NTNU
Kunnskap for en bedre verden

Department of Geosciences and Petroleum

Master's thesis in Petroleum Engineering

Condition monitoring system for all-electric subsea systems

Author:
Eirik Soland

June 2023

Acknowledgements

This Master's thesis finalizes the work of the 2-year Master of Science program Petroleum Engineering at Norwegian University of Science and Technology. The thesis was written during the spring of 2023 at the Department of Geoscience and Petroleum.

I would like to express my sincere gratitude to Professor II Audun Faanes, my supervisor, and Professor Tor Berge Gjertsvik, my co-supervisor, for their invaluable guidance and support throughout my master's thesis. The many meetings and all encouragement through the process is something I really appreciate.

I am also thankful to Ludvig Björklund and Mehman Ahmadli, PhD candidates, for their valuable input and sharing of experiences. Additionally, I extend my appreciation to Børge Pettersen from Equinor ASA for sharing crucial information and providing valuable insights.

Eirik Soland
Trondheim, 03.06.23

Abstract

Today, most subsea systems are developed with electro-hydraulic control systems. This system has been developed over many decades into the well-functioning and reliable system it is today. In these systems, the Topside communicates with power, signals, and hydraulics through umbilicals. One of the many important tasks of the control system is to give the operator control over the tree valve, as well as to have a functioning system for situations where communication is cut off in an emergency. Therefore, it is important to conduct regular tests to verify that everything works as it should, especially those components that are to ensure safety in an emergency. An example of such tests is leak testing of gate valves in the X-mas tree. The next step for the industry is to develop fully electric subsea systems. By replacing hydraulic actuators with electrical ones, the amount of data available for analysis will drastically increase.

For hydraulic valves, the position is monitored based on the hydraulic fluid used and pressure measurements. This is mainly to confirm either fully- open or closed position. All- electric systems introduce an electric motor in the actuator. The use of an electric motor will make a whole range of new data available. Linear position from the number of revolutions, immediate power consumption, and rotation speed are new data that will be available only from the motor. In addition, precise position manipulation opens the possibility for partial stroke testing. In such a test, the functionality of the valve can be tested without stop in production. One of the main challenges for further development is to set up systems that use this data in a sensible way. An example of such a system has been examined and defined by two Python programs.

Two programs have been constructed for a gate valve: one to simulate data (as this was missing), and one for a digital twin. The purpose of the programs is to demonstrate examples of how new data, in combination with traditional theory, can be used to condition monitor the valve. The Simulator generates data for force measurements made on a proposed load cell on the valve stem, as well as the electric power used by the motor. Other information such as time, position, pressure, and rpm is also generated. The data is exported to Excel as time series and represents the information being collected from the various sensors. Faults over time in the form of decreasing efficiency on the lead screw, increasing flow behavior index for the lubricant in the actuator, decreasing friction on the valve stem seal, and leakage over the valve are included. Assumptions have also been made concerning the correlation between faults: the chance and magnitude of change in leakage and friction on the stem seal increase for higher differential pressure over the valve when opening after leak testing.

The Digital Twin collects the time series and calculates the expected linear force on the valve stem in two stages. One by using data for RPM and electric power used by the motor, and one by using valve position and pressure data from the well. The idea was to assume a faultless system over time, and then compare the estimated time series with the measured ones. The results from the analysis show that it is easy to show a change in faults over time for such a system. By comparing the change in electric power

with deviations in motor and valve performance, one can see the reason for the abnormal change in power consumption. The change in efficiency of the actuator was presented for two different rpms over the number of cycles. It can then be seen that there is gradually a small increase in the flow behavior index, as there is a deviation in efficiency for high and low rpm. The deviation in necessary valve linear force decreases with time, as the friction on the valve stem seal decreases. A clear correlation between deviation in required force on the valve, and differential pressure when opening the valve can be seen. This is because the packing is assumed to wear out quicker for higher differential pressures.

Sammendrag

I dag er de fleste subsea systemene utviklet med elektro- hydrauliske kontrollsystemer. Dette systemet har i mange tiår blitt utviklet til det velfungerende og pålitelige systemet det er i dag. I disse systemene kommuniserer Topside med strøm, signaler og hydraulikk gjennom umbilicals. En av de mange viktige oppgavene til kontrollsystemet er å gi operatøren kontroll over ventiltreet, samt å ha et velfungerende system for situasjoner hvor kommunikasjon blir kuttet av i en nødsituasjon. Derfor er det viktig å gjennomføre regelmessige tester for å verifisere at alt fungerer som det skal, spesielt de komponentene som skal ivareta sikkerheten under en nødsituasjon. Et eksempel på slike tester er lekasjetesting av portventiler i ventiltreet. Det neste steget for bransjen er å utvikle helelektriske undervannssystemer. Ved å bytte ut hydrauliske aktivatorer med elektriske, vil mengden data tilgjengelig for analyse øke drastisk.

For hydrauliske ventiler monitoreres posisjonen basert på hydraulikk brukt og trykkmålinger. Dette er hovedsakling for å bekrefte åpen eller lukket posisjon. Helelektriske systemer introduserer en elektrisk motor i aktivatoren. Ved bruk av en elektrisk motor vil en hel rekke ny data bli tilgjengelig. Lineær posisjon fra antall omdreininger, umiddelbart strømforbruk og omdreiningshastighet er ny data som blir tilgjengelig kun fra motoren. I tillegg gjør nøyaktig posisjonsmanipulering at det åpnes mulighet for partial stroke test. Ved en slik test kan funksjonaliteten av ventilen testes uten stopp i produksjonen. En av de viktigste utfordringene for videreutvikling er å sette opp systemer som benytter disse dataene på en fornuftig måte. Et eksempel på et slikt system har blitt undersøkt og definert ved to Python-program.

Det er blitt konstruert to program for en gate-ventil: et for å simulere data (da dette manglet), og et for en digital tvilling. Hensikten med programmene er å demonstrere eksempler på hvordan ny data, sammen med tradisjonell teori kan benyttes for å tilstandsmonitorere ventilen. Simuleringsprogrammet produserer data for kraftmålinger gjort på en foreslått lastcelle på ventilstammen, samt elektrisk effekt brukt av motoren. Annen informasjon som tid, posisjon, trykk og rpm blir også generert. Dataen blir eksportert til Excel som tidsserier og representerer informasjonen som blir hentet inn fra systemet. Feil over tid i form av synkende effektivitet på ledeskrue, økende flow behaviour index for smørevæske i aktivatoren, synkende friksjon på pakning til ventilstammen og lekkasje over ventilen er inkludert. Det er også gjort antakelser om sammenheng mellom feil: sjanse og størrelse for endring i lekkasje og friksjon på pakning øker for høyere differensialt trykk over ventilen ved åpning etter lekkasjetest.

Den digitale tvillingen tar inn tidsseriene og kalkulerer forventet lineær kraft på ventilstammen i to omganger. Én ved å bruke data for RPM og elektrisk effekt brukt av motoren, og én ved å bruke ventilposisjon og trykkdata fra brønnen. Ideen var å anta et feilfritt system over tid, og deretter sammenligne estimerte tidsserier med de målte. Resultatene fra undersøkelsen viser at det enkelt går an å vise endring i feil over tid for et slikt system. Ved å sammenligne endring i elektrisk effekt med avvik i motor- og ventil ytelse kan en se grunnen til unormal endring i effekt. Endringen av effektiviteten til

aktivatoren ble fremstilt for to ulike rpm over antall sykluser. Det kan da sees at det etter hvert er en liten økning i flow behaviour index, da det er lavere effektivitet for høy rpm. Avvik i nødvending ventilkraft synker med tid, da friksjonen på pakningen til ventilstammen synker. En klar korrelasjon mellom avvik for behov for kraft til ventilen og differensialtrykk ved åpning kan sees. Dette da pakningen antas å slites mer for høyere differensialtrykk.

List of Abbreviations

AAV Annulus Access Valve. 12

AC Alternating Current. 10

AIV Annulus Isolation Valve. 19

AMV Annulus Master Valve. 12, 19

ASV Annulus Swab Valve. 12

AVIV Annulus Vent Isolation Valve. 19

AVV Annulus Vent Valve. 19

AWV Annulus Wing Valve. 19

BMV Bleed Monitoring Valve. 19

CAPEX Capital Expenditures. 8

CIV Chemical Injection Valve. 19

Cv Flow Coefficient. 14

DCV Directional control valve. 7

DHSV Down Hole Safety Valve. 12, 19

EPU Electrical Power Unit. 6

eSCE electric Subsea Control Module. 10

FPDD First-Principle Data-Driven. 33

FPSO Floating Production Storage and Offloading. 1

HPU Hydraulic Power Unit. 6

HSE Health, Safety and Environment. 8

HVDC High Voltage Direct Current. 10

MCS Master Control Station. 6, 7

MCSA Motor Current Signature Analysis. 34

MEGIV MEG Isolation Valve. 19

MEHCS Multiplexed Electro-Hydraulic Control Systems. 1

OPEX Operating Expenses. 8

PBT Plug Bleed and Test Valve. 19

PCV Production Choke Valve. 17, 19

PMV Production Master Valve. 12, 19

PSV Production Swab Valve. 12

PWV Production Wing Valve. 12, 19

SCM Subsea Control Module. 5, 6

SEM Subsea Electronic Module. 7

SOC State Of Charge. 63

SOH State Of Health. 63

WCIV Workover Chemical Injection Valve. 19

WHSIP Wellhead Shut In Pressure. 17

WOV Workover Valve. 19

XOV Crossover Valve. 12, 19

Table of contents

1 Introduction **1**

1.1 Motivation . . . 1

1.2 Objective . . . 2

1.3 Outline of the Thesis . . . 3

2 Literature Review **4**

2.1 Subsea components . . . 4

2.2 Subsea Control Systems . . . 6

2.3 MEHCS . . . 6

2.4 All-Electric Control Systems . . . 8

2.5 Proposed all- electric system for X-mas tree valves . . . 9

2.6 X-mas tree . . . 11

2.7 Choke valves . . . 13

2.8 Electric chokes . . . 14

2.9 The flow coefficient . . . 14

2.10 Gate valves . . . 15

2.10.1 Balanced and unbalanced gate valves . . . 16

2.11 Procedure for valve testing . . . 17

2.12 Calculations for unbalanced gate valve . . . 19

2.12.1 Local forces . . . 20

2.12.2 Net Forces . . . 22

2.13 Motor and gears . . . 23

2.13.1 Working Principles of AC Motors . . . 24

2.13.2 Root Mean Square (RMS) . . . 24
2.13.3 Electrical power to torque . . . 25
2.14 Load cell . . . 26
2.15 Rotation to linear movement . . . 27
2.16 Non newtonian fluids . . . 29
2.16.1 Newtonian Fluids . . . 29
2.16.2 Non-Newtonian Fluids . . . 29

3 Maintenance Philosophies 32
3.1 System modelling . . . 33
3.2 Digital twin . . . 33
3.3 Motor Current Signature Analysis . . . 34
3.4 Machine learning . . . 34

4 Method 36
4.1 System description . . . 37
4.2 Available sensor data . . . 37
4.3 Input data . . . 38
4.4 Failure modes . . . 40
4.5 Computational method . . . 41
4.6 Assumptions and premises for failure . . . 42
4.7 Algorithm description . . . 43

5 Results 48
5.1 Excel interface . . . 48
5.1.1 "Data" sheet for Simulator . . . 48
5.1.2 "Digital Twin" sheet for Digital twin . . . 49

5.1.3 "Analysis" sheet 49
5.2 Valve fingerprints 50
5.3 Individual Failure modes 52
5.4 Combinations and correlation 56
5.5 Forecasting 59
5.6 Partial stroke testing 59

6 Discussion **61**
6.1 Leakage test of X-mas tree valves 61
6.2 Required action 61
6.3 Partial stroke 62
6.4 Torque transducer 62
6.5 Excluding the load cell 62
6.6 Monitoring of electric chokes 63
6.7 Monitoring of battery cells 63

7 Conclusions and recommendations for future work **64**
7.1 Recommendations for future work 64

Appendix **65**
A Code for Digital Twin 65
B Code for Simulator 68

References **76**

List of Figures

1 Illustration of a subsea system [21]. 4

2 MEHCS system [20] 6

3 Architecture of an all-electric system [24] 9

4 Schematic of all-electric system, self-produced in [31] with inspiration from [20] and [24]. 10

5 Schematic of vertical X-mas tree [1] 12

6 Needle & Seat choke [34] 13

7 Illustration of a stepping actuator with two cylinders[32]. 14

8 Illustration of a hydraulic gate valve with actuator, adapted from [38] . . . 16

9 Schematic of an unbalanced gate valve: open position[9]. 17

10 Test procedure of the production cross [19]. 18

11 Loads and friction on gate seats – closed valve [9]. 20

12 Stem force vs position of gate valve with spring[9]. 22

13 Stem force vs position of gate valve without spring[9]. 23

14 Illustration of principle of AC motor [35]. 24

15 Illustration of a strain gauge load cell [37]. 27

16 Illustration of a ball screw [4]. 27

17 Illustration of an ACME screw [5]. 28

18 Shear stress vs rate of shear strain for different fluid models [18]. 30

19 System illustration[31] 37

20 Logic overview 44

21 Timeseries from Simulator displayed in Excel 48

22 Simulator data displayed for each cycle 49

23 Timeseries generated by Digital twin 49

24 Digital twin data for actuator displayed for each cycle 49
25 Analysis table for actuator 50
26 Analysis table for valve 50
27 Linear force vs position for the first cycle 51
28 Valve fingerprint for 3 cycles 51
29 Change in required linear force to operate valve vs cycle 52
30 Change actuator performance [lbf] vs cycle for "Crack Open" 53
31 Efficiency of actuator vs cycle 54
32 Effect of RPM: displays difference in efficiency between 720 and 900 RPM 54
33 Combo-chart with change in required electrical power vs motor and valve deviation 55
34 Average differential pressure for test period vs cycle 56
35 Differential pressure during pressure test vs time [s] 57
36 "dP_{random}" vs ABS change in required linear force to operate valve 58
37 Failed attempt to see correlation in leakage vs opening dP 58
38 Change in required linear force to operate valve vs cycle, forecast 59
39 Valve fingerprint of partial stroke test 60

List of Tables

1 Input for Simulator 38
2 Input for Digital twin 40
3 Failure modes considered 41

1 Introduction

1.1 Motivation

The rapid and progressive advancements in technology over recent decades have revolutionized numerous industrial sectors, including oil and gas production. Current subsea systems largely employ Multiplexed Electro-Hydraulic Control Systems (MEHCS) coupled with umbilicals that supply hydraulic fluids, chemicals, signals, and electric power to the fields. These technologies have facilitated remarkable growth and expansion in this sector, yet simultaneously pose several challenges, particularly concerning system reliability, maintenance, environmental impact, and efficiency.

The complex nature of the subsea systems results in high maintenance needs and increases the risk of system failure. It often requires regular interventions, which are costly, time-consuming, and potentially hazardous due to the harsh subsea environment. Moreover, hydraulic leaks from these systems can have a significant negative impact on the marine ecosystem, raising significant environmental concerns.

All-electric subsea systems represent an innovative solution designed to address these issues. Expected to be the next evolution in oil and gas production, these systems propose to offer increased reliability, reduced maintenance needs, enhanced operational efficiency, and significantly lower environmental impact.

Electric systems have been tested and put into production previously, for instance, with the Goliat FPSO in the Barents Sea. The FPSO introduced several innovative solutions such as valve footprint measurements and vibration analysis for electric motors, compressors, and turbines when it went into production in the fall of 2014. The production vessel was seen as a clear step in the right direction with increased automation and far more accurate condition monitoring [33]. However, the plant has unfortunately experienced several setbacks in recent times in the form of shutdowns by PTIL due to serious errors in the electric installation on deck. These errors included that half of the electric motors on board had not been condition inspected [26]. Goliat FPSO highlights both the clear advantages and also the challenges related to the use of all-electric systems. There are clear benefits when it comes to the potential for monitoring, but replacing well-established electro-hydraulic systems requires a thorough review of safety systems. Both new routines and proper infrastructure has to be put in place.

For successful adoption and operation, all-electric subsea systems require effective and efficient condition monitoring. This necessity to understand the real-time health and operational status of subsea components will ensure timely detection of anomalies and faults. Implementing such systems could potentially avert catastrophic failures and financial losses, leading to more secure, sustainable, and economically efficient oil and gas production.

An important component to address in the safety system is the X-mas tree. In an emergency, it should shut off the flow from the well. The valves that need to be closed during such an event are always open during production. Therefore, regular monitoring of whether they are functioning as intended is crucial. In today's solutions, such valves are tested exclusively by leakage testing. Such tests will also be important for all-electric systems, but a range of new data will be available here. With a well-functioning system, it will be easier to determine where a potential error occurs. This results in a greater number of options when preventive measures are to be taken or repairs carried out. For example, if it can be determined where in a valve the failure occurs, it can be decided whether pulling the X-mas tree is necessary or not. One can also compare operational routines with the development of errors and look for correlation. If a connection is found, action patterns can be changed for less wear. A possibility introduced using an electric motor is partial stroke testing. This is the process of testing valves manoeuvrability by just partly closing it. Such tests allow for tests to be executed when the well is still producing, and can potentially uncover faults separately to the periodical leakage tests. This way, prevention measures can be established earlier to prevent further damage. Consequently, there are potentially significant economic savings.

In the event of communication loss with topside, a redundant system should exist to shut down the well automatically. For hydraulic valves, the solution has been a spring mechanism that constantly forces the valve towards the closed position. Therefore, the valve will close if hydraulic pressure is lost. For electric valves, the same requirements for a redundant system apply if the power supply to the motor driving the valve is cut. Here, there are mainly two proposed solutions - one with a spring as in the hydraulic valve and one with a separate set of battery cells. Although the spring solution has proven to be reliable, it includes some uncertainty about the state of the spring. By using batteries subsea that take over if the power supply from the topside is cut, better condition monitoring of the redundant system is enabled. This is because the condition of batteries can be monitored more easily and reliably. Batteries can also potentially be replaced more easily if needed. In addition, a challenge with the spring solution is that its size depends on factors such as installation depth, which makes the system less universal and needs to be adapted to the specific well. For these and other reasons mentioned, the battery solution has recently gained more traction in the industry and is the system that will be considered in this thesis.

1.2 Objective

This thesis aims to explore the development and application of condition monitoring systems specifically for all-electric subsea systems. There are many potential components to consider for an all-electric system, so a selection has been made. The components to be addressed for potential monitoring are gate valves in the X-mas tree as well as the choke.

Most of the emphasis will be assigned to the gate valve, which will include a demonstration of a proposed setup for condition monitoring. A Simulator program is constructed to generate data for an electric gate valve. The Digital Twin will be programmed to read and process the data for further analysis. Most of the time consumed for this thesis went into the coding of these two programs. Also, a review of relevant topics will be performed in a literature review.

Solutions that will be investigated:

- Solution for monitoring of gate valves will be presented, and the programmed algorithm will be presented in the form of pseudo-code in addition to the full code.
- Plots from analysis will be displayed to address parameters that could be compared.

1.3 Outline of the Thesis

- **Chapter 1: Introduction-** includes motivation and the objective of the thesis.
- **Chapter 2: Literature Review-** includes theory for both elementary description of a subsea- system as well as more specific theory concerning all-electric systems, valves, actuators and fluid dynamics.
- **Chapter 3: Maintenance Philosophies-** Theory concerning maintenance philosophy.
- **Chapter 4: Method-** Description of the possible solution found. Also presents the pseudo code for the Digital twin and the Simulator.
- **Chapter 5: Results-** Results from the code are presented, as well as plots from analysis.
- **Chapter 6: Discussion-** Results and solutions are discussed.
- **Chapter 7: Conclusions and recommendations for further work-** A conclusion is given in the format of a bullet list, as well as some keywords for further work.

2 Literature Review

2.1 Subsea components

A subsea system in oil and gas production refers to the collection of technologies and equipment that is necessary at the seabed for remote production. These systems are specially designed to withstand high pressures and low temperatures at great depths. They typically include completed wells, wellheads, X-mas trees, manifolds, pipelines, surface connections and control systems, all of which work together to bring hydrocarbons from the reservoir to the surface. The use of remotely operated vehicles (ROVs) is common for installation, maintenance, and inspection. A typical subsea configuration is shown in figure 1.

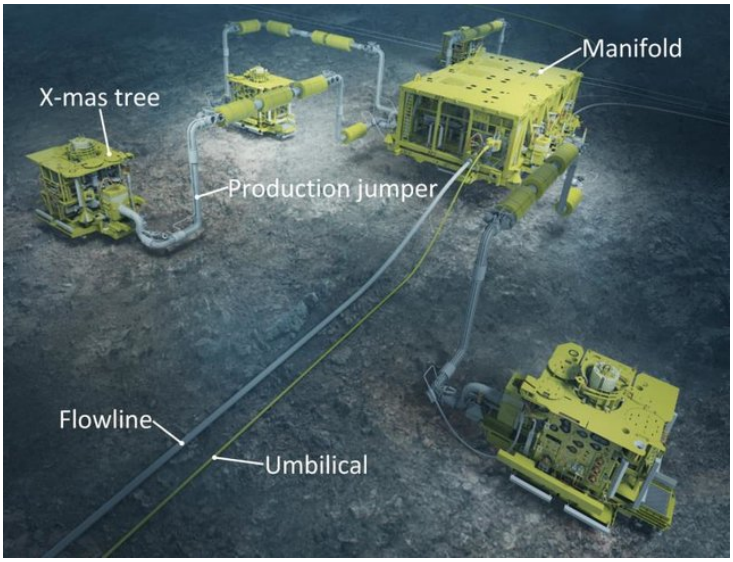

Figure 1: Illustration of a subsea system [21].

Completed Wells: Completed wells are the initial point of hydrocarbon extraction in a subsea system. These are wells that have been drilled, cased, and cemented. The well is completed with production tubing and various smaller components such as down-hole safety valve and potential gas lift valve, and is perforated to allow the oil or gas to flow into the well-bore. The process of completion involves ensuring that the well can efficiently produce oil or gas from the reservoir with a certain well integrity.

Wellheads: The wellhead serves as foundation for the rest of the subsea production system. It provides the suspension point of casing and tubing hanger, and provides pressure seals for the production equipment. The wellhead is essentially the interface between the well and the rest of the subsea system. Its main task is to provide a pressure seal for the casings and secure the production tubing.

X-mas Trees: The X-mas tree is installed on top of the wellhead. It includes a complex assemble of valves, pipes and fittings for the outlet of the well, and is crucial for controlling the flow of hydrocarbons from the reservoir to the surface. X-mas trees can also be used to control the injection of fluids like water or gas into the well, monitor pressure, and implement safety measures like shutting off the flow.

Manifolds: Manifolds are used to direct the flow of extracted hydrocarbons from the wells to the production facilities. They act like a switchboard, allowing operators to control which wells are active and to route oil and gas to different locations. Manifolds also serve the purpose of distributing injected water or gas to various wells in the reservoir.

Pipelines: Pipelines are an essential component of any subsea system, providing the conduit for the transportation of oil and gas from the subsea wells to the various subsea components such as manifolds, pumps and separators. They are designed to withstand extreme conditions and pressures, and are often insulated to maintain the temperature of the flowing hydrocarbons. The production jumper which serves as the producing connection between the X-mas tree and the manifold is an example of such pipeline.

Surface Connections: Surface connections are the various pipes and umbilicals which connect the subsea system with surface facilities, like processing plants or offshore platforms. This involves the flow-line and riser, which are pipelines that convey the extracted oil and gas from the seabed to the surface facilities. Surface connections also include umbilicals which are cables that transmit electrical power, hydraulic fluid and data.

Control Systems: Control systems are responsible for the monitoring and remote control of all subsea equipment. They often employ hydraulic and/or electric signals to operate the valves and other devices on the seafloor through the umbilicals. These systems ensure that the subsea production operations are functioning optimally and safely, while allowing for real-time data collection and analysis. The subsea control module (SCM) is located subsea, close to the production wells. The SCM functions as a link between surface operators and the various subsea components. All sensors used for monitoring can also be described as part of the control system. This includes P/T sensors, flow-meters, position sensor, acoustic sensors and more.

ROVs and AUVs: Remotely Operated Vehicles (ROVs) are used for the installation, maintenance, and inspection of subsea systems. They are remotely controlled and can maneuver in the deep-sea environment, equipped with cameras and tools to perform tasks like monitoring the integrity of equipment, detecting leaks, and performing repairs.

The description of subsea components was inspired by the specialization report written as preparation for this thesis[31].

2.2 Subsea Control Systems

The subsea control system is an essential part of a subsea production system. It has a lot of important functions but can be summarized as the communication link between subsea and topside. The control system provides the topside control over all the valves and steerable equipment subsea Additionally it provides data such as pressure, temperature and sand readings [8]. There will in this section be presented the two most time- relevant systems: MEHCS and all-electric. MEHCS is currently the most used control system, and has been enhanced through decades in the industry [8]. As for further enhancement, the all-electric system has been a rising candidate to replace the MEHCS. An initiative from Equinor was launched in 2010 as a potential solution for fulfilling future requirements for safety and environmental issues [19].

2.3 MEHCS

MEHCS can be described as a mix between an electric and a hydraulic system. The system utilizes electrical signals to activate valves, but the physical mechanisms that operate the valves are hydraulics. Such a system includes three main components topside:

- The Master Control Station, MCS, is the unit that receives data and controls the subsea equipment.
- The Master Control Station, MCS, is the unit that receives data and controls the subsea equipment.
- The Electrical Power Unit, EPU, Provides the system with electrical power. The Hydraulic Power Unit, HPU, provides the system with hydraulic power.

An illustration of a MEHCS system can be seen in figure 2.

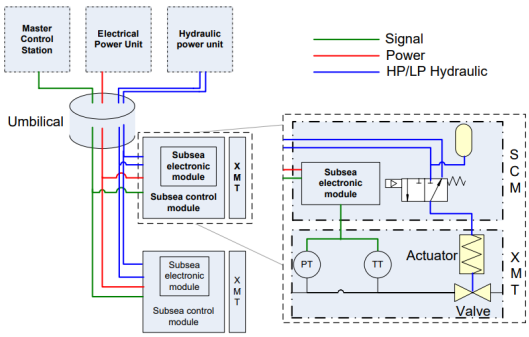

Figure 2: MEHCS system [20]

The three components communicate with the subsea system through umbilical and connect to the Subsea Control Module, SCM. The SCM has the purpose of handling and

distributing hydraulics, electrical power and signals received by either sensors or topside. It typically contains hydraulic distribution systems, accumulators, control valves and Subsea Electronics Module, SEM. The SEM distributes the electrical power and handles both output and input signals. The design of this system enables communication with multiple wells through a singular umbilical, thus making it a more popular choice for subsea applications than the sequenced hydraulic system.

The HPU has two main lines of hydraulics- for high and low pressure. The high-pressure line is connected to a subsea hydraulic accumulator that has the necessary volume and pressure to open the valve. The low-pressure line is for releasing pressure when closing. For a commonly used open loop system, this involves releasing the hydraulic fluid through a one-directional check valve on the seabed. The hydraulics used are usually water-based and approved for disposal. Regulations can however change as the impact on the environment becomes increasingly important [31].

Gate valves in the x-mas tree are operated by controlling the Directional Control Valve, DCV. The position of the DCV, which is controlled by the SEM, determines whether the low- or high-pressure line is in communication with the actuator. Hydraulic fluids are stored in subsea accumulators that are connected to the high-pressure line. When the valve is opened by the high-pressure line, the pressurized hydraulics in the accumulator expands to push the valve. In such a setup, the Master Control Station (MCS), gets operated topside and transmits electrical signals through umbilical to the Subsea Electronic Modules (SEM)s [31].

MEHCS has been proven as a reliable system, and has several advantages over its predecessors. However, it contains challenges that are both immediately-, and potentially solved by the all-electric system. Spring sizes are depth- dependent, which means that the actuators have to be made accordingly. Using an electric motor, the actuators will be more universal as the electric power input can be adjusted to account for depth. This issue was addressed by Imle in [22]. Another weakness in the MEHCS system is that there are only two positions that can be truly verified: open and closed. Verification can be done by monitoring pressure across the valve, together with the level in the hydraulic line. However, the accuracy of reading the hydraulic line is not always accurate, and the response time depends on depth [31].

Also, there is a safety issue concerning the potential failure of the spring. Even though it is rare, as they are replaced periodically, the spring could potentially break. The spring can be somewhat monitored by the activation pressure in the hydraulic line. This pressure can then be compared with a baseline- pressure set in advance. Such tests are however only conducted during closing of the valve. In an electric actuated system, there will be possible to perform partial strokes to verify mobility of the valve. Also, swift and accurate position adjustments and measurements from a rotating motor will be possible [31].

2.4 All-Electric Control Systems

The common denominator of all-electric control systems is the HPU and hence hydraulic lines, pumps, DCVs, distribution systems, and accumulators are no longer necessary. This will result in significant cost savings. A diverse group of subsea suppliers and operators, including Baker Hughes, have, according to Offshore magazine [6], demonstrated several analyses showing how a transition to all- electric systems reduces costs. It is also reported that the savings will depend on the field layout and system solutions, but the industry consensus appears to be between 10-25% for Capital Expenditures, CAPEX. It will also result in a significant reduction of Operating Expenses OPEX.

The typical benefits of all-electric systems are also presented:

- Carbon footprint and costs are reduced by removing the HPU system.
- The umbilical will be smaller in size and less complex, reducing costs and simplifying installation.
- The SCM becomes smaller and lighter, and the cost of hardware and installation is reduced.
- Health, Safety and Environment, HSE, is improved and testing is simplified, as pressurized equipment is eliminated.
- Hydraulic leaks are no longer a problem.
- Further benefits can be realized if the data made available through electric systems is utilized. [6]

The opportunities associated with new available data are numerous. For gate valves, one of the main differences with the all-electric systems is that continuous data such as position, power consumption, and rpm are made available. Using such data, the following indirect benefits of such systems are suggested:

- The state of the system is always known.
- Trust in the system's ability to function is approved.
- Data- driven approach to condition- based maintenance.
- One can estimate the lifespan of the component and perform predictive maintenance.
- New design and test philosophies are made available.
- Digital and automation strategies and processes are facilitated [6].

Although the potential is great, the system must have a carefully thought-out infrastructure that meets HSE requirements. This has long been a major challenge for suppliers. Requirements for redundancy have resulted in numerous solutions among suppliers. An example of such a requirement is the redundant mechanism for closing barrier valves. MEHCS solved this by using a spring, and for a long time, this was also the preferred solution for all-electric systems. The spring mechanism is a solution known to work, and there was probably some skepticism involved for the completely new fail- safe function. However, by a thorough establishment of a system configuration and testing, the battery system has for many become the main solution. A proposed setup for valve- control, as presented by Carsten Mahler [24], is to use a switch module in the SEMs. The normal function of the module is to forward current from the battery to its own actuator. In addition, each switch module is connected to a second actuator as shown in figure 3. In the event of a failure in one SEM, another redundant system will step in to take control. This is done through an established safety logic and actuation system controller which communicates with both valves. Ultimately, this establishes a second layer of safety, as the battery is a redundant system in itself [31].

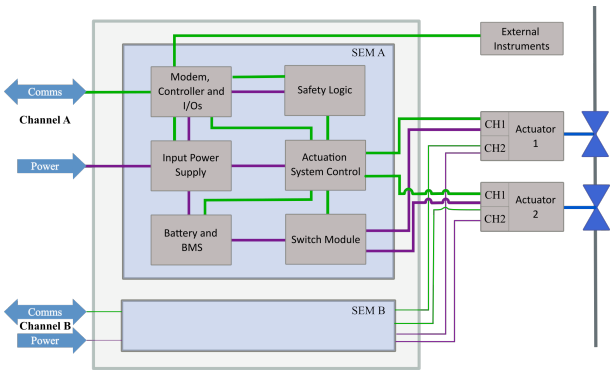

Figure 3: Architecture of an all-electric system [24]

2.5 Proposed all- electric system for X-mas tree valves

A proposal for an all-electric system setup with a battery solution can be seen in figure 4. For this system a load cell to measure linear force on the stem is included. For systems described by the service companies [2], the calculation of required actuation force is solely based on the input electrical power. The electric power input changes linearly with the required linear force. Hence, a so- called valve fingerprint can be established. A valve fingerprint typically includes either the required linear force- or torque to operate the valve. This is then plotted versus position of the valve. However, by introducing a load cell, one could establish whether a potential increase in power consumption occurs from degradation in the actuator or the valve interface.

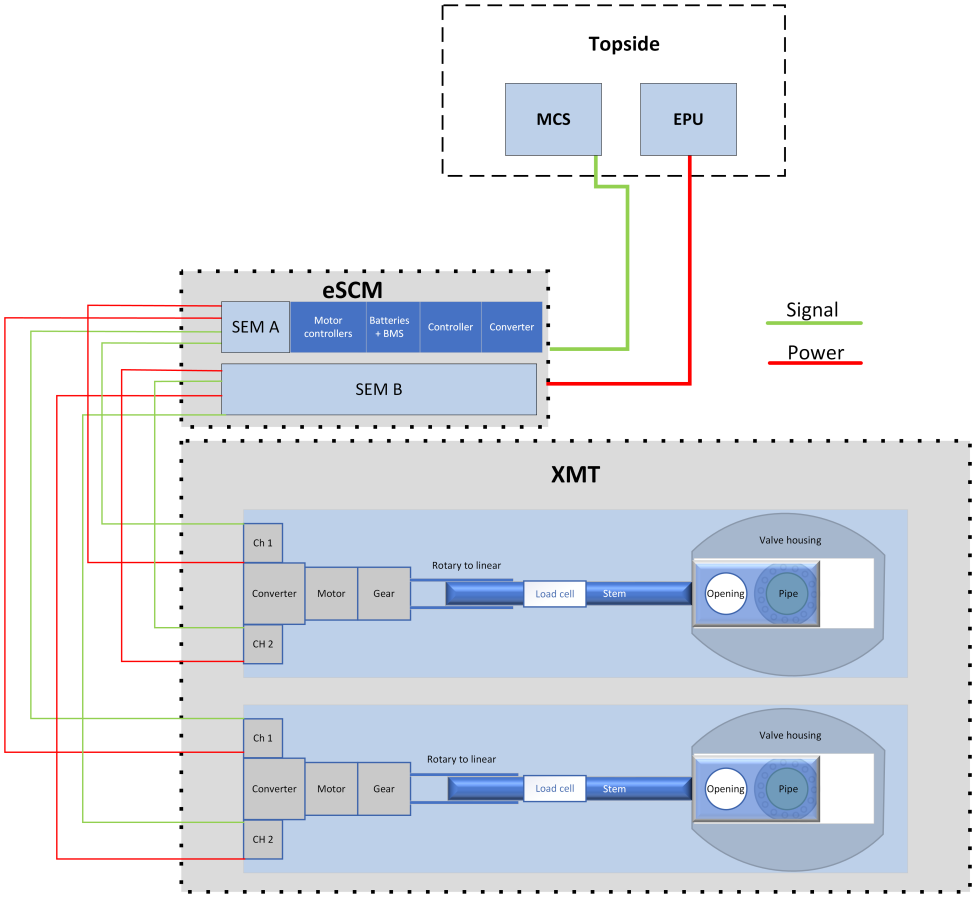

Figure 4: Schematic of all-electric system, self-produced in [31] with inspiration from [20] and [24].

Following is a description of the components from topside to valve interface:

- The proposed system includes a power supply in the EPU to send High Voltage Direct Current, HVDC down to the electric Subsea Control Module, eSCE. In the eSCE, the current is converted to usable DC current through a converter.
- Data- and signal communication from the MCS will be through fiber optic communication. This results in faster response times, which in turn will provide better performance.
- Logic in terms of electrical and signal controlling inside the eSCE will be assumed to be similar to the architecture in figure 3. Batteries are used for backup power, and a switch module is also included for redundancy. The latter explains the two channels of electrical power and signal.
- The motor used for the described system is an Alternating Current, AC, motor. Therefore, it is necessary to convert from DC to AC, hence the converter in figure 4. Theory about the AC motors can be seen in section 2.13.

- A gear box lowers the motor rpm in exchange for a higher torque. There is only a fixed gear. Theory about motor and gear output calculations can be seen in section 2.13.3.
- Conversion from rotary to linear movement is done by a lead screw to drive the valve stem. Theory about the lead screws can be seen in section 2.15.
- Including a load cell as part of the stem will provide readings for linear force. Theory about load cells can be seen in section 2.14.
- The executive part of the system is the interface of an unbalanced gate valve. Theory about valves and calculation of forces can be seen in section 2.10 and 2.12.

2.6 X-mas tree

As mentioned, the X-mas is a complex component. The tree has its own control system for regulating its many valves. The valves are used for regulating, servicing, choking and testing the production flow [8]. By utilizing most commonly hydraulic power from umbilicals, the valves can be controlled as wanted. Multiple sensors are also used for monitoring important factors such as temperature and pressure. The sensors provide data that is sent topside for analysis. The main functions of a X-mas tree can according to Bai and Bai[8] be presented as follows:

- Work as a link between well and flow-line. This is applicable for both production and injection wells.
- Collect data from the well through sensors. This can be sensors such as P/T sensors, flow-meters and acoustic sensors.
- Act as a part of the barrier element, i.e. be able to withstand reservoir pressure and act as a barrier between the flow and the environment. Common practice for well integrity is to have two levels of barriers. During production, parts of the X-mas tree acts as the secondary barrier.

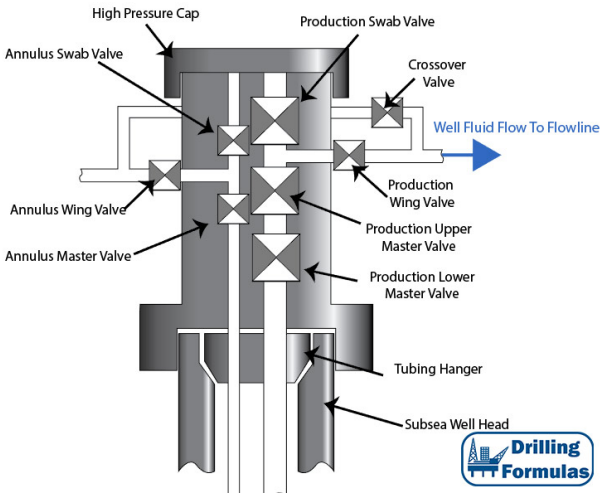

Figure 5: Schematic of vertical X-mas tree [1]

The schematic provided in figure 5 displays some of the common valves, and how they are configured in a vertical X-mas tree. However, this example is equipped with two master valves, which is uncommon. The down-hole safety valve or DHSV, is placed further down in the production tubing. The DHSV is a pure fail-safe function that is designed to shut of the well stream when under specific failure conditions such as excessive pressure build-up or leakage. [19]. The DHSV has a unique design to ensure that it stays close even if power is lost. The design can be described as a flapper that rotates from upstream to downstream before it is locked in place. This means that the well-stream ends up pushing on the valve towards closed position. The DHSV acts as the primary barrier of the well.

Moving upwards to the X-mas tree itself, one will find most of the remaining valves of the well. The production master valve, or PMV, is a valve placed in the X-mas tree across the line of production flow. This valve is typically kept open for production, but acts as a secondary barrier element (capable of resisting full reservoir pressure). The production wing valve, or PWV, is also kept open during production, whereas the production and annulus swab valves, PSV and ASV, are closed [19]. The annulus master valve and annulus access valve, AMV and AAV, are two of many valves connected to the various annuluses (spacings between the tubing, casings or formation). Bai and Bai [8] states that AMV and AAV are "used to equalize the pressure between the upper space and lower space of the tubing hanger during the normal production. When well intervention is performed, access to the well-bore and annulus are provided by the Swab Valve (PSV) and the Annulus Swab valve (ASV)".

To be able to bleed potential excess pressure in the annulus, or bleed of during a valve test, a crossover valve, XOV, is used. This valve adds communication between the production line and the annulus. If the bore experiences excess pressure, the production line can be closed to bleed of annulus pressure to the production line [19]. The XOV is also used during leak test of various valves, by regulating pressure in the test volume.

2.7 Choke valves

There are several different types of choke valves. A common characteristic is that they all have the task of regulating flow by adjusting the opening of the choke. For subsea systems, the angle choke type is most commonly used. This means that the valve body has a right-angle shape where the choke is operated perpendicular to the flow. An illustration of a typical choke valve can be seen in Figure 6.

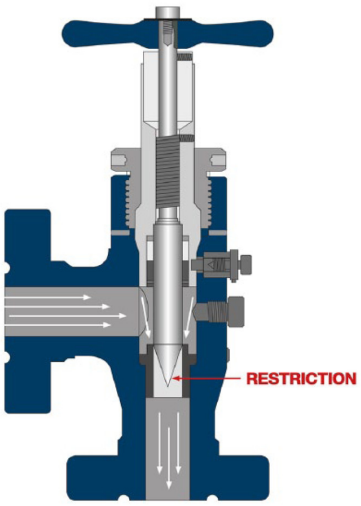

Figure 6: Needle & Seat choke [34]

The part of the choke that interacts with the flow is called the trim. There are several different trims for various needs. Examples include needle & seat and plug & cage. A common problem for many chokes is erosion. Sand production is one of the main causes of erosion. As a result, the material selection for the trim is important. The most common is to use Tungsten Carbide [30], but even with this, erosion is a common problem. Chokes have several important tasks as presented by A. Rubio, H. Abu Zeid, and J. H. Meyer[29]: to protect valves from high differential pressures during well startup and shutdown, balance pressures from different wells to join in a common manifold, reduce pressure in the flow-line, protect against reservoir collapse during startup, and control the flow rate to minimize sand and water production. These are several good reasons for why the choke should function as best as possible.

Opening and closing of the choke is done by rotating the stem. In MEHCS, hydraulic power is used for this. Therefore, it must be converted from linear force from hydraulics to rotational force to drive the choke. This is done by a stepping actuator. The stepping actuator needs two hydraulic control valves to forward hydraulic power to one of the two cylinders for either opening or closing. The conversion to rotation occurs by one of the cylinders being pumped back and forth while driving a ratchet wheel that rotates the stem. An interval from fully open to completely closed requires between 200-250 steps on the ratchet wheel. This is also how the position of the choke is monitored [29]. An illustration of the stepping actuator can be seen in Figure 7.

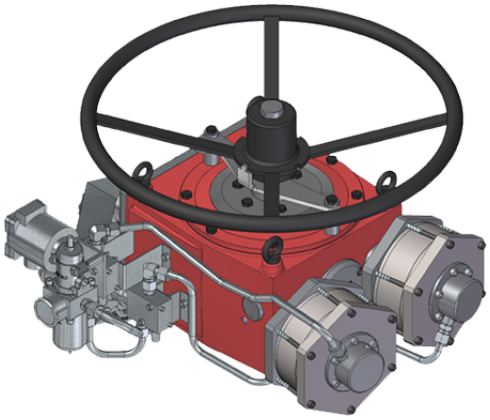

Figure 7: Illustration of a stepping actuator with two cylinders[32].

2.8 Electric chokes

The implementation of electric chokes can prove to be a huge step for the evolution of subsea systems. Several approving factors has been made by A. Rubio, H. Abu Zeid, and J. H. Meyer in [29]. Many of the same perks applies as for the gate valve. This including the removal of hydraulics that is connected to both economical and HSE concerns. Speed control, the hydro-static effect and simplicity are additional perks mentioned for the electric choke. Similar solutions can be applied to the choke as for the gate valve. By replacing the stepping actuator with an electrical motor, continuous monitoring of both position, electrical power input and rpm can be done. Power input can be monitored and compared to the differential pressure. Also, the much approved response time will allow for more accurate monitoring and decision making for ultra deep waters, which is becoming more common.

2.9 The flow coefficient

The flow coefficient Cv is a critical factor to consider in fluid dynamics and control systems. It quantifies the rate at which a fluid can flow through a valve or other control device. Simply defined, the flow coefficient Cv is the number of US gallons per minute of water that can pass through a given valve at a pressure drop of 1 psi at room temperature[30]. For chokes the initial Cv value is provided by the manufacturer. The CV can be defined by:

$$Cv = 1.167 * \frac{Q}{\sqrt{\frac{\Delta P}{\rho}}} \tag{1}$$

Where Q is the flowrate in $\frac{m^3}{h}$. ΔP is the pressure drop over the choke in bar. ρ is the specific gravity of the medium in $\frac{kg}{m^3}$.

The theoretical Cv given by the manufacturers as the optimal Cv-value. As degradation occurs by common erosion from sand, there will be a deviation from the optimal value. This way one will get an idea of the condition of the choke. Analysis of data from Cv can be also useful in correlation with other preventive factors. For instance, the Cv value is subject to the Reynolds number, which is a dimensionless quantity that helps predict flow patterns in different fluid flow situations. At lower Reynolds numbers, the flow is laminar, meaning it moves smoothly and predictably. At higher Reynolds numbers, the flow becomes turbulent, with vortexes and eddies.[11]. This can greatly affect the performance of a valve or control device, changing the effective Cv value. Thus, it's crucial to keep in mind that Cv values can change based on the flow conditions in the system. Factors such as turbulence, but also the impact velocity and ultimately angle of impact for particles should be addressed for a potential correlation. The issues concerning angle of impact is reviewed in[30]. Cutting and deformation are addressed as the two types of erosion mechanisms. For angles as low as 0-40 degrees, hard particles can make cuts on on the choke as the shear stress exceeds the shear strength of the choke. while at high angles of 30-90, erosion is caused by delamination and micro cracking[30]. The electric chokes ability to send out continuous positioning data can theoretically give more accurate estimations of Cv than for hydraulic chokes. This as the position of the stepping actuator has a certain inaccuracy.

2.10 Gate valves

Gate valves have the main objective of stopping flow. These make out most of the valves in the X- mas tree, and serves important tasks such as being barrier elements. An illustration of a gate valve can be seen in figure 8. The figure presents a typical gate valve for MEHCS, which is hydraulically actuated with a spring as fail- safe solution. For the solution in figure 8, the system has a rising stem which runs through the entire actuator. A cylinder is fed with hydraulics to push the piston towards the open position. The spring is then contracted as it gets pushed on by the piston. The piston also pushes on the stem and ultimately the slab gate to open the valve. In addition there system includes gate seat and stem seals to properly make a seal between the slab gate, valve body and actuator.

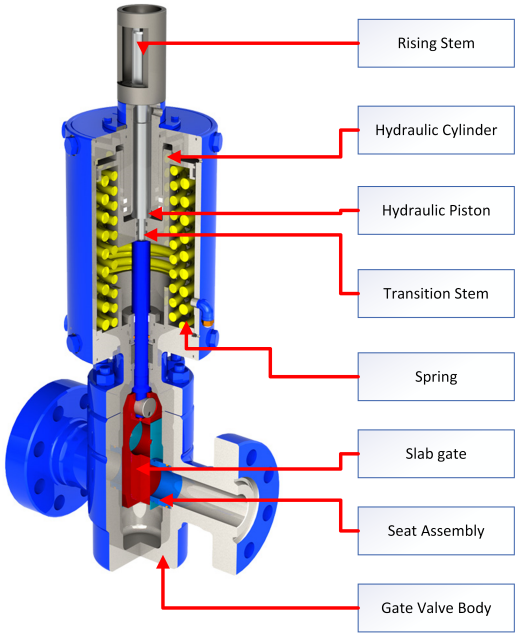

Figure 8: Illustration of a hydraulic gate valve with actuator, adapted from [38]

Gate valves, as opposed to most of the choke valves, are rarely operated. Many of the gate valves' use is reserved for testing, interventions, routing and other minor tasks. In result, their health has to be established through periodical testing. As previously mentioned, some of these valves has to be able to resist full reservoir pressure if there is an emergency. This means that the condition of the valves are of great importance. A procedure for testing of the production cross will e given in section 2.11.

2.10.1 Balanced and unbalanced gate valves

Two common types of stem- and valve body designs are unbalanced and balanced. For balanced systems, a lower stem is included on the opposite side of the gate. This is typically open to the ambient pressure of the sea, which pushes on the stem in closed direction. For unbalanced valves, the valve housing is closed to the sea on the lower part of the gate, but a connection to the well- flow provides support for closing. An illustration of an unbalance gate valve can be seen in figure 9. Balanced designs are typically used for great depths where the hydro- static pressure from the sea provides sufficient force for closing the valve. For this thesis, and further analysis, the unbalance design will be considered.

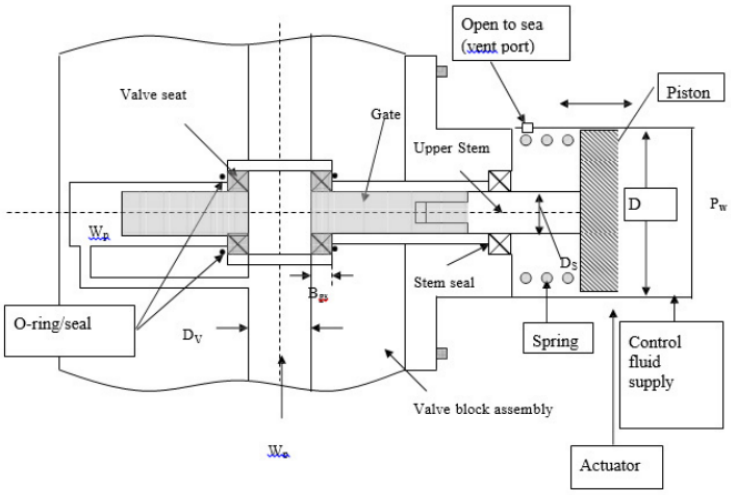

Figure 9: Schematic of an unbalanced gate valve: open position[9].

2.11 Procedure for valve testing

The procedure of leakage testing valves in the X-mas tree is required regularly. Standards such as NORSOK D-010 and API RP 14B has guidelines for how such tests should be conducted. Common practice is to initially close the Production Choke Valve, PCV. This is to stop production, and start stabilizing a Wellhead Shut In Pressure, WHSIP. The service line is then utilized to bleed of pressure in the test volume. The objective is to monitor potential change in the differential pressure over time. It is stated by API RP 14B that the differential pressure should be 70 bar, with a minimum of 30 bar. During the test, of typically 10 minutes for each valve, the gate experiences full WHSIP from below while a significantly lower pressure is helping from above [19].

A test procedure was presented by Hanne Fuglestand [19] in collaboration with Equinor. The procedure for testing of the production cross is as follows:

- Stop any possible chemical injection to the well.
- Shut in the well by closing PCV and PWV.
- Wait approximately 15 minutes for the well to stabilize before the annular shut-in pressure is recorded together with the shut-in tubing pressure and the shut-in well temperature.
- Close PMV.
- Verify that AWV is closed and inject MEG/methanol through the service line in order to equalize the pressure across XOV. This must be done with a certain over-pressure in the service line.

- Open XOV and flush the production cross with MEG/methanol supplied by the service line. Continue to pump MEG/methanol into the production cross until the pressure in the cross reaches a differential pressure of 70 bar across PWV.
- Close XOV and bled off the pressure above XOV by opening AWV. The pressure
- Can also be bled off to the service line. Bled off until sufficient differential pressure is achieved, ideally 70 bar with a minimum of 30 bar.
- Close MEGIV
- Monitor the pressure in the production cross for 10 minutes and document the results.

An illustration of the tree during the procedure is given in figure 10:

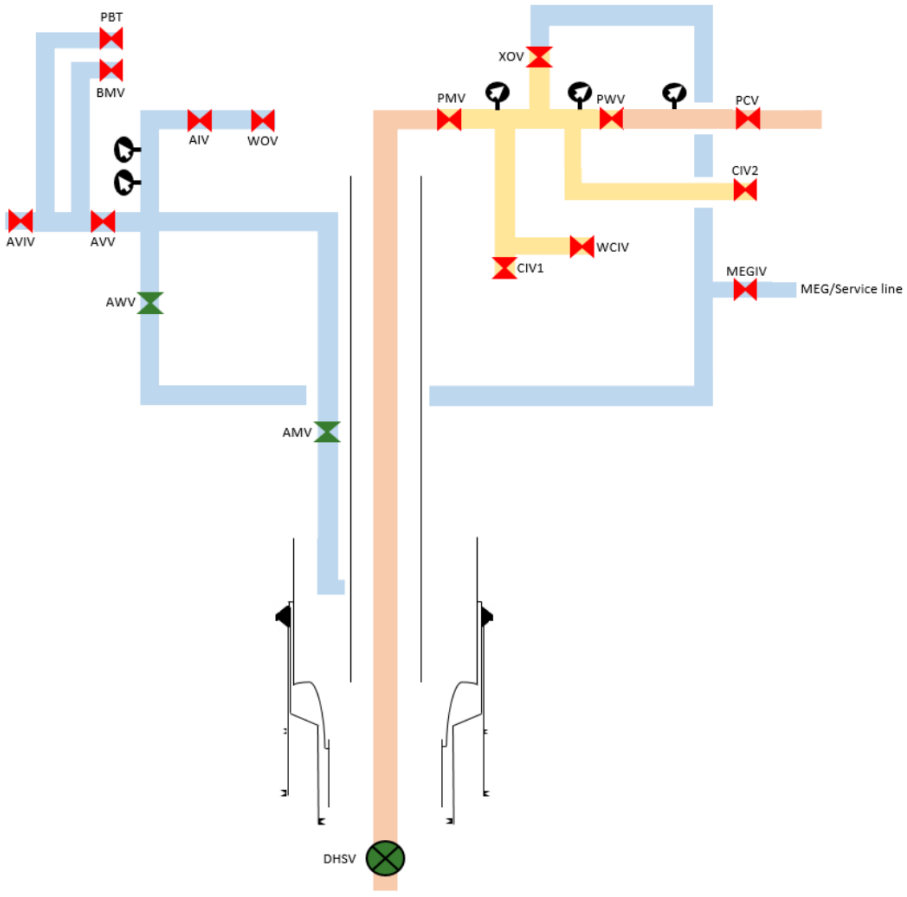

Figure 10: Test procedure of the production cross [19].

Where:

- AIV → Annulus Isolation Valve
- AMV → Annulus Master Valve
- AVIV → Annulus Vent Isolation Valve
- AVV → Annulus Vent Valve
- AWV → Annulus Wing Valve
- BMV → Bleed Monitoring Valve
- CIV → Chemical Injection Valve
- DHSV → Down Hole Safety Valve
- MEGIV → MEG Isolation Valve
- PBT → Plug Bleed and Test Valve
- PCV → Production Choke Valve
- PMV → Production Master Valve
- PWV → Production Wing Valve
- WCIV → Workover Chemical Injection Valve
- WOV → Workover Valve
- XOV → Crossover Valve

For a successful test, the differential pressure profile will gradually increase as the downstream pressure is bled off and then remain constant for the 10 minutes test period. The pressure across the valve should be equalized before opening a valve after completed test [2].

2.12 Calculations for unbalanced gate valve

The analysis of forces acting on a gate valve, and ultimately the stem, during its operation requires the understanding of both the mechanical and fluid dynamics involved. The actuator force is required to counter several factors: the pressure exerted on the valve's mobile part, the friction between the seat and the gate, the stem seal friction, and the spring force. Notably, the friction between the gate and the seat reaches its peak when the gate is in a closed position, which is when the force between the seats and the gate is assumed to be maximum. The forces acting can be categorized into two, being the friction forces and the forces directly from water- and well pressure.

2.12.1 Local forces

As the gate both changes direction and position through its cycle, the forces acting varies for each state. This section presents the individual forces that are present during the whole cycle.

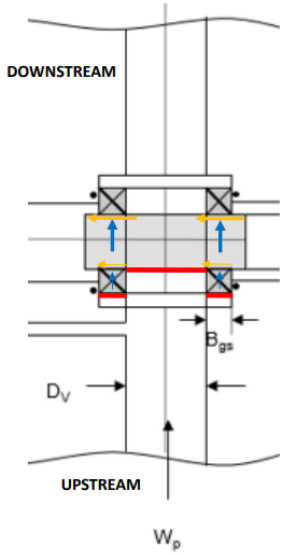

Figure 11: Loads and friction on gate seats – closed valve [9].

One friction force acting is the gate seat friction, which acts in the crack open/close states. In these states, the valve is always closed as the stroke is larger than the nominal size. Figure 11 illustrates areas that the well pressure is acting on, as well as reaction and friction forces. The red lines represents the areas that are exposed to $WHSIP$, blue arrows represent the reaction forces, and orange arrows represents the friction forces on the gate seats. In this case, the valve is moving to the right. The friction forces will move in opposite directions for open/close. The friction forces can be calculated as follows:

- F_{gsup} = Friction force for upstream gate seat
- F_{gsdo} = Friction force for downstream gate seat
- $F_{gsup} = \mu * WHSIP * A_{gs}$
- $F_{gsdo} = \mu * WHSIP * (A_g + A_{gs})$
- Total friction force from gate seats: $F_1 = WHSIP * \mu * (2 * A_{gs} + A_g)$

Where $WHSIP$ is the wellhead shut-in pressure, μ is the friction coefficient of the seat and gate, A_{gs} is the gate seat area and A_g is the valve bore area.

The second friction force acting is the stem seal friction acting between the stem seal and the stem:

- Stem seal friction: $F_2 = F_f$

Where F_f is the stem seal friction. The stem seal friction is initially a value experimentally tested in a workshop prior to installment.

Forces applied by fluid pressure contribute significantly to the gate valve's operation. These forces arise due to the pressure of the fluid inside the control system, well shut-in pressure, and the hydrostatic water pressure. The water pressure is introduced through the vent port seen in figure 9. The resulting linear force that is acting on the system pushes on the piston towards closed position. The area exposed is the piston area minus the stem area:

- Force due to water pressure: $F_3 = P_w * (A_p - A_s)$

Where P_w is the hydrostatic water pressure, A_p is the piston area and A_s is the stem area.

The unbalanced valve also includes a force resulting from the well pressure pushing on the head of the gate (see W_p in figure 9). This force will always work in closing direction. Area pushed on is assumed to be equal to the stem area :

- Force due to well pressure: $F_4 = WHSIP * A_s$

Where $WHSIP$ is the well head shut-in pressure and A_s is the stem area.

The spring in the gate valve actuator serves to close the gate when hydraulic pressure is not applied, as a fail-safe function. The spring force changes depending on the gate's position, i.e., whether it's fully closed, cracking open/close, or fully open.

The spring force is at its lowest when the gate is fully closed. For crack and fully open position the spring constant (force/distance) multiplied with position is added:

- Spring force: $F_5 = F_s + k * x$

Where F_s is the base spring force, k is the spring constant and x is the linear position of the valve. x is zero for closed state.

2.12.2 Net Forces

The net forces acting on the stem/piston during different stages of the gate valve operation can be calculated by summing up the relevant forces described above. These stages include moving crack open, run open, run close and crack closed.

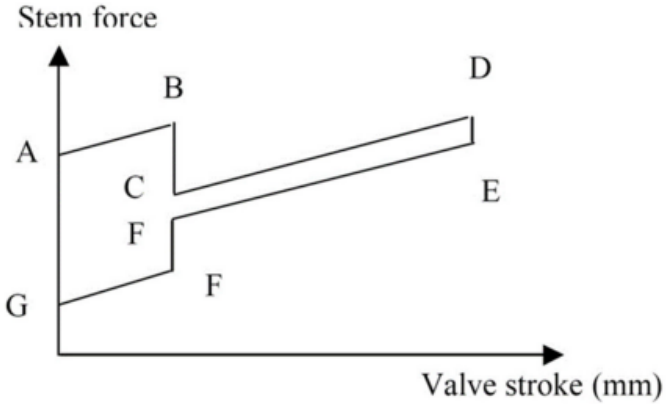

Figure 12: Stem force vs position of gate valve with spring[9].

Figure 12 illustrates the expected stem force required through one cycle of the valve. Following list explains the change in differential pressure for a valve operating in a flowing well as well as the change in position.

- $A \rightarrow B$: Fully closed $\rightarrow$ Gate is moving from fully closed to crack open position.
- $B \rightarrow C$: Pressure equalizing
- $C \rightarrow D$: No differential pressure $\rightarrow$ Gate is moving from crack open to fully open position.
- $D \rightarrow E$: No differential pressure
- $E \rightarrow F$: No differential pressure $\rightarrow$ Gate is moving from fully open to crack closed position.
- $F \rightarrow G$: Build up pressure $\rightarrow$ Gate is moving from crack closed to fully closed position.

As the spring force is dependent on the position of the gate, eight net forces has to be calculated. This assumes that spring force increases linearly with position. For this thesis, the system configuration doesn't include a spring. It is still assumed that the same equations previously mentioned can be used. However, for the calculation of net forces this means that the number of equations can be reduced to four.

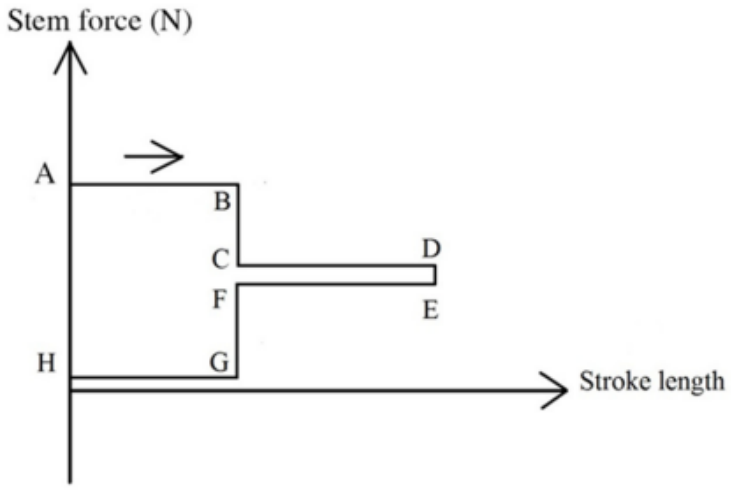

Figure 13: Stem force vs position of gate valve without spring[9].

The same intervals as above applies, but the effect from a spring is removed. The effect is illustrated in figure 13. Net forces can be calculated by combining the local forces as follows:

$$\text{Net force: } A \rightarrow B \text{ :"Crack open"} \rightarrow F_{t1} = F_1 + F_2 + F_3 + F_4 \tag{2}$$

The crack open state requires the most stem force, as all forces are acting against the movement.

$$\text{Net force: } B \rightarrow C \text{ :"Run open"} \rightarrow F_{t1} = F_2 + F_3 + F_4 \tag{3}$$

As the crack is opened, the seat gate friction is removed from the equation.

$$\text{Net force: } C \rightarrow D \text{ :"Run close"} \rightarrow F_{t1} = -F_2 + F_3 + F_4 \tag{4}$$

Closing the valve towards cracked position. The stem seal friction has now changed direction.

$$\text{Net force: } D \rightarrow E \text{ :"Crack close"} \rightarrow F_{t1} = -F_1 - F_2 + F_3 + F_4 \tag{5}$$

Fully closing the valve reuses the seat gate friction, but in opposite direction to earlier.

2.13 Motor and gears

There are two main groupings of electric motors, DC and AC. An AC motor is a type of electric motor that operates on a alternating current (AC), which periodically changes direction. This stands in contrast to a DC motor, which uses direct current (DC) that flows in one direction. For subsea gate valve actuators, the service companies are leaning towards AC motors[3]. Of all AC motor types, the three-phase AC motor is the most commonly used in modern applications[13].

2.13.1 Working Principles of AC Motors

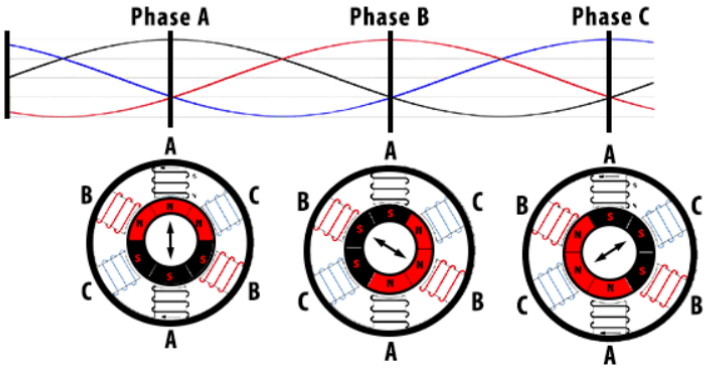

Figure 14: Illustration of principle of AC motor [35].

The fundamental principle behind AC motors is Faraday's law of electromagnetic induction, which states that a change in magnetic field within a closed loop of wire induces an electromotive force in the wire[16].

There are two main components of AC motors: the stators and the rotor. Stators are electromagnetic coils that are energized with alternating current. The rotor is a movable magnet that sits in the middle of the stators. Each pair of stators are positioned on opposite side of each other, and connected to their own AC input. Each pair of stators represents a pair of poles, that alters between north and south as the current is alternating. This causes a rotating magnetic field to rotate the rotor, which is illustrated in figure 14. The stator pairs are here represented by letters A, B and C. The AC input moves in a sinusoidal order that is shifted by a specific phase for each stator pair. For a three phase AC motor with three pairs of stators, this is done by a phase-shift of 120deg. The rotation of the rotor can then be transferred to the shaft for output[13].

2.13.2 Root Mean Square (RMS)

As the current and voltage for each stator of the AC motor alters, a mathematical conversion can be made to make the input values simplified for further calculation. The concept of Root Mean Square (RMS) values is a commonly known method for calculating the performance of AC motors. The RMS value of an alternating current or voltage is a measure of the magnitude of the effective, or "equivalent", DC (direct current) value which would provide the same power to a system. It offers a meaningful way to manage the mathematical complexities of AC quantities, simplifying several analyses[28].

Given an AC waveform, the RMS value is obtained by squaring all the instantaneous values during one cycle, averaging them, and then taking the square root of that average. This can be mathematically represented as follows:

$$V_{\text{rms}} = \frac{V_p}{\sqrt{2}}$$
$$I_{\text{rms}} = \frac{I_p}{\sqrt{2}} \qquad [28]$$

Where V_{rms} and I_{rms} represent the RMS values of voltage and current respectively, V_p is the peak voltage and I_p is the peak current. These formulas applies for an interval for a interval of a whole number of complete sinusoidal cycles.

The importance of using RMS values when dealing with AC quantities becomes particularly apparent when considering power calculations. Power, the rate at which energy is transferred or converted, is calculated by multiplying voltage (V) and current (I). In a DC system, this calculation is straightforward as V and I are constant values. However, in an AC system, both V and I are changing continuously. Here, the RMS values are the effective magnitudes that, when multiplied together, give the actual average power transferred to the load.

As for AC motors, the rms is useful for torque calculations. The torque is a measure of the turning force on an object, and is related to the power output of the motor. For the following section, the current, voltage and ultimately power will be assumed to be their rms values.

2.13.3 Electrical power to torque

When converting from electrical power to torque it is important to adjust for the specific system configuration. For this thesis, the actuator includes a gear box, which has the function of lowering the rpm transferred from the motor to the rest of the system in exchange for more torque. One will then have to adjust for this. As mentioned in the previous section, the current and voltages examined for the case of an AC motor will be assumed to be their rms values. The efficiency of the motor can be described as the motors capability of transforming from electrical to mechanical energy in the forms of power and torque. Then, the gear box has an efficiency that describes the loss of angular momentum.
Considering the assumptions above, one can derive a simplified expression for both motor- and gear box output torque. By considering Watts law in combination with definitions of rotational work, one can conclude on the following formula for motor output torque:

$$\tau_{motor} * \omega = I * V * E_1 \tag{6}$$
$$\tau_{motor} = \frac{I * V * E * 60}{rpm * 2\pi} \tag{7}$$

Where:
E_1 is the efficiency of the motor

τ_{motor} is the output torque.
ω is the angular speed
I is the input current
V is the voltage
As previously mentioned, this is a simplified approach which can be modified for a specific system. Specifically, the efficiency represents all loss of energy as a single variable.
Then, for converting to torque output of the gearbox, the gear ratio is introduced. The gear ratio is simply the relationship between the number of gear teeth on the gears used. The same statement for efficiency from above applies here:

$$G.R. = \frac{t_{out}}{t_{in}} \tag{8}$$

$$\tau_{out} = G.R. * \tau_{motor} * E_2 \tag{9}$$

Where:
$G.R.$ is the gear ratio
t_{out} is the nr. of gear teeth out
t_{in} is the nr. of gear teeth in
E_2 is the efficiency of the gears
τ_{out} is the final torque output

The theory and approach for formulas was found in [10] and [15].

2.14 Load cell

A load cell is a transducer that measures linear force applied to a component, and provides result data in the form of a measurable electrical signal. There are several different types of load cells, but the purpose is mostly the same. For a commonly used strain gauge load cell, strain gauges are placed inside the housing of the load cell. A strain gauge is connected to an electrical circuit, and it then measures the change in resistance corresponding to strain. This is then returned as the electrical signal. According to [36], a load cell usually includes four strain gauge elements. The strain gauges are attached to a stem that deforms when it experiences load. The force is measured by monitoring voltage fluctuation in the component when it undergoes deformation. A load cell for four such strain gauge elements will typically have two of them in tension and two in compression[36]. An illustration of a strain gauge load cell can be seen in figure 15.

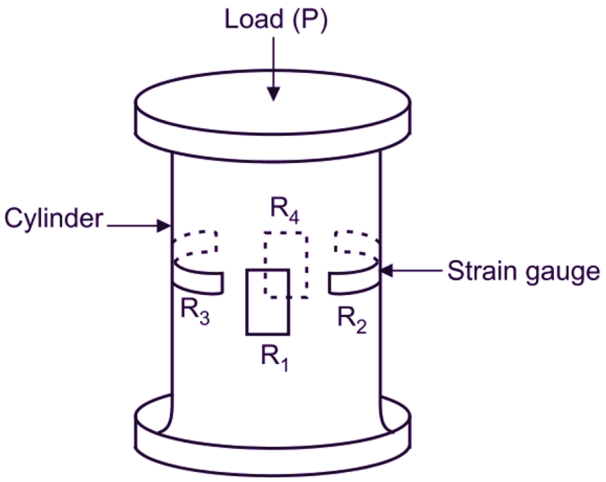

Figure 15: Illustration of a strain gauge load cell [37].

2.15 Rotation to linear movement

For the linear actuator, the next step is to convert from rotation to linear movement. There are several solutions for this, and the service companies of the oil and gas industry has different approaches. The most common parts to achieve this is ACME- or ball screws. Both solutions work by rotating a screw connected to the motor though a lead nut. The lead nut is then driven by the threads of the screw to operate the valve stem.

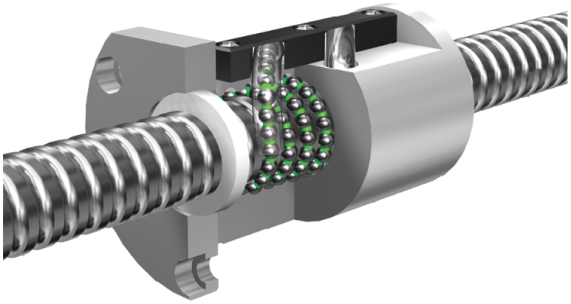

Figure 16: Illustration of a ball screw [4].

Ball screws transform rotary to linear motion using a threaded shaft and a ball nut, enhanced by ball bearings for a rolling, rather than sliding, motion. This typically results in

higher efficiency, load capacity, and accuracy than ACME screws. However, ball screws require a lot of lubrication for efficient operation and longevity. Another issue concerning ball screws are potential "back drive" which means that a linear load on the nut causes the screw to rotate. An illustration of a ball screw can be seen in figure 16[4].

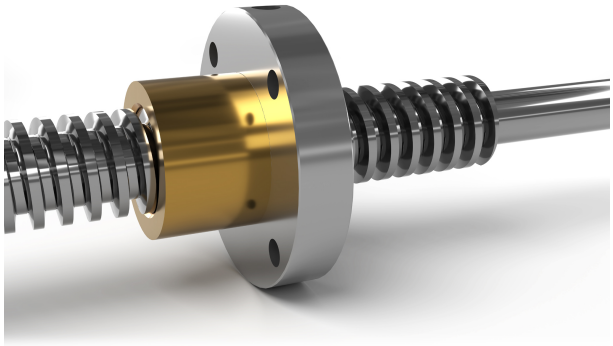

Figure 17: Illustration of an ACME screw [5].

An alternative to the ball screw is lead screws, including the mentioned ACME screw. The "ACME" term refers to the shape of the threads. Lead screws function through direct interaction between the screw shaft and nut threads, without the utilization of ball bearings. This large contact area can lead to more friction and energy loss. Nevertheless, they are frequently used in applications that require resistance to "back-drive". Advances in lead screw design have also reduced some friction issues. An illustration of an ACME screw can be seen in figure 17[27].

For this thesis an ACME screw will be considered as the converter from rotational to linear movement. For a generic lead screw, the output in linear force can be calculated as follows:

$$F = \tau * 2 * \pi * eff/L \tag{10}$$

Where:
F is the output linear force.
τ is the input torque (from gearbox)
eff is the lead screw efficiency (typically 20-80% for ACME screws)
L is the lead in distance/revolution

In the subsea actuator for this thesis, the linear output is assumed to represent the force acting on the valve stem.

2.16 Non newtonian fluids

An additional effect to consider for the above calculations is the type of lubricant for the lead screw. Depending on if it is Newtonian or not, the resulting output force may depend on the RPM used. In fluid mechanics, the behavior of fluids is often characterized by their viscosity, a measure of a fluid's resistance to shear or flow. Two major classes of fluids based on their viscosities are Newtonian and non-Newtonian fluids.

2.16.1 Newtonian Fluids

The simplest type of fluid is a Newtonian fluid, named after Sir Isaac Newton. In a Newtonian fluid, the viscosity remains constant regardless of the applied shear stress. The behavior of Newtonian fluids can be expressed by:

$$\tau = \mu \cdot \dot{\gamma} \tag{11}$$

where τ is the shear stress, μ is the fluid's constant viscosity, and $\dot{\gamma}$ is the rate of shear strain [17].

2.16.2 Non-Newtonian Fluids

Unlike Newtonian fluids, the viscosity of non-Newtonian fluids changes in response to the applied shear stress. There are several types of non-Newtonian fluids, but in general, their behavior does not follow Newton's law of viscosity. For time-independent fluids, they can be classified as shear-thickening (viscosity increases with applied stress) and shear-thinning (viscosity decreases with applied stress).

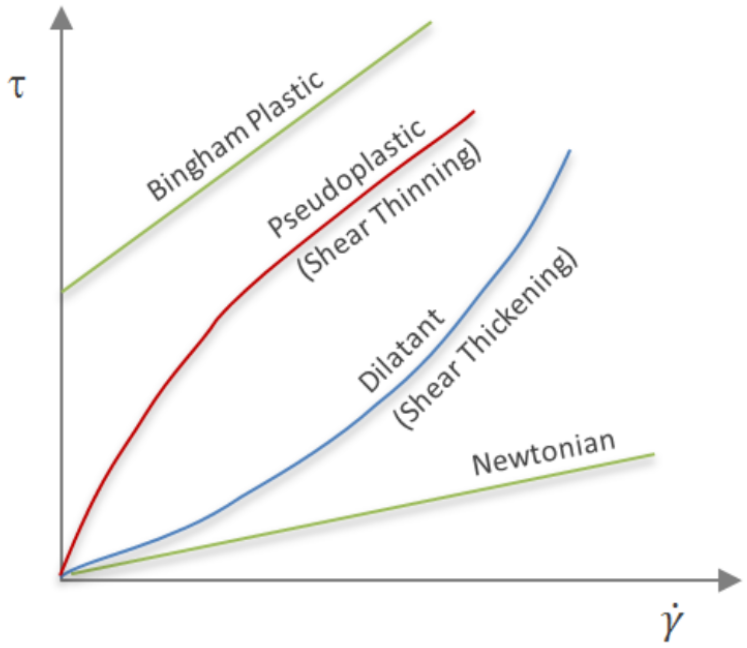

Figure 18: Shear stress vs rate of shear strain for different fluid models [18].

Non-Newtonian fluid behavior is typically described by the power-law model or the Herschel-Bulkley model for yield stress fluids. The curve representing the power-law model, in terms of shear stress vs rate of shear strain, can be seen in figure 18. In the figure, dilatant and pseudoplastic represents two versions of the power-law model- shear thinning and shear thickening. The Herschel-Bulkley model is a combination of the displayed Bingham plastic and the power-law model. A Herschel-Bulkley fluid acts as a solid until the shear stress reaches a specific value called the yield point. What differs from the Bingham plastic is that the Herschel-Bulkley fluid additionally has a shear thickening- or thinning effect. The power-law (or Ostwald-de Waele) model is given by:

$$\tau = K \cdot \dot{\gamma}^{n-1} \tag{12}$$

where τ is the shear stress, K is the flow consistency index, $\dot{\gamma}$ is the rate of shear strain, and n is the flow behavior index. K and n says something about the consistency of the viscosity of the fluid, where K decides the initial angle i.e. viscosity, and n decides the curve. When n is smaller than 1, the fluid is shear thinning. For n larger than 1, the fluid is shear thickening.

For the Herschel-Bulkley model, the equation is:

$$\tau = \tau_0 + K \cdot \dot{\gamma}^n \tag{13}$$

where τ_0 is the yield point, the required shear stress for the fluid to start flowing [18].

As the viscosity of the lubricant inside the actuator might change depending of the shear stress, a properly defined fluid model could be advantageous. As the shear rate increases with RPM of the lead screw, the required force to operate the valve might be non- linear. The described models above are simplified models, that should be adjusted for a real fluid.

3 Maintenance Philosophies

There exists a variety of methods for maintenance of mechanical systems. Which ones to employ depends on numerous factors such as cost, safety, reliability, and the type of component. In the specialization project written for this thesis[31], what can be characterized as the main maintenance methods were presented. This was based on the theory provided by R. Keith Mobley[25]. A summary from this overview can be seen here:

1. Improvement Maintenance
    - Focuses on identifying and implementing improvements to the system. This can include improving reliability, increasing efficiency, or reducing maintenance needs and costs.
2. Corrective Maintenance
    - The philosophy of waiting for system fault to trigger maintenance. This refers to the set of tasks done to pinpoint and resolve a fault in the system to get it to function properly again.
3. Preventive Maintenance
    - Preventive maintenance describes the methods used to prevent failures. It is the most relevant philosophy for this thesis. There are three main categories of preventive maintenance:

    3.1. Reactive Maintenance
        - This is a policy of fixing defects only at certain thresholds. It's not a planned form of maintenance, but often involves data analysis. The main difference from the corrective maintenance is that it acts on the specific data- thresholds and not necessarily component failure.

    3.2. Condition Monitoring
        - This is the process of monitoring a parameter of conditions in system (pressure, power, rpm, etc.) to identify development that can cause a fault. It includes the analysis of trends and correlation between faults.

    3.3. Scheduled Maintenance
        - This approach refers to performing maintenance activities on a predetermined schedule, regardless of the operating condition of the equipment. This could involve activities such as inspections, adjustments, cleaning, or replacements.

3.1 System modelling

A common method for reactive and condition-based monitoring is system modeling. As described in [12], two such types are first-principle and data-driven models. First-principle models are based on theoretical knowledge about the system's physics. This could for instance include equations for the linear force requirement of a gate valve. According to [12], the strength of these models lies in the understanding of the physical processes occurring within the system, hence their ability to predict performance. A typical weakness with this modeling approach is associated with determining underlying parameters, which can also change over time. An example might be the flow consistency index from section 2.16.2. Thus, if a highly accurate model of the system is desired, one has to consider deep theoretical knowledge that considers all these parameters and adjusts them through trial and error[12].

Data-driven models, in their simplest form, are solely based on data analysis. This means that initially, they don't have a foundation for predicting the system's performance. However, given sufficient data access, such a model can construct a good model for the system's input and output using techniques like machine learning. A typical method used for such modeling is regression, which analyses for patterns in the data, and can enable forecasting of the performance. The weaknesses of a purely data-driven model are the need for sufficient data, and the lack of insight into the physical components[12].

3.2 Digital twin

A digital twin can be described as a virtual model of a real system. Essentially, it's a computer-based model that encapsulates the key characteristics and behaviors of its physical counterpart, providing users the capacity to simulate performance and attributes. Digital twins can prove especially important for detecting faults that is difficult to establish from the raw data. Also, for predictive analysis and lifetime expectancy they prove useful. There are several possible approaches when constructing a digital twin. However, a common practise is to apply a first-principle model as a baseline. This is a model that calculates performance based on theoretical formulations. Such a model could essentially serve as a standalone digital twin. However, the numerous possibilities within data-driven models have enabled several improvements.

First-Principle Data-Driven, or FPDD, models are described in [12] and combine the two models. The logic in such a model could be to first establish a base case, then adjust parameters based on historical data. Considering the example from section 3.1, this could be to correct the flow consistency index to a new estimated value based on collected data. This could, for instance, be done if the program manages to recognize a deviation and link it to a specific error. Another example is for predictive analysis where expected input values are provided, and the data-driven part of the model can estimate the development of parameters based on historical data.

The accuracy of a digital twin using FPDD depends on the available historical data. As the gate valve system in section 2.5 lacks such data, the emphasis for further work will in this thesis be on a first-principle model. However, some relevant topics for data-driven models will be presented in the next sections as potential for further work.

3.3 Motor Current Signature Analysis

By replacing the hydraulic piston for an actuator with an electric motor, a much more accurate reading of input force becomes available. By analyzing data for input current in an AC motor, one can detect faults based on deviations from a set baseline. This type of analysis is called Motor Current Signature Analysis, MCSA. The method is presented in [7]. MCSA was introduced as early as the 1970s and provided a cost-effective alternative to other methods such as thermal- and vibration analysis, as expensive sensors could be eliminated.

The method, in principle, involves analyzing the sine wave of the alternating current. Using classification algorithms, patterns on the curve can be recognized, and specific deviations can be linked to specific faults. Some faults are mentioned in [7]:

- Unbalance/ misalignment
- Defective bearings
- Rotor bar damage
- Load issues
- Dynamic eccentricity
- Static eccentricity

These mainly denote faults in the motor itself. However, it will also be possible to create unique classifications for faults further down in the system. These faults will propagate through the system and produce a distinct current signature. The premise for such a method will be that necessary data has been collected to recognize fault patterns. An important factor is also that for a system containing several components, it may become more difficult to distinguish faults from each other. Hence, there will be a need for more accurately defined current signatures. This, in turn, requires more data.

3.4 Machine learning

Machine learning is a category of artificial intelligence that use algorithms and statistical methods to allow computers to perform tasks without explicit programming. It includes

computational models that can learn from data and make following decisions or predictions. Subsets of machine learning includes supervised learning, unsupervised learning, semi-supervised learning, and reinforcement learning[36].

The potential of a machine learning model depends on its ability to learn patterns from data-sets and predict outcomes based on those patterns. This predictive ability can be applied in various fields, including the operation and maintenance of electric actuated gate valves in subsea environments.

Predictive maintenance, made possible through machine learning, is a revolutionary approach to valve maintenance. Traditionally, maintenance has been reactive or scheduled, leading to unexpected failures or unnecessary service interruptions. Predictive maintenance uses continuous monitoring of the component and can involve machine learning algorithms to predict potential failures based on the collected data.

Data sources included for proposed system includes pressure, position, time, cycle counts, electrical power, rpm and force. These parameters can be fed into machine learning models, which learn from the data to predict when a valve is likely to fail or require maintenance. The models can be trained to recognize patterns associated with valve failures.

In a predictive maintenance setup, when the model predicts an impending failure, it triggers an alert, allowing maintenance crews to service the valve before it fails. This method minimizes unplanned downtime and could significantly reduce maintenance costs.

4 Method

The main objective is to construct a Python program that can be used for condition monitoring. In a fully electric system, there are several components and subsystems that are relevant. The chosen system was described in section 2.5. This system includes a gate valve with an associated actuator. More specifically, the intention is to look at the relationship between input electrical power to the motor, and the necessary linear force to operate the valve. Furthermore, it should also be possible to determine which part of the system that develop potential faults.

It was decided to code a proposed digital twin for the system. As described in section 3.2, a digital twin is often set up using a first-principle model as a baseline. Such a model has been constructed with the help of theory from section 2.12. The program aims to uncover the development of degradation over time. To determine more precisely where in the system degradation occurs, the system has been divided into two subsystems which, with their individual sets of equations, calculate two values for expected linear force on the valve stem.

A plot from a valve test done by a X-mas tree manufacturer was provided, which displayed estimated force on the valve stem and position versus time. This was used as a base line for time intervals of valve cycles, the size of the valve, and to verify that the output was reasonable. However, there was a lack of raw data. Hence, it was decided to construct a second program that simulates data. This program was primarily constructed as a necessity due to the lack of real data. With future availability of data, this program can still be used to simulate expected values for the future. By using methods such as regression analysis within data- based modeling, parameters can be adjusted to expected values with respect to time and the way the system is operated. In this way, predictive analysis can be conducted instead of waiting for the development of faults. Examples of where such analysis can be useful are: To estimate the lifespan of a component, or to adjust operating routines to consider the future health of the component.

In this section, the system for condition monitoring of a fully electric gate valve will be presented. This includes the two programs that were coded and their logic. The two programs are given the following names and descriptions:

- **Digital twin**: A program that estimates two expected values for linear force experienced on the valve stem. One value is based on valve interface calculations, and one is based on calculations for the actuator. The program assumes fixed parameters and is used to verify measured/generated data.
- **Simulator**: A program that simulates data for linear force on the valve stem and input electrical power to the motor. Includes assumed development of faults for certain sub- components. The premises for these assumptions can in the future be replaced with expectation based on historical data.

4.1 System description

The system considered was described in section 2.5 and can be seen again in figure 19. It includes one of the valves with an associated actuator and power supply. DC current from the eSCM is converted to AC in the converter for the motor, and supplies the system with electrical power. A gearbox with a fixed gear reduces the rpm from the motor in exchange for increased output torque. An ACME screw will convert rotational motion from the motor shaft to linear motion in the valve stem. The lubricant of the lead screw is considered to be thin, but shear thickening. On the valve stem, there is an integrated strain gauge load cell that reads the applied linear force. The actuator operates an unbalanced gate valve without a spring. It is also assumed that there are sensors for pressure in the flow line, similar to those for the MEHCS.

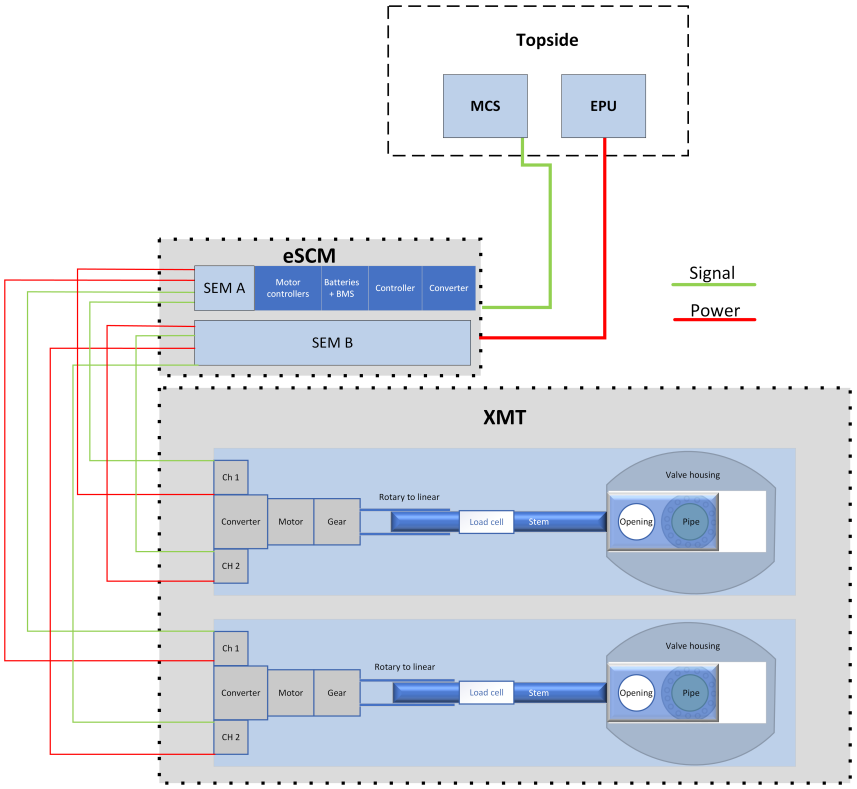

Figure 19: System illustration[31]

4.2 Available sensor data

There exists a wide range of proposed sensor data for such systems. However, some are common. Data for pressure readings across the valve, position, rpm, time, cycle nr. and input power will be considered. In addition, a load cell is proposed to read linear force applied on the valve stem. A similar configuration was proposed in [23] by Dario Leon, Sebastian Imle and Markus Glaser. For their research they implemented a torque transducer on the rotary part of the drivetrain. From conversation with Torkjell Breivik

and Noralf Vedvik at the lab of Petroleumteknisk Senter [3] it was decided that a load cell was simpler to implement. This as it requires less space, and strain gauges can alternatively be placed directly on the stem as a modification to a finished system.

4.3 Input data

Both the Simulator and the Digital twin requires input data. An overview is given below, starting with the fixed input and their assumed starting values for the Simulator:

Variable name	Value	Unit	Description
P_w	2166	psi	Ambient pressure of water.
A_p	43	in^2	Lead nut area (minus stem area). Assumed to be equivalent to piston area in MEHCS.
A_s	7	in^2	Stem area.
A_{vb}	50	in^2	Valve bore area (area of inner pipe).
A_{gs}	28	in^2	Area of gate seats.
E_1	0.9	$[-]$	Motor efficiency.
rpm_{open}	900	$\frac{Rounds}{Minute}$	Motor rpm when opening valve.
rpm_{close}	720	$\frac{Rounds}{Minute}$	Motor rpm when closing valve.
E_2	0.9	$[-]$	Gear box efficiency.
G_R	15	$[-]$	Gear Ratio.
k	0.2	$Pa * s^n$	Flow consistency index.
n	1.5	$[-]$	Flow behaviour index.
μ	0.1	$[-]$	Friction coefficient of seat and gate.
F_f	450	lbf	Stem seal friction.
L	0.02	$\frac{ft}{rev}$	"Lead": Distance traveled by lead nut per revolution.
eff	0.8	$[-]$	Efficiency of lead screw.
dP_{test}	1050	psi	Target differential pressure of leakage test.
$TestType$	Both	$[-]$	Type of test, either "partial" or "normal" for leakage test.

Table 1: Input for Simulator

Input parameters above are listed in the provided Excel file, and should be adjusted for the specific system considered. The two programs collects this data as input.

The values selected for the test run are based on a combination of Tor Berge Gjersviks lecture notes[9], plot provided by the X-mas tree supplier[2], theory discussed in section 2 and assumptions. The subsea system considered is located at a depth of 4870 feet which gives the listed ambient hydrostatic pressure. The areas listed are calculated from a system with lead nut diameter = 8 in, stem diameter = 3 in and gate seat size = 1 in from the example provided by Gjersvik in [9]. The lead nut diameter is here assumed to be equivalent to the piston diameter from the example. The valve bore diameter = 8 in is based on the provided test plot[2]. The friction coefficient for seat and gate, and the stem seal friction is also based on the example from Gjersvik.

The starting efficiencies for motor and gear is assumed to be high at 90%. This is a bold assumption to make. The X-mas tree suppliers have not provided any information about the specific configurations of the system, including these fixed parameters. However, as the scope of the program is to display the opportunities of fault detection, the initial values for these were considered somewhat irrelevant. As for the rpm values, it was loosely based on the ranges in [14]. Considering an assumed common lead of 0.02 ft/rev and the time intervals from the provided plot, the gear ratio was calculated to be 15. The lead screw efficiency is assumed to be in the high end of the interval given in section2.15. The parameters concerning the lubricant are picked to represent a somewhat thin, but shear thickening fluid.

In addition to the simulation of linear force and electrical power, the Simulator generates data for wellhead shut-in pressure and differential pressure across the valve. All data-points also have belonging timestamps, "states" and rpms, which is presented in the Excel file as timeseries. The state is essentially connected to the position and direction of movement for the valve. The shut-in pressure used starts at 1800 psi and is assumed to be linearly decreasing for each cycle of the valve until it reaches the bottom value of 1000 psi at 100 cycles. As one cycle is defined to start at "run close" and end at "stay open", this means that the pressure is dropping from "stay open" to the next "run close". It can be assumed that the system is producing in between these two states, and that this data has been cut out. This could be a realistic way of collecting data as it dedicates memory for the actual run of the valve. The dP is only simulated for the interval of a leakage test and is set to a target value given in table 1.

The input values for the digital twin are given in table 2 below:

Variable name	Unit	Description
Fixed parameters	[−]	The Digital twin reads the same input as the Simulator in table 1, except for $TestType$.
Timeseries	[−]	Parts of the timeseries generated in the Simulator is read. All values are bound to a timestamp. This constitutes the rest of this table.
WHSIP	psi	Wellhead shut-in pressure. New decreased value for each cycle.
Position	mm	Linear displacement of the valve. New value each second.
State	[−]	State of the valve, which was discussed in section 2.12.2. Changes based on position intervals and direction of movement of the valve.
Power	HP	Input electrical power of the motor. Changes for each new "state"-interval.
dP	psi	Differential pressure during leakage test.
RPM	$\frac{Rounds}{Minute}$	Operating RPM of the motor. Depends on whether the valve is closing or opening. Set to zero for "stay open" and "stay closed" states.
Cycle	[−]	Which current cycle the valve is in. Each cycle starts at "run close" and ends on "stay open".

Table 2: Input for Digital twin

4.4 Failure modes

It was made a selection of failures occurring in the system. This selection was based on conversations with supervisor Audun Faanes and co- supervisor Tor Berge Gjersvik. They represent realistic scenarios, and provides deviation in performance that can easily be used for further analysis. The following table 3 displays these failure modes:

Component	Failure mode	Description
Stem-seal	Wear on the seal	Wear results a decrease in stem-seal friction.
Lead screw	Increased friction	Efficiency of the lead screw is lowered. Causing factor may be erosion.
Lubricant	Increase in flow behaviour index	Shear thickening effect is increased. Possible aging.
Gate	Leakage	Differential pressure decreases during test period. Reasons may be seal leakage or erosion on gate.

Table 3: Failure modes considered

4.5 Computational method

The computational method for calculating linear force and power is based on the theory in sections- 2.12 for the valve, 2.13.3 for motor and gear, 2.15 for rotational to linear and 2.16.2 for the effect of the lubricant. As mentioned in section 4, the system can be divided into two sub-systems. These are the rotary and the linear sub-systems. The rotary sub-system involves the motor, gearbox and lead screw, while the linear sub-system considers the valve interface. The connection of the two sub-systems is represented by the stem, and more specifically, the load cell. The equations are used in a slightly different order for the two programs. The following is a description of how the equations are utilized.

The Simulator is based on the wellhead shut-in pressure that is first generated. This is then used as input for several equations to calculate its way through the two sub-systems. Three main steps are used for this:

- By using the current pressure and inputs from table 1, the equations for net force in section 2.12 provide data for required force to operate the valve. These values are also considered the readings of the load cell, and is displayed in Excel.
- The remaining two steps are done in the reversed order of those presented earlier. The objective of these steps is to simulate data for electric power input. First, the a conversion from linear force to torque is done by rearranging the formulas in section 2.15 to solve for torque. Also for this part, the power law model for the lubricant is included. The integration of this model has certain assumptions that will be further discussed in section 4.6. Resulting formula is:
  $\tau_l = F * L/(2 * \pi * eff) * (1 + k * (rpm^{n-1}))$.
  Where τ_l is the calculated torque on the lead screw, F is the linear force on the stem, L is the lead of the lead screw, eff is the lead screw efficiency, k is the flow consistency index and n is the flow behaviour index. The output of this part is the torque on the lead screw.

- The last part is the conversion from torque to input power to the motor. By re-arranging the equations in section 2.13.3, one can first solve for the torque applied between the motor and the gear, and then convert from torque to electrical power. First equation is:
  $\tau_m = \tau_l/(G_R * E_2)$.
  Where τ_m is the calculated torque between motor and gear box, G_R is the gear ratio and E_2 is the efficiency of the gear box. The second equation for calculating power then becomes:
  $Power = (\tau_m * rpm)/(E_1 * 5252)$.
  Where rpm is the current rpm of the motor, E_1 is the motor efficiency and 5252 is a conversion factor for the imperial unit for power, HP.

For the Digital twin, the objective is to utilize the available real or simulated data, and create estimated data assuming no failure. Calculations are here done based on both electrical power input and the wellhead shut-in pressure. This procedure is also done in three steps:

- Input electrical power is used to calculate torque output of the motor by the equations in section 2.13.3. The difference in this program is that all equations are arranged as described. The introduction of the power law model happens in the equation for torque output of the gearbox:
  $\tau_l = G_R * \tau_m * E_2/(1 + k * (rpm^{n-1}))$.
- The torque calculated in the previous step represents the torque provided by the lead screw. Then a conversion to linear force is done by the formulas described in section 2.15. The resulting linear force is the expected output value of the rotary subsystem, assuming none of the faults described in table 3.
- The last part to calculate is the expected force required to operate the valve. This is done by the equations in section 2.12.

In the simulator, the accumulation of faults described in table 3 is simulated. As the belonging parameters are changed, the required force to operate the valve, as well as the performance of the rotary-subsystem will change. By using fixed parameters in the digital twin, one can analyse these deviations.

4.6 Assumptions and premises for failure

Several assumptions were for made for the Simulator, and some for both programs. Following is an overview of these. It is assumed that:

- There is a linear decrease in wellhead shut-in pressure.

- eff, efficiency of the lead screw, is degrading from a random- based function. Assumed erosion.
- n, the flow behaviour index, is increasing from a random- based function. Assumed aging.
- F_f, stem seal friction, is reduced from wear.
- The chance for developing wear, and magnitude of potential degradation increases for large values of differential pressure when opening the valve after a leak test. Common routine is to equalize pressure before opening as described in 2.11. This is however not always performed perfectly. This excess differential pressure is in the code called dP_{random} and is simulated from a random- based function. The analysis of faults depending on dP_{random} was proposed by Børge Pettersen [2]. The proposal of how parameters are affected are personal assumptions.
- The calculations for the required linear force to operate the valve is equivalent to that of a traditional unbalanced valve, excluding the spring.
- The chance of building leakage increases with large values for dP_{random} from previous cycle.
- The formula for a power law model fluid can be applied directly into the torque equations. The shear strain is directly replaced by the RPM. This is not an accurate approach. However, shear strain and rpm are closely related. The main objective when including this model was to obtain a dependency of RPM, and in the form of a power law fluid. This was proposed by Børge Pettersen [2].
- The linear force calculated for "stay open" and "stay closed" state is assumed equal to the forces for "run close" and "crack open". This is based on the plot provided from the X-mas tree supplier, where the values were close to equal. This is probably to prepare for the next state by applying the close to the necessary force for closing and cracking open.

4.7 Algorithm description

The following section describes the logic of the code in more detail. The simulator generates time-series for real data of the load cell readings, and power input. This is then converted to excel. The method used is described in section 4.5. Calculations done depend on which "state" the valve is in within a cycle, i.e. the position and direction. Similarly, the Digital twin produces two estimates for linear force. One is from valve interface equations, using the shut-in pressure, and one is from the rotary sub-system using input power. This is also transferred to the Excel file in a separate file. A basic architecture of the logic can be seen in figure 20. All analysis is based on expected and read linear force on the valve stem. The process in figure 20 that is represented by the load cell sensor, is actual read data simulated by the Simulator. The two remaining are estimations for linear force

done by the Digital twin. The analysis part is done in Excel to check for deviations over time, as well as correlation.

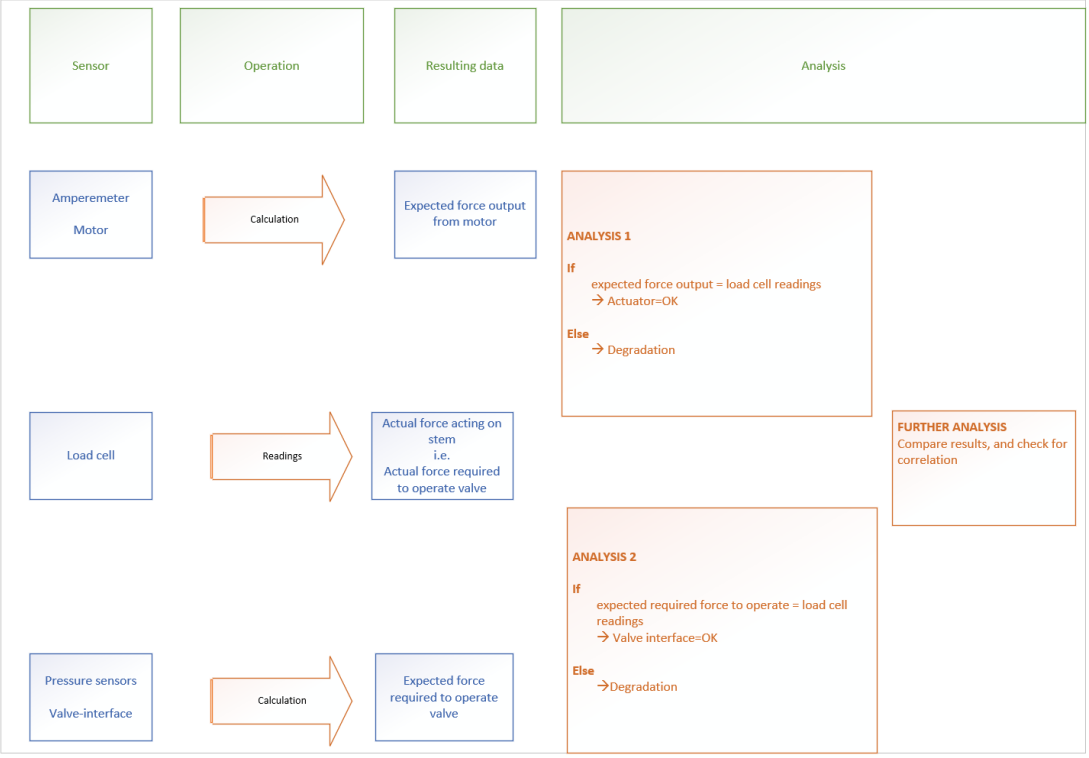

Figure 20: Logic overview

The code itself is for the Simulator represented as a pseudo code in algorithm 1. It generates data for 100 cycles of the valve, where the shut-in pressure "WHSIP" is linearly decreasing for each cycle. Each cycle includes the four net forces described in section 2.12. The first part of the code only considers these cycles, and calculates based on the current WHSIP. The second part of the code which is displayed in the continuing algorithm 1, appoints the calculated data to time-series. The time intervals are based on rpm, Gear Ratio and the lead of the screw, which results in a cycle time of about 280 seconds. This is including "stay open" and "stay closed" time. Also, position intervals is used to appoint each state to a given timestamp. The position intervals are based on the plot provided from the X-mas tree supplier. Included in the code is the degradation of factors discussed in previous chapters. 100 cycles times ca. 280 seconds equals ca. 28000 timestamps.

There is also for the real code included a part for partial stroke testing. This follows the same algorithm, except the simulation of differential pressure. The position- and state intervals are also adjusted for the valve to operate at only 20 percent of the full stroke length. This results in a shorter cycle duration.

Algorithm 1 Simulator

Constants: Read input from the Excel file. Listed in table 1.

Generate Linear Force Required To Operate Valve

Create empty lists for force and torque data.

100 iterations means 100 cycles of the valve:

for 100 iterations **do**
    Generate $WHSIP$, linearly decreasing.
    Generate potential dP_random for when opening the valve, randomly based. Described in section 4.6.
    Generate potential increase in stem seal friction, F_f, based on previous dP_{random}.
    Calculate the local forces present: (F_1, F_2, F_3, F_4).
    Calculate total force required in each state: $(F_{t1}, F_{t2}, F_{t3}, F_{t4})$.
    Above forces are calculated from valve interface equations in section2.12.
    Store to dataframe: df_{force}.
end for

Generate Torque Data For Lead Screw

for 100 iterations **do**
    Generate potential increase in flow behaviour index, n, randomly based.
    Generate potential decrease in lead screw efficiency, eff, randomly based.
    Calculate torque on lead screw for each state $(T_{t1}, T_{t2}, T_{t3}, T_{t4})$.
    Above forces are calculated from lead screw equations in section 2.15.
    Equation for power law is also included from section 2.16.2.
    Store to dataframe: df_{torque}.
end for

Generate Electric Power Data For Motor

Create empty lists for power data.

for 100 iterations **do**
    As lead screw torque is converted to motor output torque through the gear box:
    Calculate motor output torque for each state:
    (tau_motor_t1, tau_motor_t2, tau_motor_t3, tau_motor_t4).
    Calculate power input for each state:
    (power_t1, power_t2, power_t3, power_t4).
    Above forces are calculated from motor and gear equations in sections 2.13.3.
    Store to dictionary: *power_data*.
end for

Algorithm 1 Simulator, Continued

Convert To Timeseries

Set time intervals based on linear speed from rpm, Gear Ratio and ft/rev. This is equal to 280 seconds for the currently used inputs.
Position intervals are based on plot from X-mas tree supplier [3].

for 100 cycles **do**
    for each second in total time (280 seconds) **do**
        An **if**-loop that distinguish between time intervals.
        For each time interval, append a state:
        run_close, crack_close, stay_closed, crack_open, run_open or stay_open.
        Append the current force, power, torque, rpm, position, and dP.

        for State = "stay_closed" **do**
            Simulate leakage test with buildup time of 10s, test time 15s and equalizing time of 10s. Normally the test lasts for 10 min: this is for demonstration.
        end for

        for State = "crack_open" **do**
            Add a potential dP for when opening the valve.
        end for

        Store the data for each second to the dataframe: $df_{\text{timeseries}}$.
        If-loop adds potential leakage in the valve, based on previous dP for "crack_open".
    end for
end for
Save to Excel

Following is the pseudo-code of the digital twin. This code is much simpler, as it assumes that the system is in the condition of the base line. The code generates timeseries for estimated forces by using the Simulators generated pressure and power input. The code has "if" statements that decides what calculations to do based on the state read from the Simulator output.

Algorithm 2 Digital twin

Collect input data

Read timeseries data from the Excel file and store in DataFrame df_{ts}.
Read data from "Input" sheet in the Excel file and store in DataFrame df.
Extract the necessary constants and assign to respective variables.

Main Loop

Create empty lists for storing the final output.
for every timestamp in df_{ts} **do**
    Extract necessary input data.
    Estimate the local forces present: (F_1, F_2, F_3, F_4).
    Utilize **if**- loops to read the state for each timestamp, and do accordingly:
    Do an estimation of required force to operate valve (from dP).
    Do an estimation of actuator output force (from $Power$).
    Append output to the empty lists.
end for
Create a DataFrame for the output lists df_force.
Save to excel.
End

5 Results

In this section, results from running the Simulator and Digital twin will be presented. The example described in section 4.3 starting with a shut-in wellhead pressure of 1800 psi is considered. Inputs are as stated in table 1. Both general fingerprints of the valve, development of individual faults, and assumed correlation between faults will be shown. An example of forecasting will also be demonstrated.

5.1 Excel interface

First, the Excel file will be explained. The Excel file contains four main sheets. These are "Input" for input data, "Data" for Simulator output, "Digital Twin" for Digital twin output, and "Analysis" for further data analysis. The "Input" sheet contains all the initial values that should be filled in by the user prior to the test.

5.1.1 "Data" sheet for Simulator

When the Simulator runs, it generates time-series that are exported to the Excel file "results.xlsx" under the sheet name "Data". A snippet of the first three seconds can be seen in figure 21. In the actual file, data for torque, lead screw efficiency, flow behavior index, and seal friction have also been exported. This additional data is not intended for use in this analysis, but can be useful if additional data-based modeling is introduced to see the development of these parameters. For now, it is included to more easily see the development of errors independent of the analysis that has been done. For the analysis, only the data presented in figure 21 is used, as it represents the actual received data.

Timestamps [s]	Cycle	State	WHSIP [psi]	dP [psi]	Position [mm]	RPM	Linear Force [lbf]	Power [HP]
0	0	run_close	1800	0	-220.00	720	90126	26
1	0	run_close	1800	0	-215.12	720	90126	26
2	0	run_close	1800	0	-210.25	720	90126	26

Figure 21: Timeseries from Simulator displayed in Excel

As the generated data is for 100 cycles, and each cycle lasts for about 280 seconds, it was decided to collect the data in a separate table where each cycle is combined into one. Here, average values for linear force measured for each "state" throughout the cycle have been taken. It is assumed that the force is constant within a given "state" for a cycle. This table can be seen for the first three cycles in figure 22. It is primarily this table that is used for further analysis. In the Excel file, the cycle-table also includes the average electrical force for "crack open", the average differential pressure for the test period, leakage of differential pressure for the test period, change in leakage for the given

cycle, change in electrical power consumption, and cumulative change in electrical power. These are columns created for further analysis.

Cycle	WHSIP [psi]	run_close	crack_close	stay_closed	crack_open	run_open	stay_open
0	1800	90126	70899	110253	110253	91026	90126
1	1791	90073	70943	110083	110083	90953	90073
2	1783	90017	70972	109942	109942	90897	90017

Figure 22: Simulator data displayed for each cycle

5.1.2 "Digital Twin" sheet for Digital twin

As the Digital twin runs, two columns for estimated linear force are generated. This estimation assumes that the input in table 1 remains constant. One column contains estimated actuator output linear force. This is based on the input electrical force from the Simulator and the equations for motor, gearbox and lead screw as described in section 4.5. The other column displays the estimated need for linear force to operate the valve. This is based on the wellhead shut-in pressure and the equations for valve linear force. A snippet of the first three timestamps can be seen in figure 23. All data in the orange columns are Input data retrieved from the Simulator program. There is also a column for torque included, which allows for the potential use of a torque transducer as an alternative to the load cell. This will be further discussed in section 6.

Timestamps [s]	Cycle	State	WHSIP [psi]	dP [psi]	Position [mm]	RPM	Motor Torque [lbf*ft]	Est. Actuator output [lbf]	Est. Valve Req. [lbf]
0	0	run_close	1800	0.00	-220.00	720.00	169	90126	90126
1	0	run_close	1800	0.00	-215.12	720.00	169	90126	90126
2	0	run_close	1800	0.00	-210.25	720.00	169	90126	90126

Figure 23: Timeseries generated by Digital twin

Similar to the "Data" sheet, it was decided to collect the time series into individual cycles in a new table. This has been done for both the estimated actuator output linear force and the estimated need for linear force to operate the valve. A snippet for the first three cycles of estimated actuator output linear force is given in figure 24.

Cycle	WHSIP [psi]	run_close	crack_close	stay_closed	crack_open	run_open	stay_open
0	1800	90126	70899	110253	110253	91026	90126
1	1791	90073	70943	110083	110083	90953	90073
2	1783	90017	70972	109942	109942	90897	90017

Figure 24: Digital twin data for actuator displayed for each cycle

5.1.3 "Analysis" sheet

In the "Analysis" sheet, data analysis is done. This essentially involves comparing the estimated values from the Digital twin with the actual measured value for linear force

generated by the Simulator. The sheet also aims to examine the correlation between errors. Two tables have been created, one for analysis of the actuator, and one for analysis of the valve interface. Both are based on each single cycle, not timestamps.

The table for the actuator contains 4 columns for analysis that can be seen in figure 25. Two of the columns take the ratio $\frac{\text{measured linear force}}{\text{estimated actuator output}}$ for each cycle. This is to investigate the development of efficiency of the rotating subsystem or "actuator". To uncover the potential impact of a shear-thickening lubricant, this calculation is made for two states with different rpm. Respectively "Run Close" and "Run Open". The third column is the difference in efficiency for the two states, showing the effect of shear-thickening. In the last column, the pure difference between measured and estimated linear force is displayed for "Crack Open" state.

Cycle	Actuator efficiency close 720 RPM	Actuator efficiency open 900 RPM	Δ Efficiency actuator (effect of RPM)	Δ Motor output [lbf]
0	100%	100%	0.0%	0.00
1	100%	100%	0.0%	-0.18
2	100%	100%	0.0%	-0.18

Figure 25: Analysis table for actuator

The table for the valve interface analysis contains 5 columns and can be seen in figure 26. The first column contains the difference between measured and estimated linear force required to operate the valve in "Crack Open" state. This develops into a decreasingly negative value as wear on the stem seal, leading to lower friction, is assumed. The next column shows the change in this friction per cycle. Remaining columns are for analysis of the differential pressure during leakage testing. The third column shows the differential pressure that is still present when opening the valve after a completed test. As described in section 2.11, this differential pressure should in principle be equalized to zero, but as described in section 4.6, this is in practice not perfectly implemented. The fourth column contains the increased leakage of differential pressure for each test in each cycle. These are values that are taken from the same table in the "Data" sheet. The fifth column contains the average differential pressure over the test period for each cycle. This starts at the targeted 1050 psi but drops due to simulated leakage.

Cycle	Δ Valve force req. [lbf]	ABSΔ Valve force req. Each cycle [lbf]	dP when opening [psi]	Change in dP from leak [psi]	Average dP during test [psi]
0	0.00	0.0	201.0	0.0	1050.0
1	-10.19	10.2	349.0	0.0	1050.0
2	-10.19	0.0	135.0	1.7	1048.4

Figure 26: Analysis table for valve

5.2 Valve fingerprints

To validate the output of the Simulator, a comparison between generated valve fingerprint and theoretical curve in figure 13 has been made. The fingerprint of the first cycle can be seen in figure 27. This displays that the generated linear force vs position is similar to the theoretical curve.

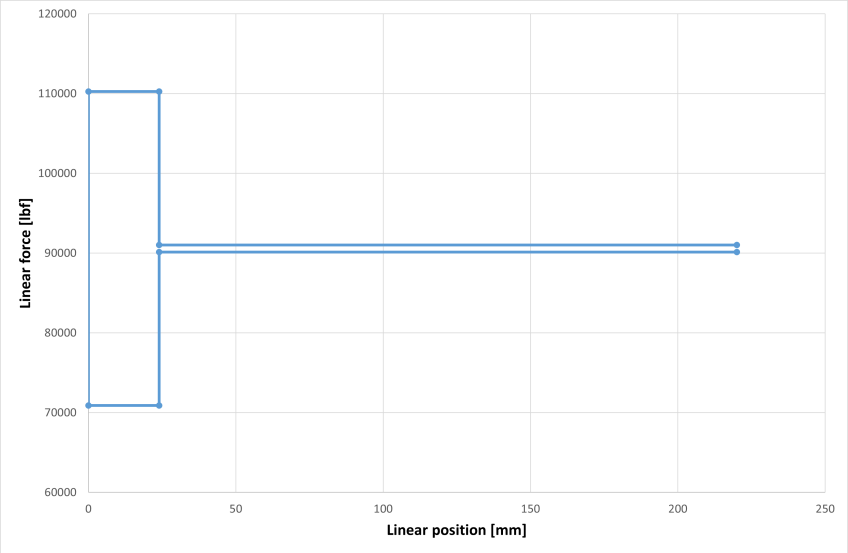

Figure 27: Linear force vs position for the first cycle

It was then of interest to make a comparison to the plot provided by the X-mas tree supplier [2]. This plot is considered classified, and will not be displayed in this thesis. However, figure 28, which illustrates the simulated linear force and position vs time, resembles the same shape as the provided plot. Figure 28 displays the first three cycles of the valve. Here, position equal to zero means that the valve is in closed position. Position -220 represents open position.

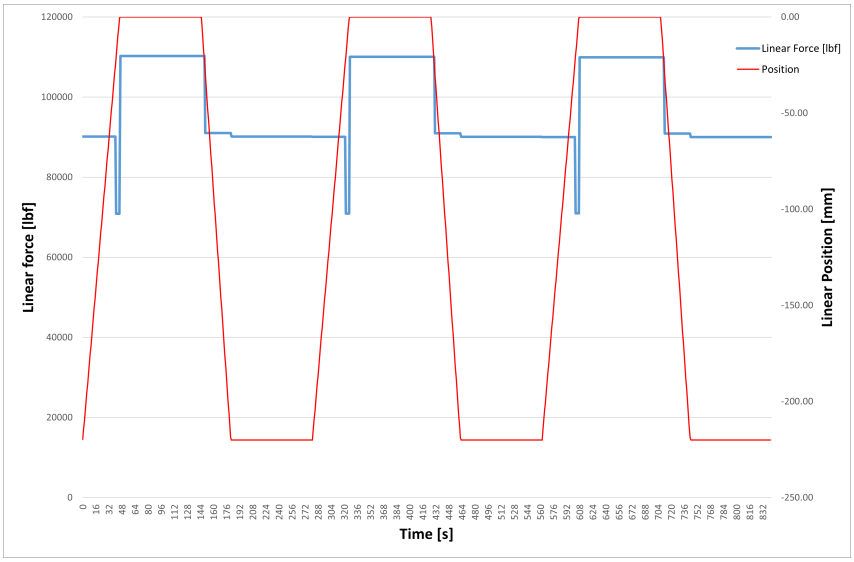

Figure 28: Valve fingerprint for 3 cycles

5.3 Individual Failure modes

The individual failure modes simulated were stated in table 3. This includes the development of wear on the stem seal, which lowers the friction between the seal and the stem. For the investigated case, reduced friction can be seen as the difference between expected and measured linear force required to operate the valve. This can then be plotted vs the cycle number to illustrate the development of such wear. Result is shown in figure 29. A way to use this information could be to set a threshold value for reduced friction that signals the requirement of intervention.

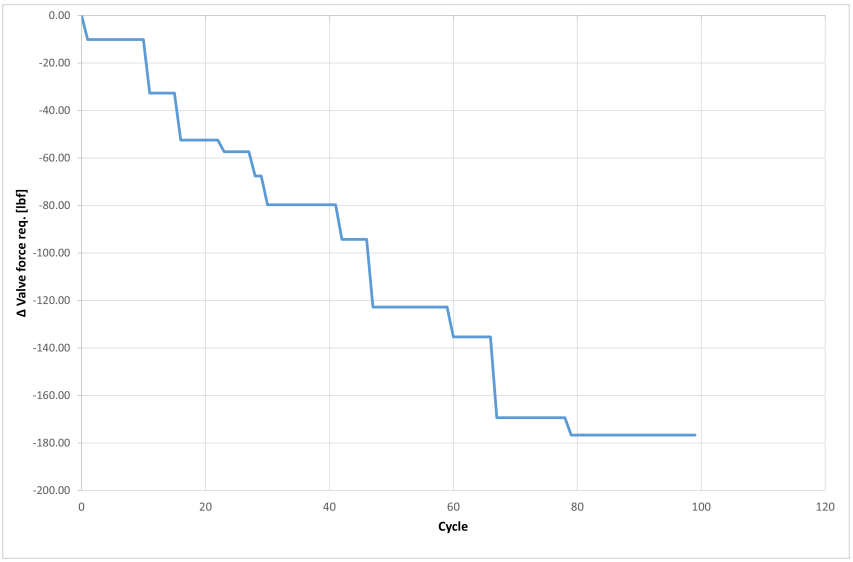

Figure 29: Change in required linear force to operate valve vs cycle

The second and third failure mode in table 3 is related to the actuator in the rotary sub-system. More specifically they describe failures on the lead screw. For the second failure mode, which is increased friction in the lead screw, degradation is simulated by a decrease in lead screw efficiency, "eff". The decrease is randomly applied with a 10% chance each cycle within the interval 5×10^{-7} to 1.5×10^{-6}. The interval was set this low as a small change made a big impact. This big impact is mostly caused by the way the shear thickening effect was introduced in the equations. One way to display the increase in friction in the lead screw is shown in figure 30. Here, the change in actuator performance is the difference between measured data for linear force, and the estimated output linear force. For this example, "Crack Open" state is used, and is plotted against number of cycles. This describes the decrease in the actuators ability to transfer electrical power to linear force on the stem. The reason for the occasional increase in performance is the decrease in shut-in wellhead pressure, and hence the need for linear force.

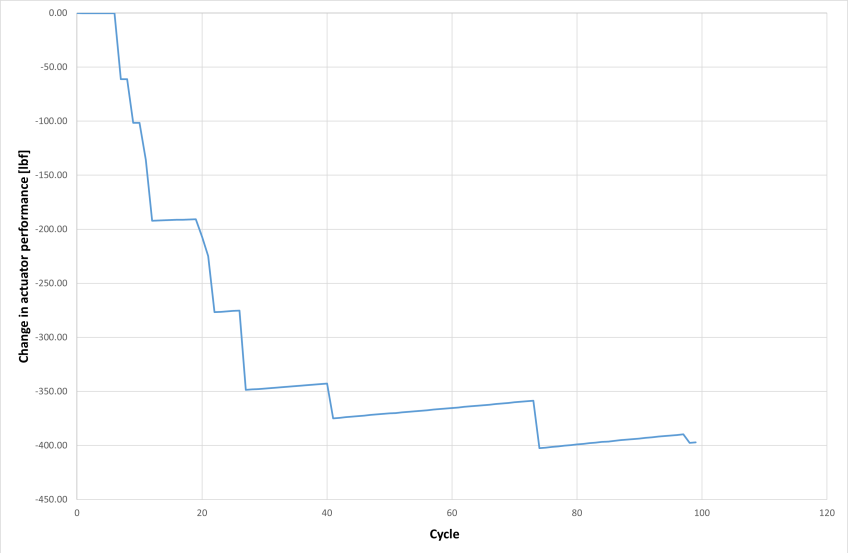

Figure 30: Change actuator performance [lbf] vs cycle for "Crack Open"

However, the decrease in performance displayed in figure 30 cannot be explained solely from the increased friction of the lead screw. As described in section 4, the lubricant for the lead screw is assumed to be shear thickening. The aging of this lubricant is assumed to be increasing the shear thickening effect. This is simulated by increasing the flow behaviour index, n. An increase is done randomly with the chance of 10% to occur each cycle within the interval 5×10^{-6} to 1.5×10^{-4}. The effect of n was shown to be of great significance, and the formula for the shear thickening effect should be adjusted for a real system. For demonstration purposes however, the model used was considered sufficient. To display development of the two failure modes for the lead screw, figure 31 was made. This figure shows the $\frac{\text{measured linear force}}{\text{estimated actuator output}}$, represented as efficiency, vs cycle number. As explained in section 5.1.3, this is done for both opening and closing rpm. Similar to the analysis for the valve interface, a threshold of efficiency can be set for when intervention is necessary.

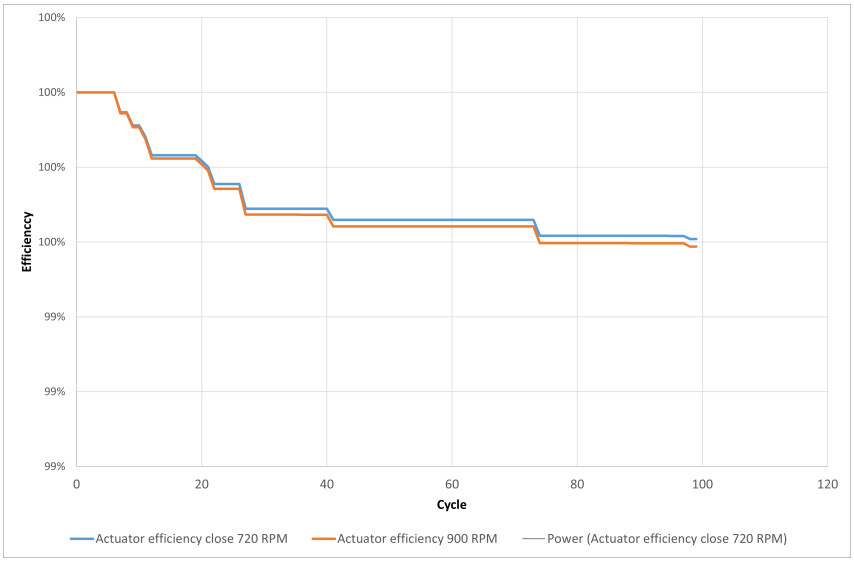

Figure 31: Efficiency of actuator vs cycle

The difference gap between the two graphs in figure 31 represents the effect of change in the flow behaviour index, n. To display this effect in further detail one could take the difference between the two "efficiencies". This is plotted vs cycles in figure 32. Using this information one could make decide whether the right type of lubricant is used, or if it is advantageous to run at a different rpm.

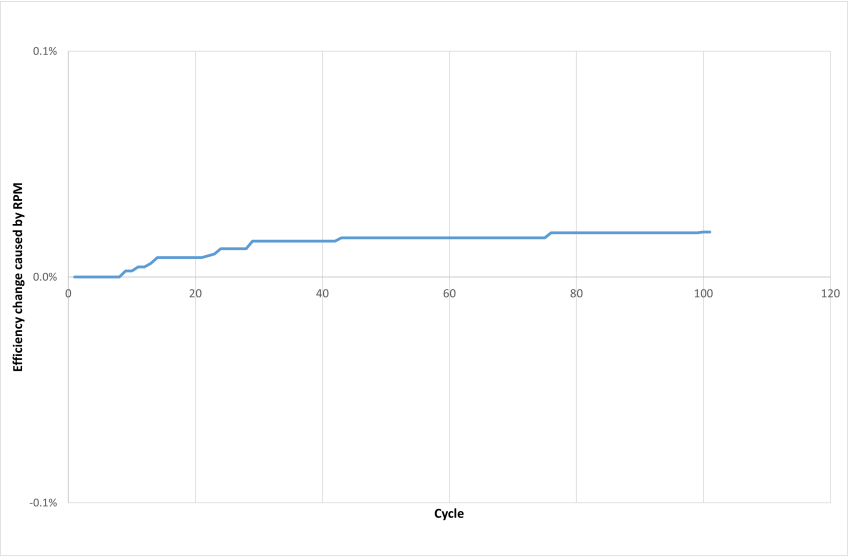

Figure 32: Effect of RPM: displays difference in efficiency between 720 and 900 RPM

The effect on electrical power consumption from failure modes 1, 2 and 3 in table 3 is displayed in the combo chart in figure 33. This chart shows the change in electrical power consumption in "Crack Open" state from cycle to cycle as the black graph. As the wellhead shut-in pressure is constantly decreasing, this graph would be a constant

line at about -0.056 HP if no degradation occurred. Additionally, the difference between expected values and the measured value of linear force is displayed. As seen, when the required linear force to operate the valve is decreased, the change in electrical power spikes below the line of -0.056. This is because required power decreases as there is less friction between the stem and the seal. Similarly the graph spikes above the line when the faults in the actuator is increased. Hence, the faults in the actuator has the opposite effect, as a less efficient lead screw requires more power to deliver the same amount of linear force. Such a chart could be used initially, to get an idea of which part of the system that causes spikes in the electrical power consumption.

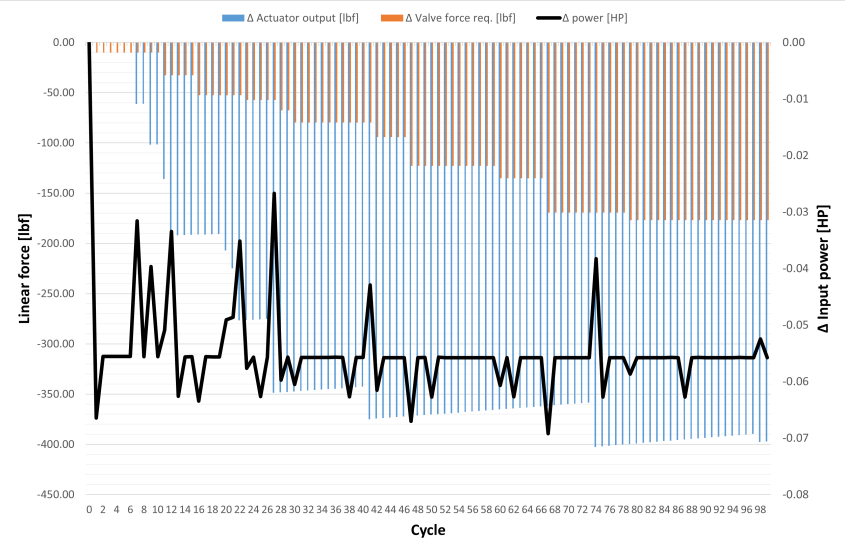

Figure 33: Combo-chart with change in required electrical power vs motor and valve deviation

The final failure mode considered was leakage across the gate valve during test. This failure mode was not intended for analysis with the Digital twin. It was meant to illustrate a correlation between excess dP when opening the valve after a leakage test and the accumulation of leakage. However, an error in the code for Simulator causes the leakage to periodically be reset. This is shown in figure 34, where the average differential pressure during the test period for each cycle is displayed. As the target differential pressure for a test is set to a constant 1050 psi, and leakage should be permanent, the graph was never supposed to increase. However, as the leakage only was intended to be a resulting factor, the error won't affect further results.

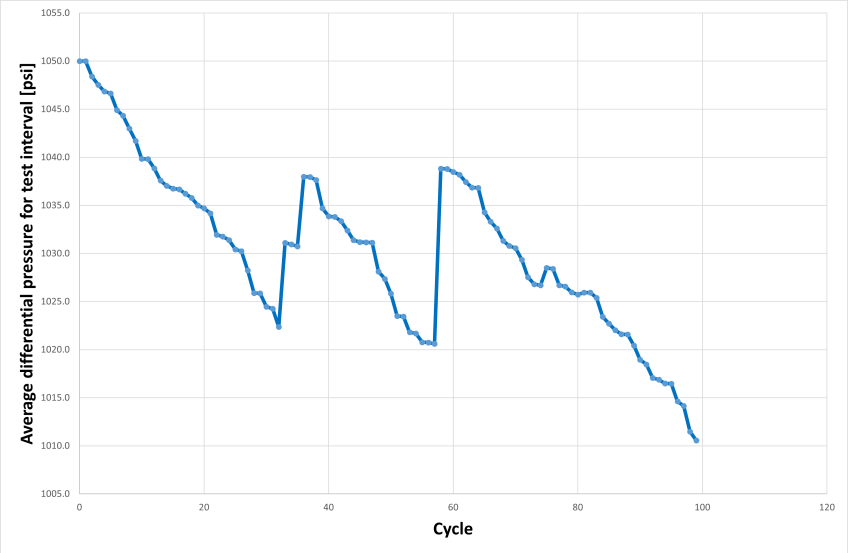

Figure 34: Average differential pressure for test period vs cycle

5.4 Combinations and correlation

The second part of the analysis is to look at correlation between faults. As stated in section 4.6, the "dP_{random}" parameter represents the excess differential pressure when opening the valve after a leakage test. The value of "dP_{random}" is generated randomly for each cycle in the interval of 0 to 435 psi which is about 30 bar. 30 bar is typically the maximum allowed differential pressure when opening the valve [2]. "dP_{random}" is assumed to affect two of the faults simulated: wear of the stem seal, and increased leakage in the valve interface. Higher values for "dP_{random}" will increase both the chance- and magnitude of change for these failure modes. A plot displaying what is meant by "dP_{random}" can be seen in figure 35. Referring to the theory in section 2.11, the test period is typically for 10 minutes. Currently, the Simulator is only generating test intervals of 15 seconds. This is purely to save data, as it is only meant as a demonstration. The build up time and equalizing time is also not for a realistic duration.

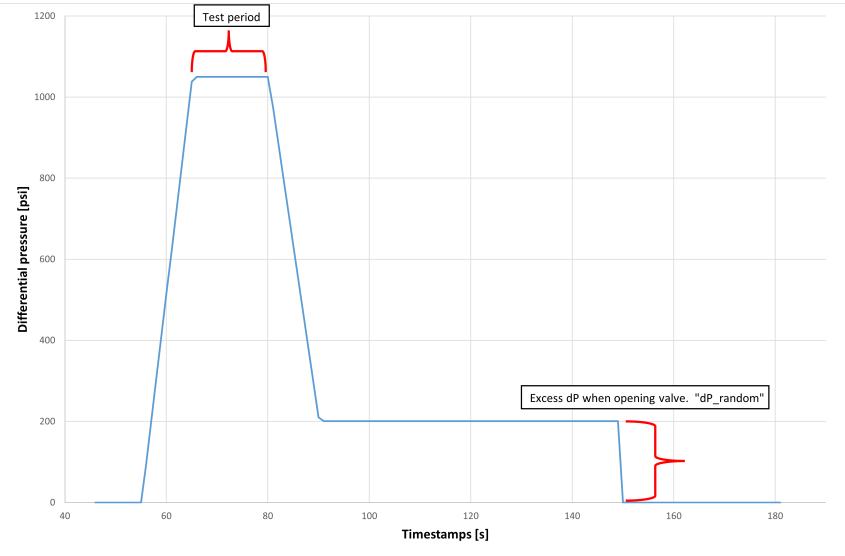

Figure 35: Differential pressure during pressure test vs time [s]

For the wear of the stem seal, which was displayed graphically in figure 29, the potential decrease in friction was set within the interval of 10 to 50 lbf. A function in the code that considers the current value of "dP_{random}" will decide whether a decrease will be in the high- or low end of this interval. Similarly for the chance of developing wear, the chance increases for higher values of "dP_{random}". As there is assumed a correlation between "dP_{random}" and the wear of the stem seal, it is of interest to investigate this further. In figure 36 this correlation is shown by a scatter plot. The plot shows that there are the largest increases of wear on the stem seal happens when "dP_{random}" is large. One could with this information predict potential wear on the stem seal (follow the trend line), get an explanation to why the wear is occurring, and change routines for valve opening to avoid wear. This is of assuming this correlation is valid. The scope of the analysis is to display one of many correlations that may be relevant. When collecting real data, similar relationships may be uncovered.

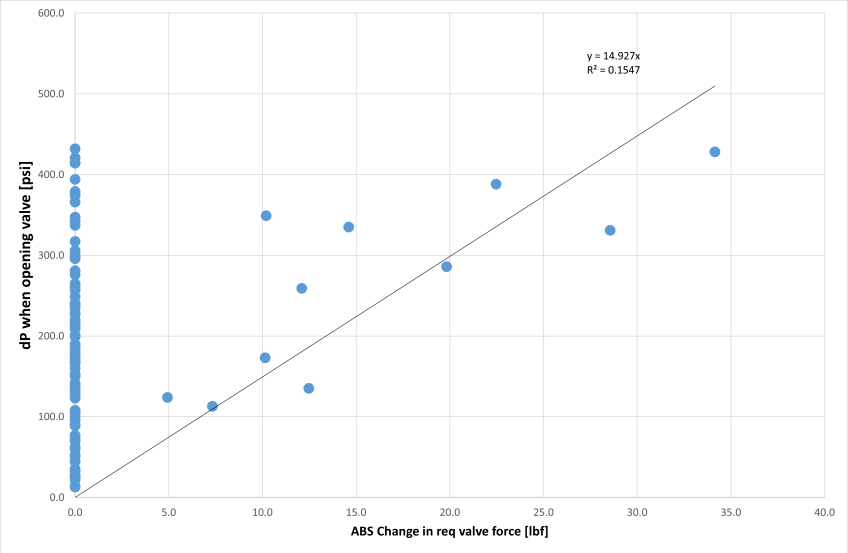

Figure 36: "dP_{random}" vs ABS change in required linear force to operate valve

The second correlation simulated was the increased chance and magnitude for leakage with large values for "dP_{random}". As there was an error in the simulation of differential pressure, mentioned in section 5.3, the correlation between these was not found in the analysis. This could however be a realistic correlation for a real system. The failed scatter plot can be seen in figure 37. The chance and magnitude of leakage was in the code meant to be based on the "dP_{random}" from the previous cycle, as this would potentially cause leakage in the next test. However, as seen, there was no correlation discovered. This is probably because of the way leakage is generated.

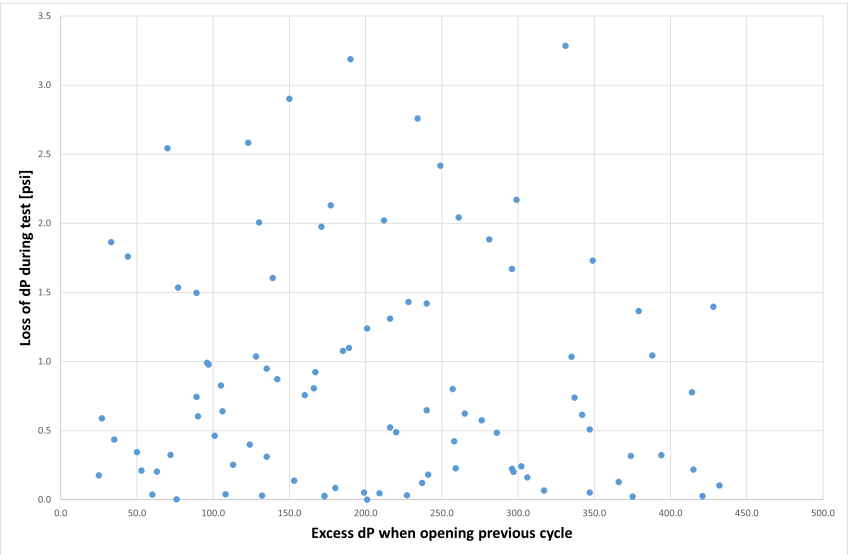

Figure 37: Failed attempt to see correlation in leakage vs opening dP

5.5 Forecasting

A possibility by using such a monitoring system is forecasting. This will be even more relevant when data-based modeling is introduced. One can however make some estimations for the lifetime of components in this setup. Figure 38 displays the same plot as in figure 29, but with a forecasting line by Excel. Certain thresholds could be made to predict useful lifetime, to prepare for intervention. For figure 38, the line ends at a stem seal friction wear of 400 lbf at 200 cycles, which is almost the full initial friction of 450 lbf. One should set a threshold earlier based on negative effects caused by the wear, such as leakage.

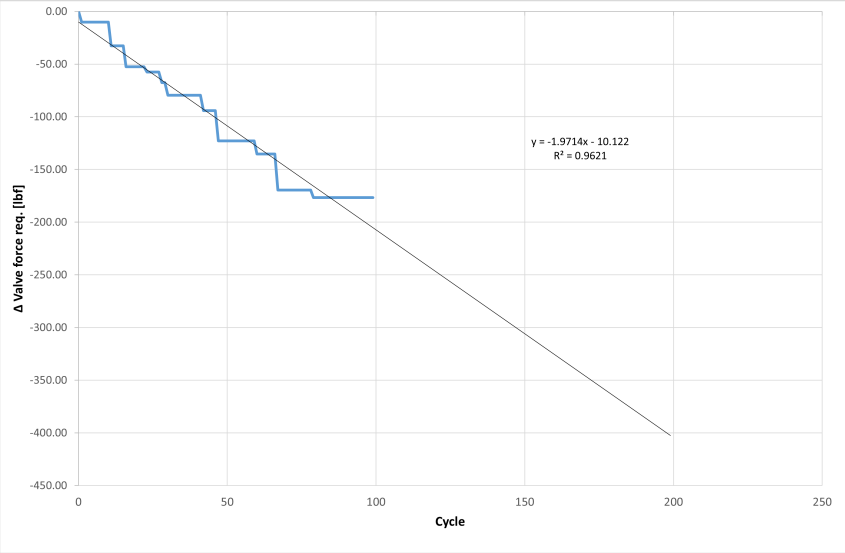

Figure 38: Change in required linear force to operate valve vs cycle, forecast

5.6 Partial stroke testing

The Simulator has also the ability to simulate a partial stroke test. For this test, the valve is only operated 20% of the "Run Close" distance. This is a new possibility for the electric actuated valve. The valve fingerprint of the first 4 cycles for such tests can be seen in figure 39. The same analysis, excluding the leakage test, can be done for partial stroke. This means that the development of wear on the stem seal, and decrease in actuator efficiency can be uncovered the same way as in the normal leakage test scenario. Such testing can be done without significantly hurting the production. This will allow for more frequent testing, so that development of faults can be uncovered much earlier. Reactive maintenance can then be executed prior to potential need for corrective maintenance (described in section 3).

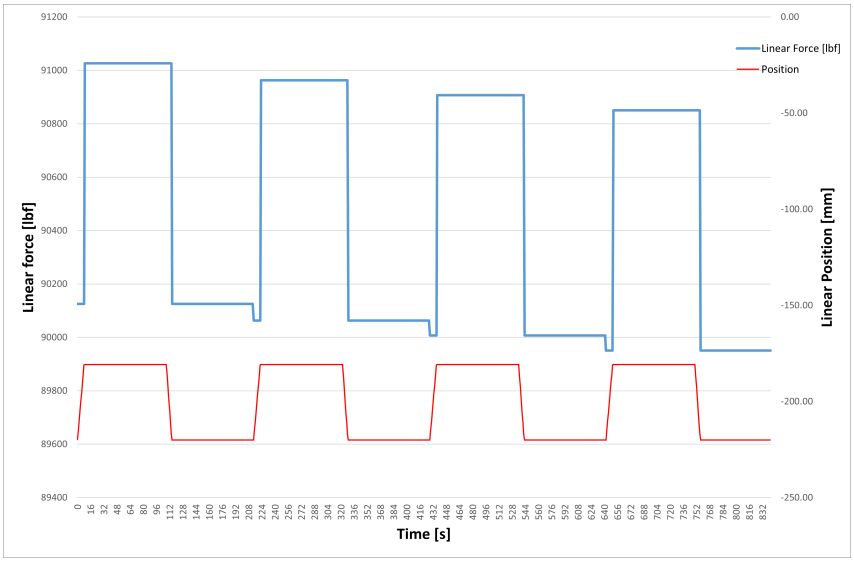

Figure 39: Valve fingerprint of partial stroke test

6 Discussion

In this section, the implications of the results from section 5 will be further discussed. Condition monitoring solutions for partial stroke testing, a system without a load cell, electric chokes, and battery cells will also be discussed briefly.

6.1 Leakage test of X-mas tree valves

The results from section 5 indicate that an all-electric system opens up new possibilities for condition monitoring of gate valves in the X-mas tree. By placing a load cell on the valve stem, individual analysis of the valve alone, or of the actuator, can be performed. Precise monitoring of linear force requirement, position, and rpm are important keywords. Assumptions have been made for the Simulator when generating data. For a real system, several of these assumptions will not be valid, but the Digital twin should still be capable of collecting and analyzing data. The difference will then be that the analysis in Excel must be performed differently.

An immediate improvement that can be made is to create a more realistic model for the shear effect of the lubricant. It would then be useful to know exactly which lubricant is being used. The shear thickening effect used in the code turned out to be too significant. Only small changes in flow behavior index led to large changes in the performance of the actuator. Another error that should be corrected is concerning the generation of leakage data. Any increased leakage should be permanent unless intervention is performed.

Perhaps the greatest potential lies in implementing a data-based model for both the Simulator and Digital twin. With such a model, the Simulator can simulate data based on historical data. This avoids the need for drastic assumptions, and simulations for all historical failure modes can be done. The same logic can still be used, but more comparisons can be made. Further detail can also be given to each individual parameter. Regardless, an overall condition analysis can still be made as demonstrated.

6.2 Required action

In the analysis for section 5, it was determined which subsystem contributes most to the change in electrical power requirement. This can be seen in figure 33. It is clear that the failure modes concerning the lead screw contribute more to the change in power requirement than wear in the stem seal. This is indicated by the blue area being larger than the orange. Although this alone should not determine whether a component needs to be replaced, it can contribute to decision-making in combination with defined thresholds for development in figure 29 and figure 31. A significant advantage of being able to determine which of the two subsystems that needs maintenance is that the type of intervention can be tailored. If only the actuator reaches a threshold value for maintenance needs, one

can avoid having to pull the entire X-mas tree, and instead just replace the actuator part that is externally on the tree. An important point to make is that regardless of the assumptions made for the analysis in this thesis, the programs should be able to detect a decrease in the two sub-systems individually. With a number of additional failure modes for a real system, the additional challenge is to pinpoint the exact components that are degrading within a sub-system.

6.3 Partial stroke

For the analysis in 5, primarily a leakage test has been considered. Only a valve fingerprint was shown for the partial stroke test. As mentioned, many of the same analyses can be performed for this type of test. Especially concerning the mobility of the valve. Partial stroke testing can help uncover faults in the mobility of the valve without stop in production. This allows for tests to be performed more frequently without significant cost cuts from production loss. The total cost can potentially be reduced by uncovering faults early. For partial stroke, erosion will also be a topic, as the valve is open while there is flow in the pipe. Therefore, it may be interesting for a future program to introduce similar monitoring for the flow coefficient, Cv, as is done for choke valves. This was described in section 2.9. By creating a program for monitoring Cv for the gate valve, potential damage to the gate, and thus leakage could be detected. However, this will not uncover leakage associated with other factors than the gate itself. Seals, etc. must be leakage tested as usual.

6.4 Torque transducer

In section 4.2, a system for an electrically actuated gate valve by Dario Leon, Sebastian Imle, and Markus Glaser [23] was mentioned. This system uses a torque transducer instead of a load cell. This was also the system that was first considered for this thesis. The first versions of the codes simulated torque data on the lead screw for comparison. The difference is that the subsystems were defined differently. For this system, the motor and gearbox acted as one subsystem, while the lead screw was considered part of the valve system. A system with a torque transducer could also be relevant, depending on which parts of the system it is desired to distinguish between.

6.5 Excluding the load cell

The possibilities for condition monitoring will increase rapidly when sufficient historical data for the all-electric systems has been collected. The example made for this assignment, using mainly first-principle modeling in the Simulator and Digital twin, is intended as a proposal for immediate application. In the future, it will be possible to exclude the load

cell, but still be able to distinguish faults from each other. One possible method for data-based modeling is the Motor Current Signature Analysis, which was presented in section 3.3. With this method, the sine curve of the input current is examined, and faults can be recognized using historical data. Specific signatures for the curve must be set to recognize the faults.

6.6 Monitoring of electric chokes

Another potential mentioned in section 2.8 is monitoring of electric choke valves. Electric chokes replace the hydraulic stepping actuator with an electric motor. As stated, this leads to a number of advantages related to the accuracy of position measurements, response time, speed control, and the hydrostatic effect, which is a weakness of hydraulic systems. As it is possible to measure position, rpm, pressure, power requirement and possibly torque requirement, a program with almost identical logic as the proposed program for gate valves can developed. In addition, it will still be useful to monitor Cv for the choke, but this can be done with greater accuracy as the position measurements are continuous and not based on a stepping function.

6.7 Monitoring of battery cells

The described system for all-electric gate valves uses battery cells as a fail-safe mechanism. For battery cells, condition monitoring can be performed to a much greater extent than for the traditional spring. The possibilities for condition monitoring were briefly described in the specialization project in preparation for this thesis [31]. It mainly involves monitoring of SOC, which is State Of Charge, and SOH, which is State Of Health. SOC describes how charged the battery is at a given time, and SOH describes the battery's capacity for charging relative to what it is rated for. Monitoring of charging capacity, charging cycles, and aging over time can then be used to describe the condition of the battery. Here, data-based modeling will also be useful to recognize characteristics of failure modes, and estimate lifespan.

7 Conclusions and recommendations for future work

The objective of this thesis has been to investigate the possibilities of and construct a condition monitoring system for all-electric subsea systems. The main outcome is two programs for monitoring of electric actuated gate valves: one Simulator and one Digital twin. Further possibilities were also considered
It was discovered that:

- All- electric systems enable several new techniques within condition monitoring.
- A first-principle based model can be used to construct two programs: one for simulating data that includes failure modes, and one for estimating data assuming no failure modes. A comparison of output can unveil the development of component failure. Following failure modes could be monitored by the program for certain assumptions:
    1. Development of wear on the stem seal.
    2. Increased friction for the lead screw.
    3. Increased shear thickening effect of the lead screw lubricant.
- A correlation between wear on the stem seal and excess pressure when opening a valve after leakage test can be monitored.
- Forecasting of developed failure-modes can be done. Example being the wear on stem seal.
- Analysis of mobility of the valve can be done similarly for partial stroke testing.
- The programs should be able to unveil a decrease in overall performance of the two sub-systems, valve interface and actuator, regardless of assumptions.
- Monitoring of flow coefficient, Cv, could be beneficial for both electric chokes and gate valves during partial stroke test.
- Electric chokes could utilize programs with similar logic as the ones constructed as it has an electric motor.

7.1 Recommendations for future work

Future work includes implementing data-based modeling. This can be both for optimizing parameters and equations in the Simulator for forecasting, or Motor Current Signature Analysis.

Appendix

A Code for Digital Twin

```
import pandas as pd
import math



# Read timeseries from Excel
df_ts = pd.read_excel('results.xlsx', sheet_name='Data')
df = pd.read_excel('results.xlsx', sheet_name='Input')

# Create a dictionary with variable names as keys and their
    corresponding values as values
fixed_values_constants = {row['Variable']: row['Value'] for _, row in df
    .iterrows()}

# Fixed values
A_p = fixed_values_constants['A_p']
A_s = fixed_values_constants['A_s']
A_vb = fixed_values_constants['A_vb']
A_gs = fixed_values_constants['A_gs']
mu = fixed_values_constants['mu']
F_f = fixed_values_constants['F_f']
P_w = fixed_values_constants['P_w']
L = fixed_values_constants['L']
eff = fixed_values_constants['eff']
G_R = fixed_values_constants['G_R']
E_1 = fixed_values_constants['E_1']
E_2 = fixed_values_constants['E_2']
k = fixed_values_constants['k']
n = fixed_values_constants['n']
rpm_open = fixed_values_constants['rpm_open']
rpm_close = fixed_values_constants['rpm_close']
# Create lists to store output
timestamps = []
Actuator_force = []
valve_req = []
states = []
cycle = []
WHSIP_values = []
dP_values = []
positions = []
rpms = []
motor_torque = []


for idx, row in df_ts.iterrows():

    # Uses following input data:
    WHSIP = row['WHSIP [psi]']
    position = row['Position [mm]']
```

```
    state = row['State']
    power = row['Power [HP]']
    dP = row['dP [psi]']
    position = row['Position [mm]']
    rpm_val = row['RPM']

    # Required force estimation for valve states
    F_1 = WHSIP * mu * (2*A_gs + A_vb)
    F_2 = F_f
    F_3 = P_w * (A_p - A_s)
    F_4 = WHSIP * A_s



    # Loops through the state data from the "Data sheet" to check whick
   state to calculate for
    if state == 'run_close':
        F = -F_2 + F_3 + F_4
        tau_motor = power * E_1 * 5252 / (rpm_close)
        tau_gear = G_R * tau_motor * E_2/(1 + k * (rpm_close ** (n-1)))
        force_out = tau_gear*2*math.pi*eff/L

    elif state == 'crack_close':
        F = -F_1 - F_2 + F_3 + F_4
        tau_motor = power * E_1 * 5252 / (rpm_close)
        tau_gear = G_R * tau_motor * E_2/(1 + k * (rpm_close ** (n-1)))
        force_out = tau_gear*2*math.pi*eff/L


    elif state == 'crack_open':
        F = F_1 + F_2 + F_3 + F_4
        tau_motor = power * E_1 * 5252 / (rpm_open)
        tau_gear = G_R * tau_motor * E_2/(1 + k * (rpm_open ** (n-1)))
        force_out = tau_gear*2*math.pi*eff/L

    elif state == 'run_open':
        F = F_2 + F_3 + F_4
        tau_motor = power * E_1 * 5252 / (rpm_open)
        tau_gear = G_R * tau_motor * E_2/(1 + k * (rpm_open** (n-1)))
        force_out = tau_gear*2*math.pi*eff/L

    elif state == 'stay_closed':
        F = F_1 + F_2 + F_3 + F_4
        tau_motor = power * E_1 * 5252 / (rpm_open)
        tau_gear = G_R * tau_motor * E_2/(1 + k * (rpm_open ** (n-1)))
        force_out = tau_gear*2*math.pi*eff/L
    else:
        state = 'stay_open'
        F = -F_2 + F_3 + F_4
        tau_motor = power * E_1 * 5252 / (rpm_close)
        tau_gear = G_R * tau_motor * E_2/(1 + k * (rpm_close ** (n-1)))
        force_out = tau_gear*2*math.pi*eff/L
```

```
102
103
104
105     # Append to output lists
106     timestamps.append(row['Timestamps [s]'])
107     Actuator_force.append(force_out)
108     valve_req.append(F)
109     states.append(state)
110     cycle.append(row['Cycle'])
111     WHSIP_values.append(WHSIP)
112     dP_values.append(dP)
113     positions.append(position)
114     rpms.append(rpm_val)
115     motor_torque.append(tau_motor)
116 # Create DataFrame from output lists
117 df_force = pd.DataFrame({'Timestamps [s]': timestamps, 'Cycle': cycle, '
    State': states, 'WHSIP [psi]': WHSIP_values, 'dP [psi]': dP_values, '
    Position [mm]': positions, 'RPM': rpms, 'Motor Torque [lbf*ft]':
    motor_torque, 'Est. Actuator output [lbf]': Actuator_force, 'Est.
    Valve Req. [lbf]': valve_req})
118
119 # Write the updated DataFrame back to the Excel file
120 try:
121     with pd.ExcelWriter('results.xlsx', mode='a', engine='openpyxl',
    if_sheet_exists='overlay') as writer:
122         df_force.to_excel(writer, sheet_name='Digital Twin', index=False
    )
123
124
125 except FileNotFoundError:
126     with pd.ExcelWriter('results.xlsx', mode='w', engine='openpyxl') as
    writer:
127         df_force.to_excel(writer, sheet_name='Digital Twin', index=False
    )
```

B Code for Simulator

```
import math
import random
import pandas as pd
import numpy as np
```

```
#CONSTANTS

df = pd.read_excel('results.xlsx', sheet_name='Input')

# Create a dictionary with variable names as keys and their corresponding values as values
fixed_values_constants = {row['Variable']: row['Value'] for _, row in df.iterrows()}

# Replace the fixed values and constants in your code with the values from the dictionary
A_p = fixed_values_constants['A_p'] # area of lead nut
A_s = fixed_values_constants['A_s'] #Valve stem area
A_vb = fixed_values_constants['A_vb'] #Well bore area
A_gs = fixed_values_constants['A_gs'] #Gate seat area
mu = fixed_values_constants['mu'] #Friction coeff, seat and gate
F_f = fixed_values_constants['F_f'] #Stem seal friction force
P_w = fixed_values_constants['P_w'] #Water pressure
L = fixed_values_constants['L'] #Lead screw ft/rev
eff = fixed_values_constants['eff'] #Lead screw efficiency
G_R = fixed_values_constants['G_R'] #Gear ratio
E_1 = fixed_values_constants['E_1'] #Motor efficiency
E_2 = fixed_values_constants['E_2'] #Gear efficiency
rpm_open = fixed_values_constants['rpm_open']  # RPM during opening.
rpm_close = fixed_values_constants['rpm_close']  # RPM during closing.
k = fixed_values_constants['k'] #flow consistency index
n = fixed_values_constants['n'] #flow behaviour index
dP = fixed_values_constants['dP_test'] #Differential pressure set for leakage test
test_type=fixed_values_constants['Test Type'] #Normal or Partial for type of valve testing

#VALVE INTERFACE

# Set up empty list for torque data
force_data = []
torque_data = []


# Calculate the range for WHSIP which decreases and random dP_extra that represents excess dP when opening valve after test. Also set a fixed dP for testing.
WHSIP_range = np.linspace(1800, 1000, num=100)
dP_extra = np.random.uniform(low=0, high=435, size=100)
```

```
47
48
49 #Friction decrease base parameters
50 base_decrease_chance_f_f = 0.2
51 base_decrease_interval_f_f = (10, 50)
52
53 #Loop for simulating linear forces present in each state of the valve
   cycle. Also simmulates decrease in stem seal friction: magnitude and
   chance is based on excess dP when opening valve.
54 for i in range(100):
55     WHSIP = int(WHSIP_range[i])
56     dP_random = int(dP_extra[i])
57     # Calculate the decrease chance and interval for F_f based on D_P
58     decrease_chance_f_f = base_decrease_chance_f_f * dP_random / max(
   dP_extra)
59     decrease_interval_f_f = (base_decrease_interval_f_f[0] * dP_random /
       max(dP_extra),
60                              base_decrease_interval_f_f[1] * dP_random /
       max(dP_extra))
61
62     # Check if F_f should be decreased and update its value
63     if random.random() < decrease_chance_f_f:
64         F_f -= random.uniform(*decrease_interval_f_f)
65         F_f = max(0, F_f)  # Ensure F_f doesn't become negative
66
67     # Forces
68     F_1 = WHSIP * mu * (2*A_gs + A_vb)  # Force due to seat gate
   friction
69     F_2 = F_f                           # Force due to seal friction
70     F_3 = P_w * (A_p-A_s)               # Force on piston from water
   pressure
71     F_4 = WHSIP * A_s                   # Force on lower stem from well
   shut in pressure
72
73
74     # step 1: interval before pressure equalization
75     F_t1 = F_1 + F_2 + F_3 + F_4
76
77     # step 2: interval from crack open to fully open
78     F_t2 = F_2 + F_3 + F_4
79
80     # step 3: interval from fully open to crack close
81     F_t3 = -F_2 + F_3 + F_4
82
83     # step 4: interval after pressure build up
84     F_t4 = -F_1 - F_2 + F_3 + F_4
85
86     # add force data to list
87     force_data.append([WHSIP, dP, dP_random, F_t1, F_t2, F_t3, F_t4, F_f
   ])
88     # convert data to pandas dataframe
89     df_force = pd.DataFrame(force_data, columns=['WHSIP','dP','dP_random
   ', 'F_t1', 'F_t2', 'F_t3', 'F_t4', 'F_f'])
90
91
```

```
92
93 #LEAD SCREW
94
95 # New loop to convert from linear forces to torques
96
97 # eff represents the efficiency of the lead screw and decreases randomly
       of given chance within the given interval
98 eff_decrease_interval = (0.0000005, 0.0000015)
99 eff_decreasechance=0.1
100 # n represents the flow behaviour index of the fluid around the lead
      screw and increases randomly of given chance within the given
      interval
101 n_increase_interval = (0.000005, 0.00015)
102 n_increasechance=0.1
103 for i in range(100):
104     if random.random() < eff_decreasechance:
105         eff_decrease = random.uniform(*eff_decrease_interval)
106         eff -= eff_decrease
107
108     if random.random() < n_increasechance:
109         n_increase = random.uniform(*n_increase_interval)
110         n += n_increase
111
112     WHSIP, dP, dP_random, F_t1, F_t2, F_t3, F_t4, F_f = force_data[i]
113
114     T_t1 = F_t1 * L / (2 * math.pi * eff) * (1 + k * (rpm_open ** (n-1))
      )
115     T_t2 = F_t2 * L / (2 * math.pi * eff) * (1 + k * (rpm_open ** (n-1))
      )
116     T_t3 = F_t3 * L / (2 * math.pi * eff) * (1 + k * (rpm_close ** (n-1)
      ))
117     T_t4 = F_t4 * L / (2 * math.pi * eff) * (1 + k * (rpm_close ** (n-1)
      ))
118
119     torque_data.append([WHSIP, dP, dP_random, T_t1, T_t2, T_t3, T_t4,
      eff, n])
120
121 df_torque = pd.DataFrame(torque_data, columns=['WHSIP', 'dP', 'dP_random
      ', 'T_t1', 'T_t2', 'T_t3', 'T_t4', 'eff','n'])
122
123
124
125
126 #MOTOR AND GEAR
127
128 # set up empty lists for power data
129 power_t1 = []
130 power_t2 = []
131 power_t3 = []
132 power_t4 = []
133
134
135
136 # loop through torque data and calculate power for each torque value.
      This is to simulate power needed as motor input
```

```
for row in df_torque.itertuples(index=False):

    # extract torque values
    tau_t1 = row[3]
    tau_t2 = row[4]
    tau_t3 = row[5]
    tau_t4 = row[6]

    # calculate motor output torque and power input for each torque value
    tau_motor_t1 = tau_t1 / (G_R * E_2)
    power_t1.append((tau_motor_t1 * rpm_open) / (E_1 * 5252))

    tau_motor_t2 = tau_t2 / (G_R * E_2)
    power_t2.append((tau_motor_t2 * rpm_open) / (E_1 * 5252))

    tau_motor_t3 = tau_t3 / (G_R * E_2)
    power_t3.append((tau_motor_t3 * rpm_close) / (E_1 * 5252))

    tau_motor_t4 = tau_t4 / (G_R * E_2)
    power_t4.append((tau_motor_t4 * rpm_close) / (E_1 * 5252))

# create dictionary of power data with column names
power_data = {'WHSIP': df_torque['WHSIP'], 'dP': df_torque['dP'], 'dP_random': df_torque['dP_random'], 'power_t1': power_t1, 'power_t2': power_t2, 'power_t3': power_t3, 'power_t4': power_t4}



#CONVERT TO TIMESERIES

#Set time intervals based on rpm, Gear ratio and ft/rev
distance_24mm = 24 * 0.00328084                             # approximately 0.07874 feet for crack
distance_196mm = 196 * 0.00328084                           # approximately 0.64304 feet from crack to end
crack_open_time = distance_24mm/(rpm_open/G_R*L)*60         # Time used for crack open
run_open_time = distance_196mm/(rpm_open/G_R*L)*60          # Time used for crack open
stay_open_time = 100                                        # Time used for crack open
run_close_time = distance_196mm/(rpm_close/G_R*L)*60        # Time used for crack open
crack_close_time = distance_24mm/(rpm_close/G_R*L)*60       # Time used for crack open
stay_close_time = 100                                       # Time used for crack open
total_time = round(run_close_time + crack_close_time + stay_close_time + crack_open_time + run_open_time + stay_open_time)
total_time_partial = round(run_close_time*0.2 + stay_close_time + run_open_time*0.2 + stay_open_time)



```

```
# Main timeseries loop

# Assigns WHSIP and calculated force, torque and power data to timeseries (seconds and cycles). Also assigns the state, dP, rpm and position on premisses. Additional data for eff, n and f_f is included for verification in excel.
leakage = 0
dP_random_prev = None
#Cycles through calculated data with length 100
for cycle in range(100):

    #Defines a dataframe to store generated data

    # Collect torque data
    WHSIP, dP, dP_random, T_t1, T_t2, T_t3, T_t4, eff, n = torque_data[cycle]
    # 7th index of force_data contains F_f
    F_f = force_data[cycle][7]

    #Loop for normal valve testing

    if test_type == 'Normal':
        df_timeseries_cycle = pd.DataFrame(index=range(total_time), columns=["Timestamps [s]", "Cycle", "State", "WHSIP [psi]", "dP [psi]", "Position [mm]", "RPM", "Linear Force [lbf]", "Power [HP]", "Torque [lbf*ft]", "Lead Screw eff", "Flow beh.index, n", "Seal Friction [lbf]"])
        for t in range(total_time):
            if t < run_close_time:
                state = "run_close"
                dP_current = 0
                force = df_force['F_t3'][cycle]
                power = power_t3[cycle]
                torque = T_t3
                rpm = rpm_close
                position = -220 + ((220 - 24) / run_close_time) * t

      # Position changes linearly from -220 to -24
            elif t < run_close_time + crack_close_time:
                state = "crack_close"
                dP_current = 0
                force = df_force['F_t4'][cycle]
                power = power_t4[cycle]
                torque = T_t4
                rpm = rpm_close
                position = -24 + (24 / crack_close_time) * (t - run_close_time)
              # Position changes linearly from -24 to 0
            elif t < run_close_time + crack_close_time + stay_close_time:
                state = "stay_closed"
                force = df_force['F_t1'][cycle]
                power = power_t1[cycle]
                torque = T_t1
                rpm = 0
```

```
                position = 0  # Valve stays closed
                if t < run_close_time + crack_close_time + 10:
                    dP_current = 0
                elif t < run_close_time + crack_close_time + 20:
                    dP_current = dP * (t - run_close_time -
crack_close_time - 10) / 10  # Gradually increase to testing dP over
10 seconds
                elif t < run_close_time + crack_close_time + 35:
                    # Start decreasing dP with potential leakage after
15 seconds
                    dP_current = dP - leakage * (t - run_close_time -
crack_close_time - 15) / 15
                else:
                    # Decrease dP to dP_random = excess dP over next 10
seconds
                    dP_current = max((dP - leakage * 15) - (dP - leakage
 * 15 - dP_random) * (t - run_close_time - crack_close_time - 35) /
10, dP_random)
                    # dP decreases in preparation for opening. Normaly
after 10 minutes, but here 15 sec.

            elif t < run_close_time + crack_close_time + stay_close_time
 + crack_open_time:
                state = "crack_open"
                force = df_force['F_t1'][cycle]
                power = power_t1[cycle]
                torque = T_t1
                rpm = rpm_open
                position = -24 * ((t - run_close_time - crack_close_time
 - stay_close_time) / crack_open_time)
    # Position changes linearly from 0 to -24
                dP_current = dP_random

    # Excess dP remains for crack open
            elif t < run_close_time + crack_close_time + stay_close_time
 + crack_open_time + run_open_time:
                state = "run_open"
                dP_current = 0
                force = df_force['F_t2'][cycle]
                power = power_t2[cycle]
                torque = T_t2
                rpm = rpm_open
                position = -24 - ((220 - 24) / run_open_time) * (t -
run_close_time - crack_close_time - stay_close_time - crack_open_time
)     # Position changes linearly from -24 to -220
            else:
                state = "stay_open"
                dP_current = 0
                force = df_force['F_t3'][cycle]
                power = power_t3[cycle]
                torque = T_t3
                rpm = 0
                position = -220

    # Valve stays open
```

```
258
259             df_timeseries_cycle.loc[t] = [t + cycle * total_time, cycle,
     state, WHSIP, dP_current, position, rpm, force,  power, torque, eff,
     n , F_f]
260
261             #Generates potential leakage based on the excess dP from
    previous cycle
262         if dP_random_prev is not None:
263             if np.random.rand() < (dP_random_prev/10): # Adjust the
    denominator for a suitable chance
264                 leakage_increase = np.random.rand() * dP_random_prev *
    0.01 # Adjust the multiplier for a suitable increase magnitude
265                 leakage = min(max(leakage + leakage_increase, 0),
    dP_random) # Ensure leakage does not exceed dP_random_prev
266         dP_random_prev = dP_random
267
268         if cycle == 0:
269             df_timeseries = df_timeseries_cycle.copy()
270         else:
271             df_timeseries = pd.concat([df_timeseries,
    df_timeseries_cycle])
272
273     #Loop for partial stroke testing ---> Does not include crack close
    and crack open. Position and time is ajusted for only 20% of run
    close
274     elif test_type == 'Partial':
275         df_timeseries_cycle = pd.DataFrame(index=range(total_time_partial
    ), columns=["Timestamps [s]", "Cycle", "State", "WHSIP [psi]", "dP [
    psi]", "Position [mm]", "RPM", "Linear Force [lbf]", "Power [HP]", "
    Torque [lbf*ft]", "Lead Screw eff", "Flow beh.index, n", "Seal
    Friction [lbf]"])
276         for t in range(total_time_partial):
277             end_position_run_close = -220+(220-24)*0.2
                                    # end position for run_close state in
    partial stroke test
278             if t < run_close_time*0.2:
279                 state = "run_close"
280                 dP_current = 0
281                 force = df_force['F_t3'][cycle]
282                 power = power_t3[cycle]
283                 torque = T_t3
284                 rpm = rpm_close
285                 position = -220 + ((end_position_run_close + 220) / (
    run_close_time*0.2)) * t       # Position changes linearly from -220
    to end_position_run_close over the time 'run_close_time*0.2'
                                                                # Position
    changes linearly from -220 to -180.4 = (-220+(220-24)*0.2)
286
287             elif t < run_close_time*0.2 + stay_close_time:
288                 state = "stay_closed"
289                 dP_current = 0
290                 force = df_force['F_t2'][cycle]
291                 power = power_t2[cycle]
292                 torque = T_t2
293                 rpm = 0
```

```
                position = end_position_run_close
                                # Valve stays at end_position_run_close


            elif t < run_close_time*0.2 + stay_close_time + run_open_time*0.2:
                state = "run_open"
                dP_current = 0
                force = df_force['F_t2'][cycle]
                power = power_t2[cycle]
                torque = T_t2
                rpm = rpm_open
                position = end_position_run_close - ((end_position_run_close + 220) / (run_open_time*0.2)) * (t - run_close_time*0.2 - stay_close_time) # Position changes linearly from end_position_run_close to -220 over the time 'run_open_time*0.2'

            else:
                state = "stay_open"
                dP_current = 0
                force = df_force['F_t3'][cycle]
                power = power_t3[cycle]
                torque = T_t3
                rpm = rpm_open
                position = -220
                                # Valve stays open at -220
            df_timeseries_cycle.loc[t] = [t + cycle * total_time_partial, cycle, state, WHSIP, dP_current, position, rpm, force, power, torque, eff, n , F_f]
        if cycle == 0:
            df_timeseries = df_timeseries_cycle.copy()
        else:
            df_timeseries = pd.concat([df_timeseries, df_timeseries_cycle])
df_timeseries.reset_index(drop=True, inplace=True)



# Save to Excel
df_timeseries.to_excel("timeseries.xlsx")

# Write the updated DataFrame back to the Excel file
try:
    with pd.ExcelWriter('results.xlsx', mode='a', engine='openpyxl', if_sheet_exists='overlay') as writer:
        df_timeseries.to_excel(writer, sheet_name='Data', index=False)


except FileNotFoundError:
    with pd.ExcelWriter('results.xlsx', mode='w', engine='openpyxl') as writer:
        df_timeseries.to_excel(writer, sheet_name='Data', index=False)
```

References

[1] DrillingFormulas.Com |. *What is a Vertical Subsea Christmas Tree (Conventional Subsea Tree)?* en-US. Aug. 2016. URL: https://www.drillingformulas.com/what-is-vertical-subsea-christmas-tree-conventional-subsea-tree/ (visited on 11/20/2022).

[2] Equinor 2023. *Personal Communucation, Børge Pettersen.*

[3] NTNU 2023. *Personal Communication, Head Engineers Torkjell Breivik and Noralf Vedvik.*

[4] Nic Abraham. *What are ball screws?* en-US. Oct. 2016. URL: https://www.medicaldesignandoutsourcing.com/what-are-ball-screws/ (visited on 06/30/2023).

[5] *Acme Screws*. en-US. URL: https://rockfordballscrew.com/ballscrews/acme/ (visited on 06/30/2023).

[6] *All-electric subsea systems deliver intelligence on demand*. June 2018. URL: https://www.offshore-mag.com/subsea/article/16762050/allelectric-subsea-systems-deliver-intelligence-on-demand (visited on 07/02/2023).

[7] Artesis. *Motor Current Signature Analysis - Artesis*. en-US. Section: Predictive Maintenance. Apr. 2021. URL: https://www.artesis.com/motor-current-signature-analysis/ (visited on 07/07/2023).

[8] Yong Bai and Qiang Bai. *Subsea Engineering Handbook*. Oxford, UNITED STATES: Elsevier Science & Technology, 2012. ISBN: 978-0-12-397805-9. URL: http://ebookcentral.proquest.com/lib/ntnu/detail.action?docID=842202 (visited on 06/30/2023).

[9] Tor Berge Gjersvik. *Lecture notes from TPG4200 - Subsea Technology. Chapter: subsea production systems.*

[10] *Calculations*. en-US. URL: https://simplemotor.com/calculations/ (visited on 12/19/2022).

[11] *Control Valve Flow Coefficients*. en. ISSN: 0733-947X. URL: https://ascelibrary.org/doi/epdf/10.1061/%28ASCE%290733-947X%281985%29111%3A4%28358%29 (visited on 12/09/2022).

[12] P Czop et al. "Formulation and identification of First- Principle Data-Driven models". en. In: *Journal of Achievements in Materials and Manufacturing Engineering* 44.2 (2011).

[13] EEQnA. *Working principle of AC motors*. Nov. 2016. URL: https://www.youtube.com/watch?v=qbNpONXRvj8 (visited on 06/29/2023).

[14] *Electric motors and everything about them | Elektromotory.cz*. en. URL: http://www.k-drives.eu/electric-motors-and-everything-about-them (visited on 07/08/2023).

[15] Engineers Academy. *Simple Gear Ratios, Input and Output Speed, Torque and Power*. Feb. 2019. URL: https://www.youtube.com/watch?v=H9_Llzcz2cY (visited on 12/19/2022).

[16] *Faraday's law of induction | Definition, Formula, & Facts | Britannica*. en. May 2023. URL: https://www.britannica.com/science/Faradays-law-of-induction (visited on 06/29/2023).

[17] *Flow Properties of Polymers, newtonian fluids*. URL: https://polymerdatabase.com/polymer%20physics/Viscosity.html (visited on 06/30/2023).

[18] *Flow Properties of Polymers, non Newtonian fluids*. URL: https://polymerdatabase.com/polymer%20physics/Viscosity2.html (visited on 06/30/2023).

[19] Hanne Fuglestad. "Service Line-Less Subsea Production and Injection". eng. Accepted: 2020-06-04T16:04:15Z. MA thesis. NTNU, 2020. URL: https://ntnuopen.ntnu.no/ntnu-xmlui/handle/11250/2656768 (visited on 12/20/2022).

[20] Einar H. Winther-Larssen. *Design of an Electric X-mas Tree Gate Valve Actuator*. June 2007. URL: https://ntnuopen.ntnu.no/ntnu-xmlui/bitstream/handle/11250/259861/348501_FULLTEXT01.pdf?sequence=1 (visited on 11/10/2022).

[21] Thomas Henanger, G. Muller, and Luca Piciaccia. "Managing Installation Tolerances through System Modeling and Tolerance Budgeting". In: *INCOSE International Symposium* 26 (July 2016), pp. 1176–1191. DOI: 10.1002/j.2334-5837.2016.00219.x.

[22] Sebastian Imle et al. "Safety and Reliability Analysis of an Actuation System". en. In: *Proceedings of the 29th European Safety and Reliability Conference (ESREL)*. Research Publishing Services, 2019, pp. 3675–3682. ISBN: 978-981-11-2724-3. DOI: 10.3850/978-981-11-2724-3_0885-cd. URL: http://rpsonline.com.sg/proceedings/9789811127243/html/0885.xml (visited on 12/01/2022).

[23] Dario Leon, Sebastian Imle, and Markus Glaser. "Wear-optimized partial stroke test for all-electric actuation systems". en. In: *Gas Science and Engineering* 111 (Mar. 2023), p. 204875. ISSN: 29499089. DOI: 10.1016/j.jgsce.2023.204875. URL: https://linkinghub.elsevier.com/retrieve/pii/S2949908923000031 (visited on 07/05/2023).

[24] Carsten Mahler et al. "Safety Capability of an All-Electric Production System". en. In: *Day 3 Wed, May 08, 2019*. Houston, Texas: OTC, Apr. 2019, D031S041R003. DOI: 10.4043/29472-MS. URL: https://onepetro.org/OTCONF/proceedings/19OTC/3-19OTC/Houston,%20Texas/181387 (visited on 11/22/2022).

[25] R. Keith Mobley. "ROLE OF MAINTENANCE ORGANIZATION". en. In: *An Introduction to Predictive Maintenance*. 2nd ed. 2002. DOI: 10.1016/B978-075067531-4/50003-8. URL: https://reader.elsevier.com/reader/sd/pii/B9780750675314500038?token=53152BEEF118AAA315A119B01EB58E10E38695443237BA2141F7C04D830C36B81233F16CB87originRegion=eu-west-1&originCreation=20221212065706 (visited on 12/12/2022).

[26] *Petroleumstilsynet stenger Goliat-plattformen*. nb. Oct. 2017. URL: https://www.aftenposten.no/okonomi/i/n8pAn/petroleumstilsynet-stenger-goliat-plattformen (visited on 07/02/2023).

[27] RBS. *Ball Screw vs Lead Screw: Everything You Need to Know*. en-US. Jan. 2023. URL: https://rockfordballscrew.com/ball-screw-vs-lead-screw-everything-you-need-to-know-2/ (visited on 06/30/2023).

[28] *Root mean square*. en. Page Version ID: 1162165222. June 2023. URL: https://en.wikipedia.org/w/index.php?title=Root_mean_square&oldid=1162165222 (visited on 06/30/2023).

[29] A. Rubio, H. Abu Zeid, and J. H. Meyer. "Benefits of Using an Electric Choke for Subsea Applications". en. In: *Day 2 Wed, October 25, 2017*. Rio de Janeiro, Brazil: OTC, Oct. 2017, D021S013R004. DOI: 10.4043/28138-MS. URL: https://onepetro.org/OTCBRASIL/proceedings/17OTCB/2-17OTCB/Rio%20de%20Janeiro,%20Brazil/93912 (visited on 07/03/2023).

[30] Jørgen Hagemo Sæther. "Choke condition and performance monitoring". eng. Accepted: 2014-12-19T12:06:11Z. MA thesis. Norges teknisk-naturvitenskapelige universitet, Fakultet for ingeniørvitenskap og teknologi, Institutt for marin teknikk, 2010. URL: https://ntnuopen.ntnu.no/ntnu-xmlui/handle/11250/237833 (visited on 07/03/2023).

[31] Eirik Soland. *Condition monitoring system for all-electric X-mas trees. Specialization project 2022*.

[32] *Stepping actuator operated choke » Products » JVS Engineers*. URL: https://www.jvsengg.com/products/stepping-actuator-operated-choke/ (visited on 07/03/2023).

[33] *Tar siste stikk på diagnostikk*. Dec. 2012. URL: https://www.tu.no/artikler/tar-siste-stikk-pa-diagnostikk/218743 (visited on 07/02/2023).

[34] Simon Tattersall. "Choke Valve Technology in Subsea Environments". en. In: *Measurement and Control* 49.3 (Apr. 2016). Publisher: SAGE Publications Ltd, pp. 104–108. ISSN: 0020-2940. DOI: 10.1177/0020294016640556. URL: https://doi.org/10.1177/0020294016640556 (visited on 07/03/2023).

[35] Jake Wattenphul. *How a 3 Phase AC Induction Motor Works*. en-US. June 2023. URL: https://www.kebamerica.com/blog/how-a-3-phase-ac-induction-motor-works/ (visited on 06/29/2023).

[36] *What are Load Cells and How Do They Work?* en-US. URL: https://www.omega.com/en-us/resources/load-cells (visited on 07/07/2023).

[37] Author Electrical Workbook. *What is Strain Gauge Load Cell? Working Principle, Construction & Applications*. en-US. July 2021. URL: https://electricalworkbook.com/strain-gauge-load-cell/ (visited on 07/07/2023).

[38] Alberto Portapuglia www.portapuglia.com. *Quam valves | GAH-SDV Series Full bore through conduit Slab Gate valve complete with Rising Stem Hydraulic Actuator ASH Series*. en-US. URL: https://www.iamquam.it/Slab-Gate-valve-hydraulic-actuated-single-acting-fail-close-standard.php (visited on 12/18/2022).