graphs of the result from the three different input sequences.
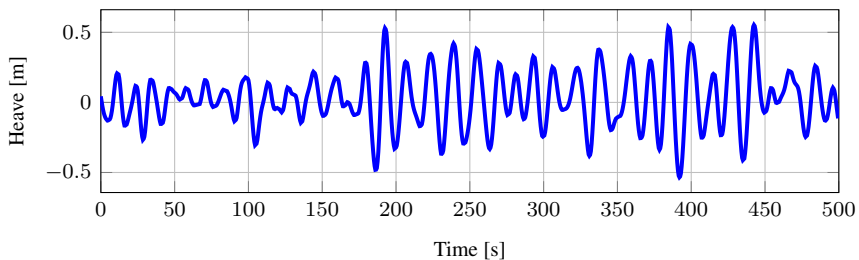

Figure 6.1: MRU Training Data, Case 1

The blue line shows the actual response, where the first 70 seconds (140 measurements)
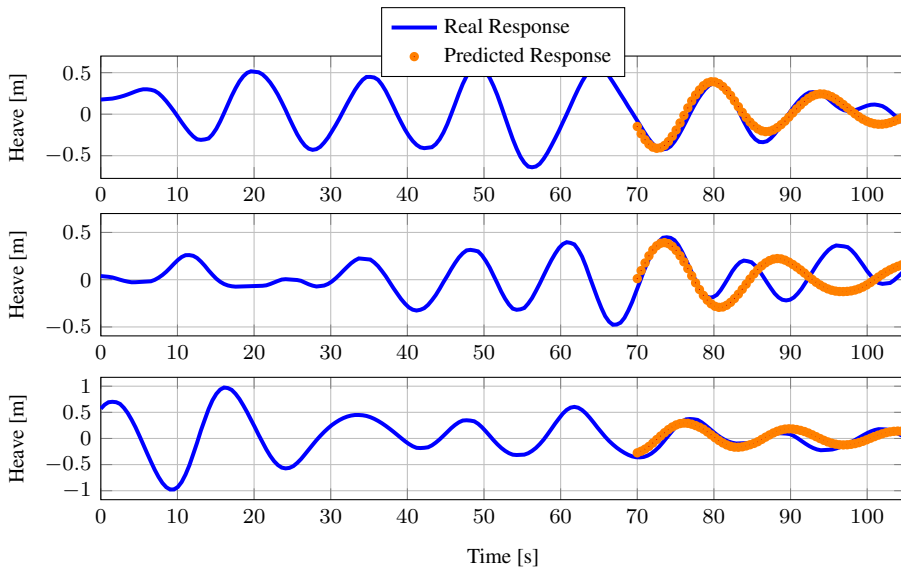
Figure 6.2: LSTM Heave Predictions, Case 1

Prediction Horizon	5 s	10 s	15 s	20 s	25 s	30 s	35 s
RSS_I [m]	0.012	0.044	0.111	0.209	0.247	0.338	0.611
RSS_{II} [m]	0.019	0.031	0.744	1.717	2.376	3.992	4.087
RSS_{III} [m]	0.121	0.179	0.217	0.243	0.676	0.747	0.827

Table 6.1: RMSE of Heave Predictions, Case 1

are passed as input to the neural network, and the next 35 seconds (70 measurements) are used for evaluating the accuracy of the prediction. An orange dotted line indicates the predictions made based on the input. The metric used for evaluating the prediction accuracy is the residual sum of squares $RSS = \sum_{i=1}^{n}(\hat{y}_i - y_i)^2$, where an estimated value is denoted $\hat{y}_i$ and the actual is denoted y_i. Table 6.1 shows the development of the *RSS* for a prediction horizon varying from 5 seconds up to 35 seconds. In the first graph for simulation case 1, the *RSS* values are only slightly increased for increasing prediction horizons. By visual inspection, the predictions are reasonable for the first 30 seconds.

For the second case, there is a large increase in *RSS* starting for prediction horizons longer than 10 seconds. This is also identified when looking at the graph, where it also can be seen that the prediction even goes 90 degrees out of phase compared to the actual response after 20 seconds of predictions have been made, which means at 90 seconds.

In the last case, the *RSS* is high already for low prediction horizons with 0.121 for a 5 seconds prediction horizon. It is observed that the predictions start to slowly move out of phase from the actual response after 20 seconds into the future.

For all three cases, it can be seen that a prediction horizon of 10 seconds yields reasonably accurate predictions, based on a *RSS* of 0.044, 0.031 and 0.179 respectively.
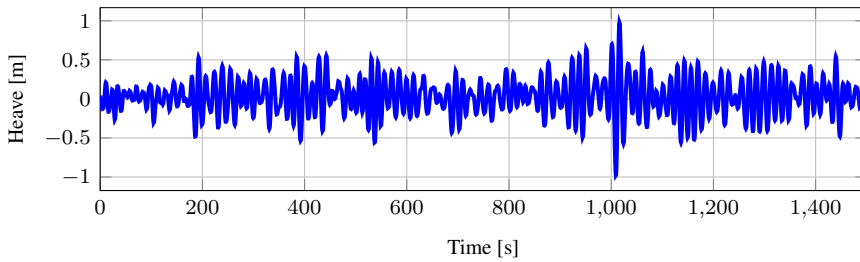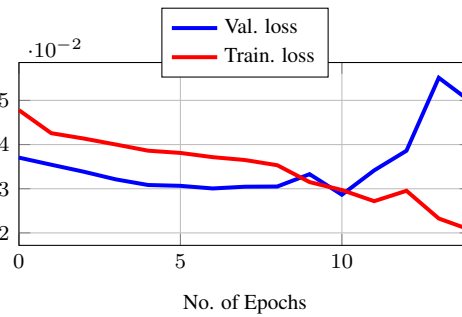
Figure 6.3: MRU Training Data, Case 2

Figure 6.4: Training and Validation Loss, Case 2

The heave training data for the second simulation case is increased to 1500 seconds as shown in Figure 6.3. The batch size and number of epochs are the same as for the last case. In this case, there will be validation samples available during the training procedure. The concept of early stopping was therefore employed for this simulation case. The validation loss and training loss are shown in Figure 6.4, and the development in the validation loss indicates that increasing the number of epochs even further can lead to overfitting of the data, so the learning process has terminated before the maximum amount of epochs have been reached. Table 6.2 shows an overview of the RSS development for the three different inputs over different prediction horizons. The first observation is that the increased amount of data used in training the model indicates to have improved the predictive abilities for longer horizons, with a RSS of respectively 2.494, 0.265 and 0.368 for a 35 seconds prediction horizon. With 10 seconds of prediction horizon, the RSS were 0.096, 0.011 and 0.127 for the three simulations.

Prediction Horizon	5 s	10 s	15 s	20 s	25 s	30 s	35 s
RSS_I [m]	0.016	0.096	0.331	1.071	1.983	2.331	2.494
RSS_{II} [m]	0.004	0.011	0.025	0.029	0.041	0.139	0.265
RSS_{III} [m]	0.031	0.127	0.202	0.244	0.331	0.359	0.368

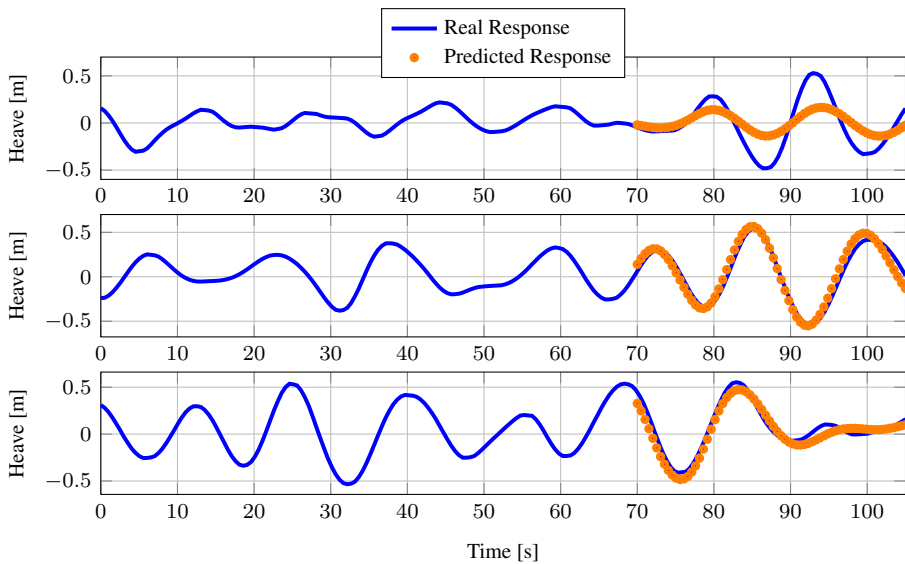**Table 6.2:** RMSE of Heave Predictions, Case 2

Figure 6.5: LSTM Heave Predictions, Case 2

6.2 Drilling Performance

The drilling performance is assessed through two simulation modes, the advisory mode and closed loop mode. The advisory mode is based on data from a real well drilled in the North Sea from a floating drilling installation. Closed-loop simulation mode use the OpenLab simulation environment, such that the optimized drilling parameters from the MPC can be applied in a drilling environment to generate a response. The input drilling data for the advisory mode is shown in Figure 6.6 for the time horizon of drilling a stand, and will also be used to assess the accuracy of the parameter estimation method. The ROP setpoint in the auto-driller was set to 30 m/h during the drilling operation. The logs show that the measured ROP in the input data has amplitudes ranging from as low as 25 m/h up to as high as and even exceeding 35 m/h. The block position decreases as the stand is drilled, and there is a corresponding increase in the measured depth. The WOB is ranging from above 6 tons to below 1 tonne. The RPM is at 140 rpm, with occasional deviations of up to 1 rpm. The flow rate is stable around 4375 lpm.

   Both operating modes require to configure the parameters of the Hareland ROP model. The static properties that have been used in the model for both operating modes are shown in Table 6.3. The bit properties used are representative for a PDC bit and are going to be constant for a BHA run. The UCS was assumed to be constant during the drilling simulation, which is a reasonable assumption for short simulation drilling only a couple of meters. Also, because eventual changes in UCS will be compensated for in the model through the fitting of the lithology coefficient a.
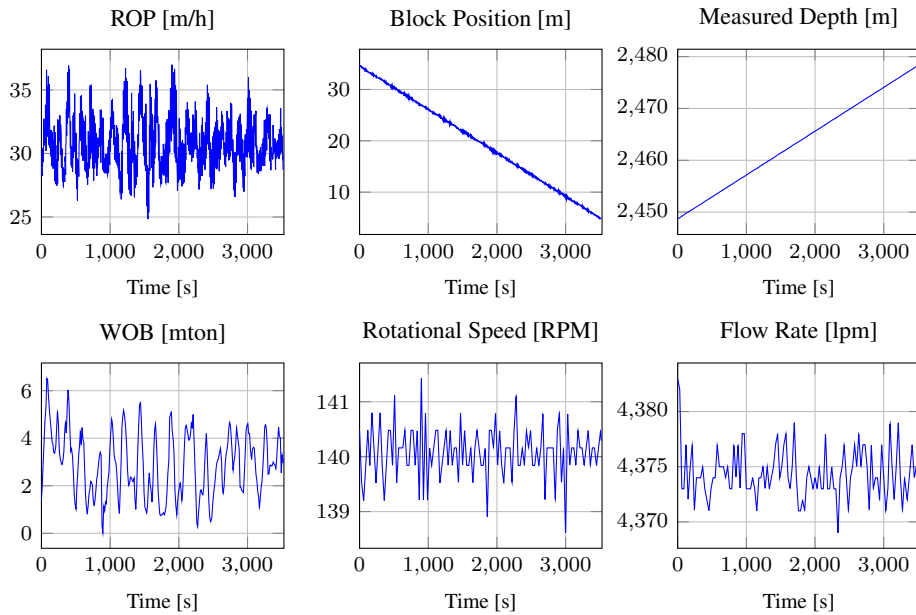
Figure 6.6: Input Drilling Data

Parameter	Value
Bit Diameter	12.25 [in]
Number of Cutters	50 [-]
Cutter Diameter	0.7 [in]
Backrake Angle	10 [deg]
Siderake Angle	30 [deg]
Bit Wear	1 [-]
UCS	80 [psi]

Table 6.3: Hareland Model Properties

6.2.1 Model Parameter Estimation

Having reasonable accuracy on the real-time estimations of the coefficients in the ROP model are important for achieving accurate predictions in the MPC. Coefficients a, b and c capture the lithology properties and the effectiveness of the WOB and RPM applied in the drilling operation. The results of the parameter estimation process for a sliding window size of $L = 20$ samples are shown in Figure 6.7. The first graph shows the estimation of the lithology coefficient a. The value of a is mostly trending around 2000. Some areas including before 500 seconds, the coefficient is increasing above 5000 and up to the predefined coefficient bound of 10000. The second graph shows the estimated RPM coefficient b, which is fluctuating between the coefficient bounds of 0.5 and 1. The third graph shows the WOB coefficient c, that mostly trends towards the lower limit at 0.5 with areas increasing up to the upper limit of 1.5.
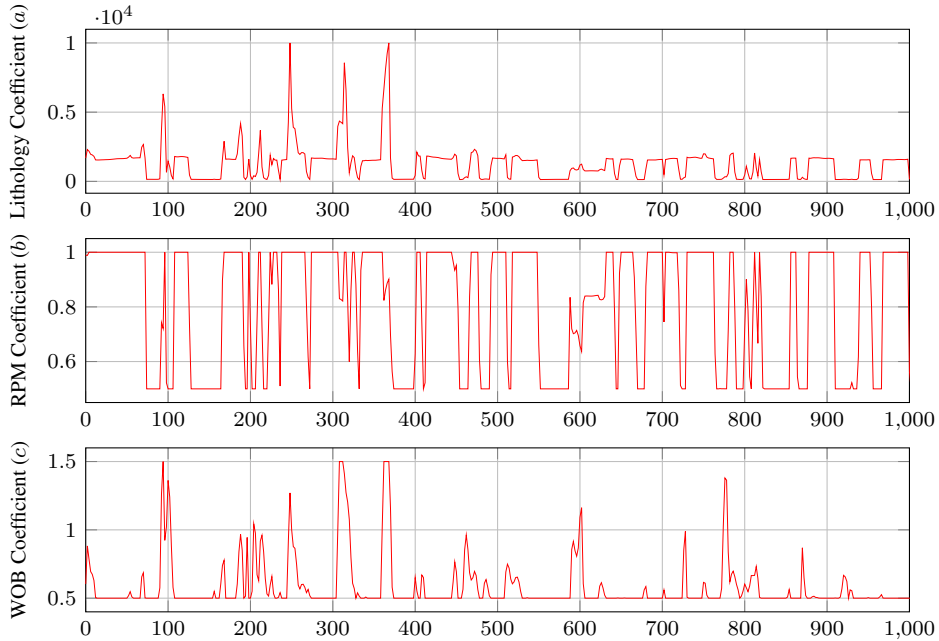
Figure 6.7: Parameter Estimation, $L = 20$

Figure 6.8 shows the estimated ROP in red against the real ROP in blue, indicating a good model fit to the actual data based on the estimated model parameters a, b and c. The estimation accuracy between the predicted value $\hat{y}_i$ and the actual value y_i is assessed with the $RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y})}$. The RMSE between the estimated ROP and the actual ROP was 1.25 m/h for a sliding window size $L = 20$.

The parameter estimation exercise was carried out for a range of different window lengths L, with the results shown in Table 6.4. The accuracy is lowered with increasing window size, which is due to the inclusion of the process dynamics from longer back in time when estimating the coefficients at the current timestep.

Figure 6.8: ROP Based on Estimated Parameters

Window Length L	20 s	30 s	40 s	50 s
$RMSE_{ROP}$ [m/h]	1.25	1.61	2.01	2.05

Table 6.4: RMSE ROP Estimations for different window sizes L

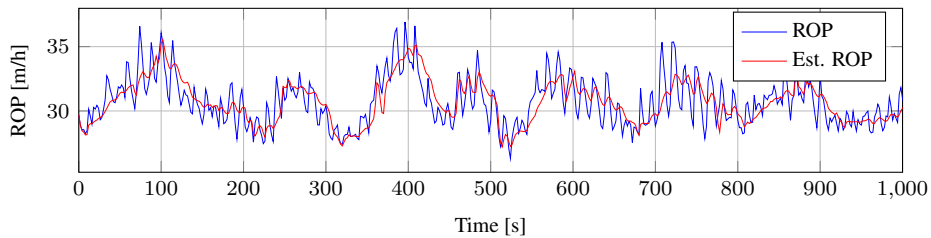## 6.2.2   Parameter Recommondations in Advisory Mode

In advisory mode, the controller is running in the background providing recommendations on the set of optimum WOB and RPM in open loop. The control action can be chosen to be, or not to be, applied to the system. The predefined ROP setpoint was set to 30 m/h. The setpoint corresponds to the ROP setpoint that was desired and used in the auto-driller during the operations. The sampling period of the MPC was set to 1 second. The output tracking penalizing weight was set to 1000. The linear and quadratic weights on the slack in the ROP was set to 1. Minimum and maximum rate of change in WOB was set to 2 tons. For RPM it was set to 3 rpm. The weights penalizing the input deviation was set to 1. The prediction horizon of the MPC was set to 12 samples, corresponding to 12 seconds. The heave correction factor C_η was set to 0.001. At each timestep in the simulation, the first recommended advice over the prediction horizon have been extracted.

The advisory mode simulations are carried out for two different heave time series inputs that are assumed to be the output from the LSTM heave prediction module. The results from the two simulations are shown in Figure 6.9 and Figure 6.10. The time horizon of the simulations has been set to 800 seconds. The first graph in both the figures shows the heave response that was provided to the MPC. The second plot shows the ROP setpoint (dotted green line), against the actual ROP (red line) that was achieved during the offshore drilling operation. The third and fourth graph shows the actual values of WOB and RPM in red, against the controller recommendations in blue. In the legends of the plots, Rec. is shorthand for recommended and Act. is shorthand for actual.

In the first simulation, shown in Figure 6.9, the heave input to the algorithm had a amplitude mostly between 0.5 meters and 2 meters. The recommendation during the first 150 seconds is to lower the WOB below the actual applied value in order to drive the ROP towards the setpoint. Then, for the next 50 seconds the recommendation is to raise and lower the WOB to meet the objective of the desired ROP and the heave motion suppression. At around 400 and 600 seconds it recommends running with lower values for the WOB.

In the second simulation, shown in Figure 6.10, the heave input to the algorithm had a amplitude mostly between 0.2 meters and 0.5 meters. The recommendation are like the first case, but with larger frequency than in the first case. Also, the WOB recommendation are oscillating at a lower amplitude than in the first case.

For both simulation cases, the effect of the heave compensation on the drilling parameters are best shown in the last 200 seconds. Here, the field ROP data is mostly stabilized around the ROP setpoint and the parameter recommendation mostly reflects the effects due to the predicted heave motions. When the process value deviates from the set point, the WOB correction can be seen as the sum of correcting the deviation and suppressing the wave induced motion. A significant deviation from the setpoint leads to minor parameter oscillations for the periods where this is the case, such as at 100 seconds and 400 seconds.
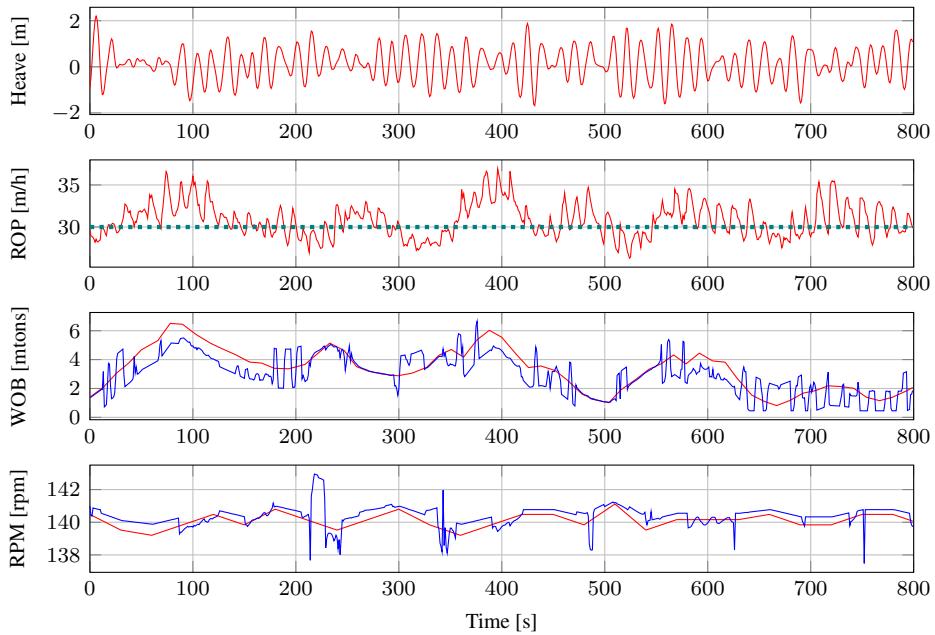
Figure 6.9: Parameter Recommondations (Blue) and Actual Parameters (Red), Case 1
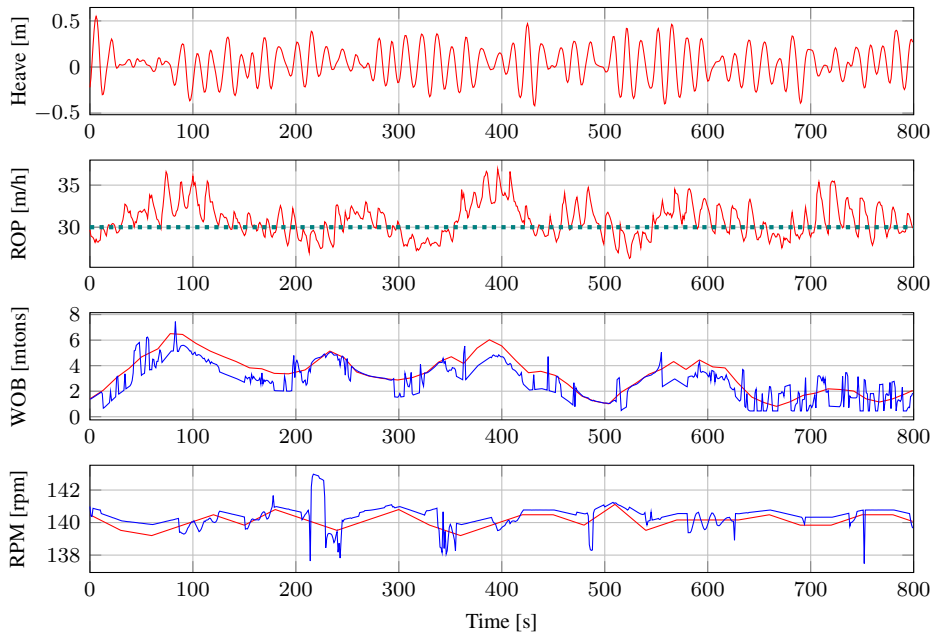

Figure 6.10: Parameter Recommondations (Blue) and Actual Parameters (Red), Case 2

6.2.3 Closed-Loop Controller Performance on a Simulated Well

To evaluate the closed-loop performance of the controller, two simulation scenarios have been run against the OpenLab drilling simulator. The first simulation of the MPC is for a constant ROP setpoint to look at the heave compensation capabilities of the system when it has reached steady state. A second simulation scenario involving multiple ROP setpoints have been run to assess the tracking capabilities of the MPC. The tuning parameters of the MPC have been set to the same as for the advisory mode.

The well, BHA and mud specific configuration data that were used for setting up the drilling simulations will be briefly outlined. In OpenLab this configuration can be done through the Python interface, or by accessing the web client. The simulations used the inclined well template that are available through the web client.

Table 6.5 shows the dimensions of the hole section to be drilled and the dimensions of the casing that are going to be run. The $12\frac{1}{4}$" section has been configured to be drilled. The hole size to be drilled will be most relevant for the ROP dynamics, but the casing dimensions must also be configured, even though they are most relevant for the hydraulic calculations.

Hole Size [in]	Type	OD [in]	ID [in]	Hanger Depth [m]	Shoe Depth [m]
36	Casing	32	$30\frac{3}{4}$	200	260
26	Casing	20	$18\frac{23}{32}$	200	830
16	Casing	$13\frac{3}{8}$	$12\frac{13}{32}$	200	1550
$12\frac{1}{4}$	Casing	$10\frac{3}{4}$	$9\frac{15}{16}$	200	2183

Table 6.5: Hole Section Configuration

Table 6.6 shows the configuration data for the BHA. A $12\frac{1}{4}$" PDC bit is used for drilling the section. A M/LWD package is included in the BHA, which reflects the sensors that gather the directional and drilling dynamics data. 5" drillpipe is used in the BHA.

In the first simulation case, the ROP setpoint is set to 30 m/h for the entire simulation duration. The objective for the MPC is to reach a ROP of 30 m/h, and suppress heave motion disturbances when it has reached steady state. The heave compensation factor was set to $C_\eta = 0.001$. Figure 6.11 shows the simulation results for the first closed-loop simulation. For this compensation factor it can be seen that the amplitude of the recommendations of WOB for most of the large heave prediction inputs, are equal. This is due to that the computed WOB recommendation is restricted by the bounds in the MPC formulation, such that pairs of recommended values appear equal. The ROP reaches steady state between 300 seconds and 400 seconds. Due to the inertia of the drawworks and the controller associated with it, the actual WOB deviates from the recommended value from the MPC.

Figure 6.12 shows the result from a simulation with a lower heave compensation factor of $C_\eta = 10^{-6}$. Here it can be seen that there are almost no action in the recommended WOB compared to the first simulation case.

Figure 6.13 displays only the data from 400 seconds up to 600 seconds in the first
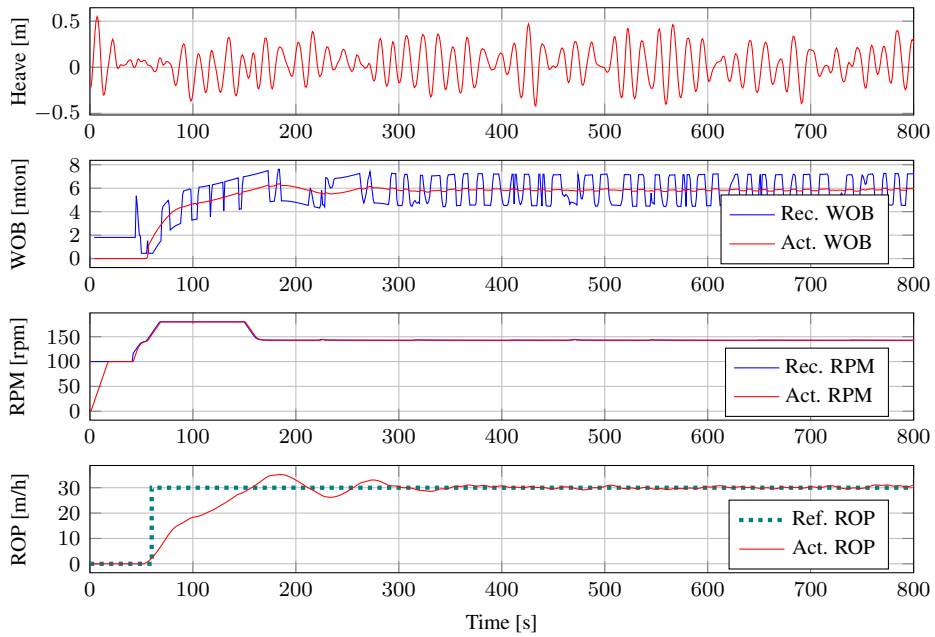
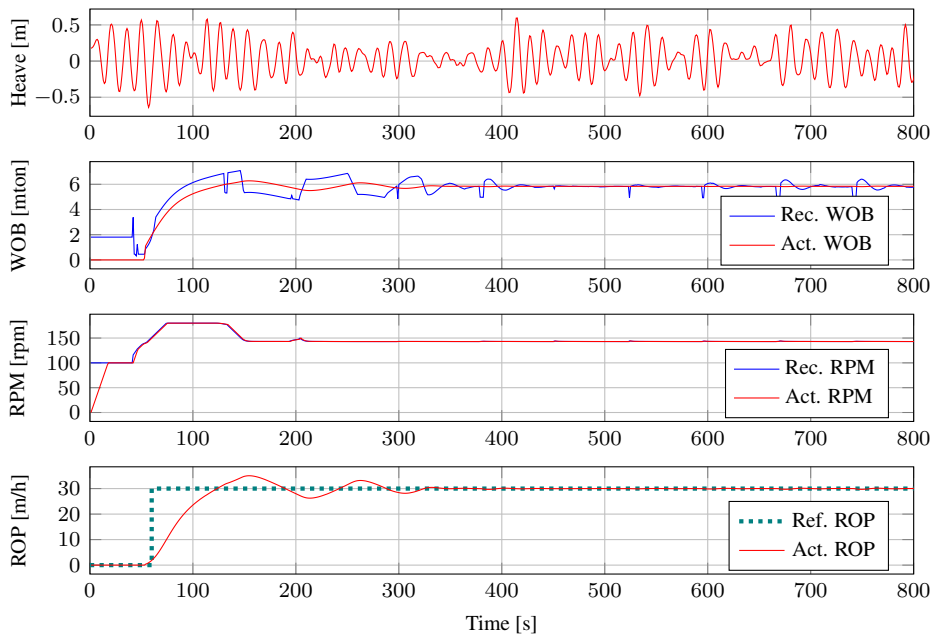**Figure 6.11:** Controller Performance, Fixed Setpoint ($C_\eta = 0.001$)

Figure 6.12: Controller Performance, Fixed Setpoint ($C_\eta = 10^{-6}$)

#	Type	Length [m]	OD [in]	ID [in]	Weight [kg/m]
1	Drillpipe	2000	5	$4\frac{9}{32}$	33
2	HW Drillpipe	54	5	3	76
3	Jar	9.4	$6\frac{1}{2}$	$2\frac{1}{2}$	139.7
4	HW Drillpipe	44.9	5	3	76
5	Drillcollar	28.8	$6\frac{1}{2}$	$2\frac{13}{16}$	145.7
6	MWD	7.9	$6\frac{7}{8}$	3	139.6
7	LWD	6.4	$6\frac{7}{8}$	3	134.6
8	Drillcollar	2	$5\frac{1}{2}$	3	84.5
9	Float sub	1	$6\frac{3}{4}$	3	146
10	RSS	7.7	$6\frac{3}{4}$	3	148
11	Bit	0.3	$12\frac{1}{4}$	3	150

Table 6.6: BHA Configuration

simulation case. From maximum values of the heave response from 520 seconds to 600 seconds, there is a corresponding maximum recommended WOB, which was the purpose of the controller in the first place. There is not possible to simulate rig heave in OpenLab, so the only reasonable metric to observe is the development of the controller WOB. But to estimate an ROP value, in order to compare the effects with and without the controller, an ROP disturbance has been assumed to be proportional to the heave motion disturbance. The blue graph shows the ROP development based on the previous assumption and is marked with ROP w/o (without) in the legend. It serves as an indication of the ROP development in the case where the controller runs without heave compensation. The red graph is the actual development of the ROP with the controller active. Under the assumptions that have been made, the actual ROP tracks the reference closer than the ROP without heave compensation.

For the last simulation case, the initial ROP setpoint is set to 40 m/h at 50 seconds. At 350 seconds the setpoint is set to 70 m/h, before it is lowered to 20 m/h at 550 seconds. During the whole simulation, the maximum ROP is set to 60 m/h. The heave correction factor is set to 0 during the whole simulation period. The reason for running the second simulation case is for validating the tracking properties of the MPC, as well as ensuring that the ROP does not surpass the maximum bound. It also serves as a validation simulation for employing the Hareland model in a MPC. Figure 6.14 shows the results from the simulation case. The ROP manages to reach the specified setpoints, and the output does not violate the specified maximum bound on the ROP.
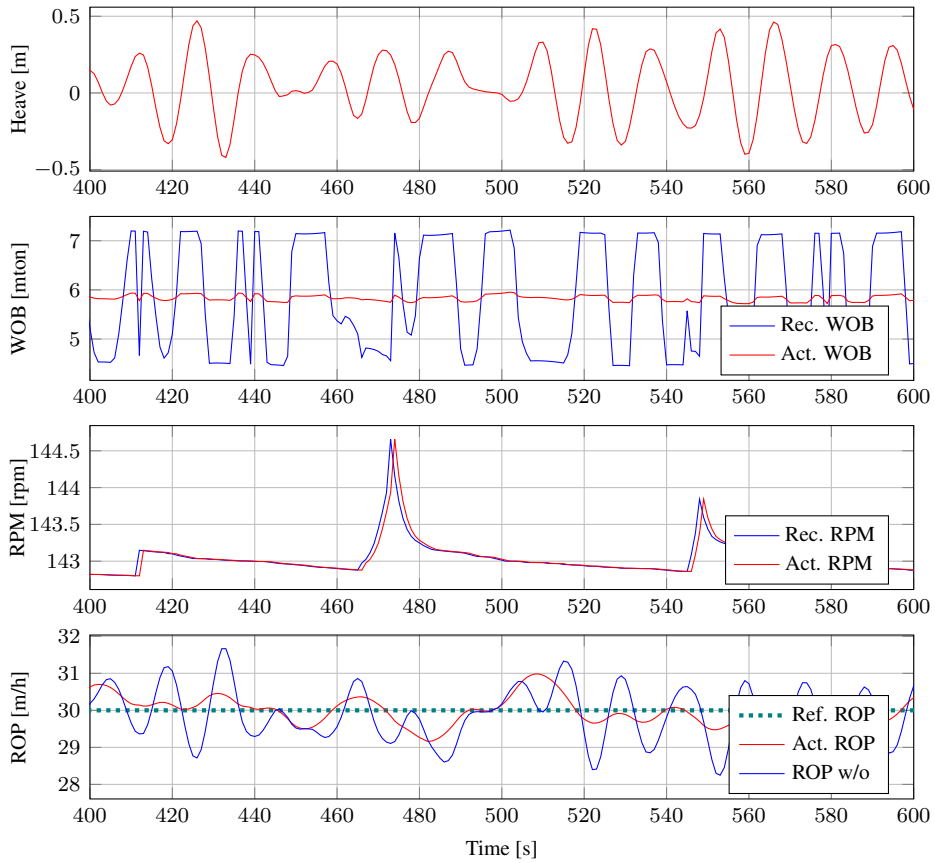
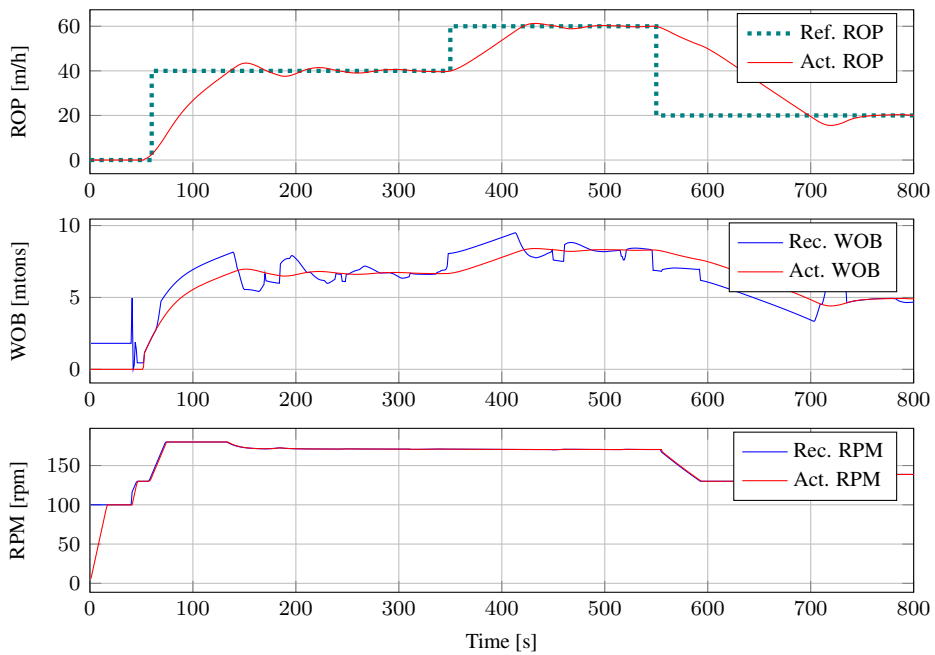**Figure 6.13:** Closed Loop Controller Performance ($C_\eta = 0.001$)

Figure 6.14: Closed Loop Controller ROP Tracking

7

Discussion

During the literature review process there was identified that there had not been done any previous work on combining heave response predictions and ROP optimization in real time. This thesis has done an effort in merging the fields of marine, drilling and control engineering in order to realize a controller that address this gap. Simulation results shows that the system can adjust drilling parameters based on the predicted heave and that it has potential to stabilize the ROP by mitigating effects due to heave. There are however some weaknesses associated with the implementation, mostly around the verification strategy. This chapter will discuss the strengths and weaknesses that have been identified.

7.1 Heave Predictions using Neural Networks

Employing a LSTM neural network to predict heave motions are a feasible strategy, based on the simulation results. The accuracy of the prediction model is high for short prediction horizons. When the horizon is longer than 2 response periods, the accuracy drops significantly. A neural network has the learning capabilities, pattern recognition properties and possibility to deal with nonlinear relations which made it a robust strategy. There have been done previous work on using LSTM for heave prediction as mentioned in the literature review section. The model proposed by Guo et al. (2021) used wave measurements and heave response as input to a LSTM model. The LSTM model implemented in this thesis only used the heave response as input, meaning that the corresponding neural network had a single feature. A benefit of implementing it with a single feature is that the neural network is a lot less complex to construct. With only a single feature to fit parameters for, the number of layers and nodes required are less than for multiple features. The model will therefore be simpler to train. For use in an operation, a single feature means that it only requires measurements from a single sensor to function. The model in this project was trained on, and compared against real data from a semi-submersible in the North Sea, which further proves the capabilities of the method.

Another point that must be taken into consideration is how much data should be used to train the neural network and how often to update the weights in the model. It was

observed that when increasing the amount of data to train the model on, the predictions became more accurate for long horizons. The network architecture that was described and implemented, had a layer consisting of 50 LSTM nodes and a layer of 70 dense nodes. The implemented network configuration provided reasonable predictions, but the optimal number of layers and nodes to use in a response prediction neural network can differ from the selection used in this thesis.

However, there are some weaknesses involved with the selected method for heave prediction. Firstly, it does require offline training, meaning that it cannot update the weights and biases of the network as new measurements come in. Also, since the method is entirely data-driven, it does not contain details about the fundamental physics of the system to guide the system. This translates into that the data used in the training process must be properly handled beforehand, such that irrelevant dynamics are not introduced to the neural network. A weakness of using only the MRU measurements, is that the neural network does not know anything about the incident wave properties when making the prediction. The mentioned weaknesses are the reason that the method performs worse for longer prediction horizons. The implementation by Guo et al. (2021) included wave measurements in the neural network, and this can be one of the reasons they reported high accuracies on predictions with horizons of over 40 seconds.

For operations that require prediction horizons longer than 10 seconds, such as for downhole pressure control when making a connection, the strategy should be enhanced by improving the mentioned weaknesses. But for the case of adjusting drilling parameters, a 10-15 seconds look ahead in time of the response will be more than enough, making a LSTM neural network with a single feature sufficient for the purpose.

7.2 Drilling Optimization using MPC

A least squared parameter estimation method was able to fit the model coefficients efficiently within the specified bounds. For a sliding data window size of $L = 20$ samples, the RMSE of the estimated ROP was 1.25 m/h. For increasing window sizes, the RMSE of the ROP increased accordingly. Selecting the window size was a tradeoff between the accuracy of the estimation and how long dependency of the ROP dynamics to include. The RPM coefficient goes in saturation between the bounds of 0.5 and 1, which is a consequence of restricting the allowable values for the coefficient.

Based on simulations in advisory mode, the simulation that was running with more substantial heave motions was more aggressive in terms of the WOB corrections. During periods of significant offsets from the setpoint, the recommendation shows signs of providing reasonable recommendations for how to correct for the offset.

The closed loop performance of the controller was tested by running simulations against the OpenLab simulator. Tracking performance of the ROP was assessed through a simulation scenario with different ROP setpoints. The controller was successfully able to steer the ROP to the desired value through applying an optimal set of WOB and RPM through the OpenLab simulator interface. There was identified that the drawworks seems to lag the recommendations, such that the applied WOB shows indications of not being able to follow the recommended WOB. The second simulation case was done for a constant ROP setpoint to study the effect of the heave compensation. Due to that the applied WOB by

the drawworks lagged the recommendations of the MPC, the heave correction factor was set to 0.001 to generate a sufficient ROP response.

Sui et al. (2013) and Kommedal (2021) both employed an MPC for use in a drilling operation. Compared to the mentioned contributions, the MPC presented in this thesis considers the predicted future wave induced motion of the rig when solving the optimization problem. Another difference compared to the previous efforts are that the Hareland ROP model have been used in this thesis, compared to the Bourgoyne and Young model that was used in the mentioned efforts. Various ROP model candidates have been outlined, each with different parameters that are taken into account in the model. The Hareland model was chosen due to the abilities to accurately represent a PDC bit that is penetrating the formation. Simulations show that the Hareland model is a promising model to employ in a MPC. In the effort by Sui et al. (2013), the optimization problem was formulated as a nonlinear MPC, which can be a more promising solution than linearizing the model. This thesis did like Kommedal (2021), and linearized the process model to use it in a MPC.

7.3   Verification of System

There are advantages and disadvantages of the selected verification strategies in this thesis, which were the advisory mode and closed-loop mode. A benefit of running the controller in advisory mode is that one gets to run real-world drilling data through the algorithms. The parameter recommendations are then based on data from a real well, and if the study proves the recommendations to be reasonable, then there is some evidence for deploying it offshore. But there are some weaknesses in assessing the full system potential through the advisory mode. It is not possible to apply the recommended WOB, and RPM to the system and study the closed-loop response without testing it on an actual offshore rig. A benefit of running the MPC in a simulation environment, such as the OpenLab drilling simulator used in the thesis, is that it is possible to study the closed-loop response. This provides insight into whether the control action is going to drive the ROP to the desired value. Another benefit of a simulation environment is the possibility to isolate the factors that are not of interest. An example from the simulations was to make sure that the ROP was around the setpoint during most of the simulation period. This was identified as an issue when running it in advisory mode, since it was difficult to directly identify the effect of the heave compensation due to periodic significant deviations from the setpoint. In a more controlled environment it was simpler to isolate the heave compensation effects in the recommended drilling parameters.

Some additional words to the simulations carried out in the OpenLab drilling simulator. It was impossible to recreate drilling instabilities such as stick-slip in the simulator, which means that it is difficult to say how exact drilling performance of the controller would perform in a real environment. Another issue was that the WOB controller in the simulator controls the speed of the drawworks in order to get a stable WOB. This had large inertia, so the drawworks could not keep up with the recommendations from the MPC controller. Also, the simulator could not simulate the effect of rig heave on ROP, and it was proposed to make an assumption of the ROP development such that a controller with and without the heave compensation could be compared. Further improving the verification strategy requires to pursue new simulation models that fully capture the dynamics of rig heave.

8

Conclusion

This thesis has investigated the possibility of using rig heave predictions to optimize the drilling process through the development of a heave prediction module together with an MPC. MRU measurements from the rig are passed through a LSTM artificial neural network, providing rig heave predictions multiple seconds ahead in time. The MPC selects the optimal combination of WOB and RPM based on a linearized Hareland ROP model corrected for the predicted heave motion through a heave correction factor. The coefficients of the Hareland model are calibrated at each timestep through sampling a sliding data window and minimizing the least squares error of the estimated ROP.

Using a LSTM artificial neural network to predict the heave motion seconds ahead of time based on MRU measurements has shown promising results. The method proved to have the best accuracy for shorter time horizons. The performance of the heave compensated MPC was assessed through two simulation cases. The first simulation case involved running the controller in advisory mode based on downhole sensor data from a real well in the North Sea. The second case assessed the closed-loop performance of the controller by using the OpenLab Drilling Simulator. Both simulation cases indicate that the controller has promising abilities to mitigate drilling instabilities due to rig heave by predicting the rig heave motion ahead in time, and compensating for it in the controller.

8.1 Further Work

During the work carried out in this thesis, there were identified further work actions which are outlined in the list below.

- Explore how to improve the accuracy in the heave prediction module for longer prediction horizons. Possibly through including wave radar data as a feature in the artificial neural network. By introducing the properties of the incident waves, it could potentially improve the predictions. Another potential measure to increase the accuracy is to combine the data-driven model that have been outlined in this thesis with a physics based model.

- Look into a strategy for updating the weights in the artificial neural network online, such that the training process can happen continuously and not requiring to be fully re-trained as the implemented system in this thesis.

- Investigate the possibility to use the heave predictions as input to other operational decision making systems onboard the rig, or even for other marine vessels such as a wind turbine installation vessel.

- Look into steering the drawworks directly, by taking in the heave predictions in the controller steering the drum rotational speed. In the work that have been presented, the drawworks action is provided through the increased or decreased WOB compensation through the MPC.

- Provide future values of the UCS to the MPC formulation. The MPC can therefore guide the drilling process to modify parameters based on known formation properties beforehand (proactive), and not after it has been detected.

Bibliography

Aadnoy, B., 2011. *Modern Well Design*. Taylor & Francis Group. ISBN: 978-0367577131.

Abadi, M., et al., 2015. TensorFlow: Large-scale machine learning on heterogeneous systems. URL: `https://www.tensorflow.org/`.

Afaq, S., Rao, D.S., 2020. Significance of epochs on training a neural network. *International Journal Of Scientific & Technology Research* 9. URL: `https://www.ijstr.org/final-print/jun2020/Significance-Of-Epochs-On-Training-A-Neural-Network.pdf`.

API, 1993. *Recommended Practice for Design, Selection, Operation and Maintenance of Marine Drilling Riser Systems*, American Petroleum Institute. URL: `https://pslcolombia.com/documentos/API%20RP%2016Q%20Marine%20Drilling%20Risers%201993.pdf`. (Accessed: 2023-04-22).

API RP 7G, 1998. *Recommended Practice for Drill Stem Design and Operation Limits*, American Petroleum Institute. (Accessed: 2023-04-22).

Badgwell, T., Pastusek, P., Kumaran, K., 2018. *Auto-Driller Automatic Tuning*. *Society of Petroleum Engineers* SPE-191417-MS. URL: `https://doi.org/10.2118/191417-MS`, doi:`10.2118/191417-MS`.

Balchen, J.G., Andresen, T., Foss, B.A., 2016. *Reguleringsteknikk*. Institutt For Teknisk Kybernetikk, NTNU, Trondheim. ISBN: 978-82-7842-202-1.

Bingham, M., 1964. A New Approach To Interpreting Rock Drillability. Oil Gas J. 94-101. URL: `https://books.google.no/books/about/A_New_Approach_to_Interpreting_Rock_Dril.html?id=NpmJAQAACAAJ&redir_esc=y`.

Bjørlo, J., 2022. *Specialization Project (TMR4510 - Marine Control Systems)*. Norwegian University of Science and Technology, Trondheim .

Bonner, S., Burgess, T., Clark, B., Decker, D., Orban, J., Prevedel, B., 1993. Measurements at the Bit: A New Generation of MWD Tools. *Oilfield Review* 01. URL: `https://www.slb.com/-/media/files/oilfield-review/p44-54`.

Bourgoyne, A.T., J., Young, F.S., J., 1974. A Multiple Regression Approach to Optimal Drilling and Abnormal Pressure Detection. *Society of Petroleum Engineers Journal* 14, 371–384. URL: `https://doi.org/10.2118/4238-PA`, doi:`10.2118/4238-PA`.

Branch, M.A., Coleman, T.F., Li, Y., 1999. A subspace, interior, and conjugate gradient method for large-scale bound-constrained minimization problems. *SIAM Journal on Scientific Computing* 21, 1–23. URL: `https://doi.org/10.1137/S1064827595289108`, doi:`10.1137/S1064827595289108`.

Brownlee, J., 2023. Machine learning mastery blogposts. URL: `https://machinelearningmastery.com/`.

Chollet, F., et al., 2015. Keras. URL: `https://keras.io`.

Chung, J., Gulcehre, C., Cho, K., Bengio, Y., 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. *Neural Computation* URL: `https://doi.org/10.48550/arXiv.1412.3555`, doi:`arXiv:1412.3555`.

DNV, 2010. *ENVIRONMENTAL CONDITIONS AND ENVIRONMENTAL LOADS*. DNV-RP-C205. URL: `https://home.hvl.no/ansatte/tct/FTP/H2021%20Marinteknisk%20Analyse/Regelverk%20og%20standarder/DnV_documents/RP-C205.pdf`. (Accessed: 20 September 2022).

Eaton, B.A., 1975. The Equation for Geopressure Prediction from Well Logs. URL: `https://doi.org/10.2118/5544-MS`, doi:`10.2118/5544-MS`. SPE-5544-MS.

Faltinsen, O.M., 1990. *Sea Loads On Ships And Offshore Structures*. Cambridge University Press. ISBN: 978-0521458702.

Fausett, L., 1993. *Fundamentals of Neural Networks: Architectures, Algorithms And Applications*. Pearson. ISBN: 978-0133341867.

Fivelstad, O., Verhoef, R., Ogg, J., Davila, S., 2014. Dual Active Heave Drilling Drawworks: From Concept to Operational Life. *SPE/IADC Drilling Conference and Exhibition* URL: `https://doi.org/10.2118/168019-MS`, doi:`10.2118/168019-MS`. SPE-168019-MS.

Gravdal, J., Sui, D., Nagy, A., Saadallah, N., Ewald, R., 2021. A hybrid test environment for verification of drilling automation systems. *SPE/IADC International Drilling Conference and Exhibition* SPE/IADC-204064-MS. doi:`https://doi.org/10.2118/204064-MS`.

Gravdal, J., et al., 2019. Openlab: Design and applications of a modern drilling digitalization infrastructure. *SPE Norway One Day Seminar* SPE-195629-MS. doi:`https://doi.org/10.2118/195629-MS`.

Guo, X., Zhang, X., Tian, X., Li, X., Wenyue, L., 2021. Predicting heave and surge motions of a semi-submersible with neural networks. *Applied Ocean Research* 112. URL: `https://www.sciencedirect.com/science/article/pii/S0141118721001851`, doi:`https://doi.org/10.1016/j.apor.2021.102708`.

Halliburton, 2019. DrillDOC® Drilling Downhole Optimization Collar URL: `https://cdn.brandfolder.io/VUJJLY3X/as/q1vy7j-3t1fe8-acz5vc/DrillDOC-Drilling-Collar-Datasheet.pdf`. (Accessed: 1 April 2023).

Hareland, G., Rampersad, P., 1994. Drag - Bit Model Including Wear All Days. URL: `https://doi.org/10.2118/26957-MS`, doi:`10.2118/26957-MS`. SPE-26957-MS.

Hatleskog, J., Dunnigan, M., 2006. Heave compensation simulation for non-contact operations in deep water doi:`10.1109/OCEANS.2006.307096`.

Hochreiter, S., Schmidhuber, J., 1997. Long short-term memory. *Neural Computation* 9. URL: `https://doi.org/10.1162/neco.1997.9.8.1735`.

Huisman, 2019. *Heave Compensated Floor (HCF)*. URL: `https://www.huismanequipment.com/documenten/brochures_2019/huisman_product_brochure_drilling_07-2019_v3.pdf`. (Accessed: 2023-04-22).

Jellison, M.J., Prideco, G., Hall, D.R., 2003. Intelligent Drill Pipe Creates the Drilling Network All Days. URL: `https://doi.org/10.2118/80454-MS`, doi:`10.2118/80454-MS`. SPE-80454-MS.

Johannessen, M., Myrvold, T., 2010. *Stick-Slip Prevention of Drill Strings Using Nonlinear Model Reduction and Nonlinear Model Predictive Control*. Master's Thesis Norwegian University of Science and Technology (NTNU). URL: `http://hdl.handle.net/11250/259796`. (Accessed: 1 October 2022).

Kingma, D.P., Ba, J., 2015. Adam: A Method for Stochastic Optimization. 3rd International Conference for Learning Representations, San Diego doi:`https://doi.org/10.48550/arXiv.1412.6980`.

Kommedal, A., 2021. *Development of Model Predictive Control for Setpoint Regulation in Offshore Drilling*. Master's Thesis Norwegian University of Science and Technology (NTNU). URL: `https://hdl.handle.net/11250/2788518`. (Accessed: 4 February 2023).

Kyllingstad, A., Nessjøen, P., 2009. A new stick-slip prevention system. SPE/IADC Drilling Conference and Exhibition SPE/IADC 119660. URL: `https://doi.org/10.2118/119660-MS`.

Matthews, W., Kelly, J., 1967. How to predict formation pressure and fracture gradient. Oil and Gas Journal .

Mitchell, R., Miska, S., 2011. *Fundamentals of Drilling Engineering*. Society of Petroleum Engineers. ISBN: 978-1555632076.

National Oilwell Varco, 2018. *Top Drive Technologies*. URL: `https://www.nov.com/-/media/nov/files/products/rig/rig-equipment/top-drive-systems/top-drive-technologies-brochure.pdf`. (Accessed: 20 April 2023).

Nielsen, U., Brodtkorb, A., Jensen, J., 2018. Response predictions using the observed autocorrelation function. *Marine Structures* 58. URL: `https://doi.org/10.1016/j.marstruc.2017.10.012`.

Odfjell Drilling, 2022. *Deepsea Nordkapp*. URL: `https://www.odfjelldrilling.com/rig/deepsea-nordkapp/`. Accessed: 1 November 2022.

Pastusek, P., et al., 2016. Drill rig control systems: Debugging, tuning, and long term needs. *Society of Petroleum Engineers* SPE-181415-MS. URL: `https://doi.org/10.2118/181415-MS`.

Payette, G.S., Pais, D., Spivey, B., Bailey, L.W.J.R., Pastusek, P., Owens, M., 2015. Mitigating drilling dysfunction using a drilling advisory system: Results from recent field applications. *International Petroleum Technology Conference* IPTC-18333-MS. URL: `https://doi.org/10.2523/IPTC-18333-MS`.

Qin, S., Badgwell, T.A., 2003. A survey of industrial model predictive control technology. *Control Engineering Practice* 11, 733–764. URL: `https://www.sciencedirect.com/science/article/pii/S0967066102001867`, doi:`https://doi.org/10.1016/S0967-0661(02)00186-7`.

Rick von Flatern, 2016. *Blowout Preventers*. URL: `https://www.slb.com/-/media/files/oilfield-review/defining-bops.ashx`. (Accessed: 22 April 2023).

Saipem, 2022. *SCARABEO 8, Brochure*. URL: `https://www.saipem.com/sites/default/files/2018-12/1612spm_scara8_web.pdf`. (Accessed: 10 May 2023).

Salehinejad, H., Sankar, S., Barfett, J., Colak, E., Valaee, S., 2018. Recent Advances in Recurrent Neural Networks. doi:`https://doi.org/10.48550/arXiv.1801.01078`, arXiv:1801.01078.

Schmidt, R.M., 2019. Recurrent neural networks (rnns): A gentle introduction and overview doi:`https://doi.org/10.48550/arXiv.1912.05911`.

da Silva, I.N., Spatti, D.H., 2016. *Artificial Neural Networks*. Springer Cham. ISBN: 978-3-319-43161-1.

Soares, C., Daigle, H., Gray, K., 2016. Evaluation of pdc bit rop models and the effect of rock strength on model coefficients. *Journal of Natural Gas Science and Engineering* 34, 1225–1236. URL: `https://www.sciencedirect.com/science/article/pii/S1875510016305595`, doi:`https://doi.org/10.1016/j.jngse.2016.08.012`.

Stathakis, D., 2008. How many hidden layers and nodes? *International Journal of Remote Sensing* 30. doi:`https://doi.org/10.1080/01431160802549278`.

Sugiura, J., 2008. Systematic Testing with Push- and Point-the-Bit Rotary-Steerable Systems Leads to the Optimal BHA Design for Stability, Steerability and Borehole Quality URL: `https://cdn.brandfolder.io/VUJJLY3X/as/q1vy7j-3t1fe8-acz5vc/DrillDOC-Drilling-Collar-Datasheet.pdf`. AADE-08-DF-HO-39.

Sui, D., Nybø, R., Azizi, V., 2013. Real-time Optimization of Rate of Penetration during Drilling Operation doi:`https://doi.org/10.1109/ICCA.2013.6564893`.

Takami, T., Nielsen, U., Jensen, J., 2021. Real-time deterministic prediction of wave-induced ship responses based on short-time measurements. *Ocean Engineering* 221. doi:`https://doi.org/10.1016/j.oceaneng.2020.108503`.

Virtanen, P., et al., 2020. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods* 17. doi:`10.1038/s41592-019-0686-2`.

Warren, T., 1987. Penetration-Rate Performance of Roller-Cone Bits. SPE Drilling Engineering doi:`https://doi.org/10.2118/13259-PA`.

Woodacre, J., Bauer, R., Irani, R., 2015. A review of vertical motion heave compensation systems. *Ocean Engineering* 104, 140–154. URL: `https://www.sciencedirect.com/science/article/pii/S0029801815001729`, doi:`https://doi.org/10.1016/j.oceaneng.2015.05.004`.

Zhakatayev, A., Rakhim, B., Adiyatov, O., Baimyshev, A., Varol, H.A., 2017. Successive linearization based model predictive control of variable stiffness actuated robots doi:`10.1109/AIM.2017.8014275`.

Appendix

A Wellpath OpenLab

MD	Azi.	Incl.	MD	Azi.	Incl.	MD	Azi.	Incl.
0	0	0	852.51	230.49	12.72	1440	236.61	40.31
43.5	0	0	862.78	230.9	13.54	1450	236.82	41.16
152	40.6	0	873.08	231.27	14.36	1460	237.02	41.99
291	360	0	883.43	231.5	15.23	1470	236.98	42.8
301	180.37	0.27	893.81	231.58	15.89	1480	236.76	43.59
311	174.74	0.29	904.21	231.69	16.33	1490	236.76	44.51
321	172.9	0.24	914.65	231.86	16.71	1500	236.82	45.37
331	192.63	0.17	925.1	231.83	17.13	1510	236.87	46.35
341	232.7	0.17	935.57	231.75	17.57	1520	237.31	47.14
351	231.94	0.19	946.07	231.9	17.93	1530	237.94	47.62
361	195.7	0.25	956.6	232.03	18.29	1540	238.17	47.96
371	194.19	0.13	967.14	232.05	18.71	1550	238.24	48.12
381	309.46	0.1	977.71	232.06	19.17	1560	237.82	48.25
391	288.51	0.12	988.32	232.2	19.69	1570	238.01	48.24
401	248.95	0.14	998.96	232.28	20.28	1580	237.99	48.3
411	214.46	0.25	1009.64	232.19	20.97	1590	237.79	48.32
421	193.84	0.39	1020.38	232.37	21.85	1600	237.58	48.3
431	173.65	0.33	1031.19	232.83	22.78	1610	237.3	48.48
441	115.8	0.23	1042.07	233.08	23.54	1620	237.29	48.37
451	77.82	0.37	1053	233.17	24.17	1630	237.35	48.41
461	72.57	0.39	1063.99	233.29	24.61	1640	237.41	48.37
471	95.72	0.2	1074.99	233.49	24.77	1650	237.05	48.18
481	162.91	0.2	1086	233.78	24.72	1660	237.05	48.12
491	154.82	0.31	1097.01	233.83	24.66	1670	237.18	48.27
501	149.29	0.26	1108.01	233.64	24.64	1680	237.37	48.37
511	152.96	0.23	1119.01	233.55	24.59	1690	237.29	48.59
521	145.39	0.37	1130	233.57	24.52	1700	237.32	48.62
531	145.9	0.44	1140.99	233.7	24.48	1710	237.86	48.28
541	144.1	0.39	1151.98	233.73	24.38	1720	237.84	48.07
551	134.02	0.27	1162.95	233.59	24.23	1730	238.16	47.82
561	144.09	0.19	1173.91	233.51	24.17	1740	238.15	47.82
571	136.05	0.32	1184.88	233.46	24.21	1750	238.08	47.82
581	129.87	0.46	1195.84	233.53	24.21	1760	238.28	47.91
591	139.12	0.43	1206.8	233.82	24.07	1770	238.43	47.92
601	141.82	0.48	1217.74	234.01	23.93	1780	238.31	48.02
611	135.37	0.54	1228.68	234.03	23.91	1790	238.39	47.72
621	127.57	0.47	1239.62	234.03	23.86	1800	238.3	47.7
631	122.87	0.39	1250.55	234.05	23.77	1810	238.08	47.82
641.01	129.13	0.39	1260	237.71	24.71	1820	238	47.91
651.01	143.68	0.44	1270	239.38	25.51	1830	237.82	47.97
661.01	152.3	0.55	1280	239.75	26.06	1840	237.8	47.84
671.01	161.7	0.71	1290	239.95	26.62	1850	237.76	47.86
681.01	179.94	0.86	1300	239.76	27.18	1860	237.79	47.88
691.01	203.66	1.18	1310	239.56	27.8	1870	237.75	47.82
701.01	218	1.86	1320	238.99	28.76	1880	237.98	47.64
711.02	222.5	2.6	1330	238.92	29.65	1890	238.39	47.38
721.03	223.45	3.32	1340	238.52	30.47	1895.96	238.48	47.52
731.05	223.02	4.04	1350	238.02	31.46	1917.09	240.3	47.34
741.08	222.88	4.75	1360	237.78	32.29	1951.24	239.64	47.66
751.12	224.41	5.48	1370	237.54	33.23	1978.29	240.38	47.53
761.18	225.84	6.27	1380	237.35	34.15	2005.57	242.56	47.54
771.25	226.82	7.1	1390	236.89	35.25	2032.83	246.01	47.52
781.33	227.48	7.91	1400	236.86	36.28	2060.11	250.16	47.66
791.44	228.12	8.63	1410	236.78	37.35	2087.38	253.25	47.03
801.56	229	9.36	1420	236.59	38.44	2124.31	256.97	45.44
811.71	229.57	10.12	1430	236.72	39.45	2151.72	259.17	44.66
821.88	229.78	10.73	1440	236.61	40.31	2179.2	259.12	42.66
832.06	229.91	11.31	1450	236.82	41.16	2206.68	259	41.12
842.27	230.17	11.95	1460	237.02	41.99	2233.58	260.78	39.96

B Source Code

The source code can be accessed by scanning the QR code below, or by following the URL to the Git repository: `https://github.com/jonasbjorlo/mpc-drilling`.