

Structural Integrity Analyses of Inline Structures in the Overbend Section

ME54035: Graduation Project

Ivan Lucas Gonzalo

TU Delft

Allseas

Structural Integrity Analyses of Inline Structures in the Overbend Section

By

Ivan Lucas Gonzalo

Master Thesis

in partial fulfilment of the requirements for the degree of

Master of Science
in Mechanical Engineering

at the Department Maritime and Transport Technology of Faculty Mechanical Engineering of Delft University of Technology
to be defended publicly on Tuesday **May 6**, **2026**

Student Number:	5004969	
MSc track:	Multi-Machine Engineering	
Report Number:	2026.MME.9162	
Thesis Committee:	Ir. W. van den Bos	TU Delft Supervisor, ME
	Dr. J. Jovanova	TU Delft Committee Chair, ME
	Dr. Eng. F. Gerspach	Company Supervisor, Allseas
Project Duration:	03, 2025 - 04, 2026	

An electronic version of this thesis is available at `http://repository.tudelft.nl/`.

It may only be reproduced literally and as a whole. For commercial purposes only with written authorization of Delft University of Technology. Requests for consult are only taken into consideration under the condition that the applicant denies all legal rights on liabilities concerning the contents of the advice.

Cover: Audacia [1]
Style: TU Delft Report Style, with modifications by Daan Zwaneveld

TU Delft

Allseas

Preface

As I am typing this preface, I am filled with a profound sense of accomplishment and gratitude. The thesis that lays before you represents not just a culmination of my knowledge, dedication and hard work towards becoming a Mechanical Engineer. It also represents a journey of immense personal growth, discovery, failures and victories.

From leaving my home country and settling 7804 kilometres away, to taking a path paved with countless hours of hard work, late nights, and moments of doubts just to name a few - this road is the one I was meant to take, and I would not have wanted it any other way.

This thesis would not have been possible without the immense support of my thesis committee, comprising of Dr. Eng. François Gerspach, ir. Wouter van den Bos, and Dr. Jovana Jovanova. Their feedback, knowledge and expertise have been of immense help in following the correct path towards reaching the ultimate goal of successfully completing this graduation project. I would also like to thank my colleagues at Allseas, who enriched my time at the company, with knowledge and joy.

I would like to extend my heartfelt thanks to my family and friends. Your belief in me has been a constant source of strength. To my parents, thank you for your endless support and for always being my biggest cheerleaders. To my friends, thank you for your patience and understanding during the times when I was consumed by my research.

Lastly, I want to acknowledge the personal growth I have experienced throughout this process. This thesis has challenged me to step out of my comfort zone, to embrace failure as a stepping stone to success, and to persist in the face of adversity. It has taught me the value of resilience and the importance of passion in pursuing one's goals.

As I present this work, I do so with a deep sense of gratitude and fulfilment. It is my hope that this thesis will contribute to the field of mechanical engineering and inspire future research.

Ivan Lucas Gonzalo
Delft, April 2026

AI Statement

For this report for the course *ME54035*, I have used Generative AI to:

- Improve the grammar, style, layout, and/or spelling of the text.

In all cases I have reviewed and corrected the work and remain fully responsible for the content of the report.

Summary

This research project investigates pipeline installation analyses by means of using non-linear finite element models and multi-body models. Non-linear finite element models allow for high-fidelity modelling, and is therefore mostly used for localized analyses. Multi-body models allow for complete modelling of installation scenarios, where global behaviour of the pipeline can be analysed. Multi-body analyses are standard procedure for pipeline installation analyses due to their significantly low runtime for capturing global behaviour.

Non-linear finite element models allow for high-fidelity, thus models created are considered realistic. These are also used to analysed global behaviour, however are only used to benchmark the multi-body models as global behaviour analyses models have significantly high runtime due to their high fidelity. The multi-body models have an overall low fidelity, allowing for low runtimes whilst still modelling the complete scenario and capturing pipeline behaviour. For the case of homogeneous pipelines, the multi-body models can closely approximate its behaviour. However, when including an inline structure within the pipeline, questions regarding current modelling viability and result fidelity arise.

This research project aims to analyse the current pipeline installation analyses method. This entails assessing both the multi-body models and non-linear finite element models, and respective roles for both homogeneous pipeline and inline structure analyses. An inline structure of a past project is used in order to benchmark its results from multi-body and non-linear finite element analyses. The viability of a time-efficient inline structure modelling method is also investigated.

Both modelling setups are compared in order to gain further knowledge on their similarities and differences. Additionally, possible pain points and respective solutions & improvements are proposed. Resulting from this comparison, two of these improvements are directly implemented in order to minimize the modelling setup difference. By minimizing this difference, any further difference in results can be attributed to inherent systematic differences.

Due to the fact that both multi-body and non-linear finite elements models can closely approximate the behaviour of a homogeneous pipeline, a pipeline is modelled in both software using similar setups, and their static analysis results are compared. By comparing the results of an identical homogeneous pipeline, these resulting inherent systematic differences can be highlighted. Knowing these differences beforehand, one can take these into account when comparing results of an inline structure.

This research project concludes that, for homogeneous pipeline installation analyses, multi-body models close approximate the results of non-linear finite element models. For the case of an inline structure, result discrepancies arise. Multi-body models slightly overestimate these results, yet this overestimation is deemed as a conservative design approach. Additionally, the alternative time-efficient modelling method resulted in two simplified inline structures, where each had a different material definition. Both inline structures delivered similar results, despite having a lower fidelity when compared to the original multi-body inline structure model. A modification to this new time-efficient method is proposed in hindsight in order to increase its viability.

In cases where time is of essence and quick analyses are needed, simplifying the inline structure in such a way delivers conservative results which approximate actuality. Compliance with the established design criteria using this simplified inline structure implies that its design is structurally sound and viable for installation. In cases where there is no such time crunch, it is suggested to slightly improve the fidelity in OrcaFlex by increasing the number of sections used - i.e. more than three as used in this research project. Higher fidelity reduces the level of design conservatism and lowers over-engineering, thereby reducing the overall production cost

Contents

Preface i

AI Statement ii

Summary iii

Nomenclature x

1 Introduction 1
- 1.1 Pipeline Installation . . . 1
  - 1.1.1 Installation process . . . 2
  - 1.1.2 Overbend Section . . . 2
  - 1.1.3 Sagbend Section . . . 2
  - 1.1.4 Stinger . . . 3
  - 1.1.5 Rollerboxes . . . 3
- 1.2 Pipeline Installation within Allseas . . . 4
  - 1.2.1 Inventor . . . 5
  - 1.2.2 Multi-body Modelling . . . 5
  - 1.2.3 Non-linear Finite Element Modelling . . . 7
- 1.3 Report Structure . . . 8

2 Problem Formulation 9
- 2.1 Current Process . . . 9
  - 2.1.1 Inline Structure Design . . . 10
- 2.2 Current Issue . . . 12
- 2.3 Approach . . . 13
  - 2.3.1 Research Sub-question 1 . . . 13
  - 2.3.2 Research Sub-question 2 . . . 13
  - 2.3.3 Research Sub-question 3 . . . 13
  - 2.3.4 Research Sub-question 4 . . . 14
  - 2.3.5 Research Sub-question 5 . . . 14
  - 2.3.6 Research Question . . . 14
  - 2.3.7 Research Scope . . . 14

3 Method 17
- 3.1 Pipeline Installation Analysis . . . 17
  - 3.1.1 Comparison Results . . . 18
  - 3.1.2 Setup Modifications . . . 20
- 3.2 Pipeline Comparison . . . 22
  - 3.2.1 Lumped-mass Pipeline Model . . . 23
  - 3.2.2 Non-linear Finite-Element Pipeline Model . . . 25
  - 3.2.3 Extracted Result Variables . . . 31
- 3.3 Inline Structure Positioning . . . 32
  - 3.3.1 Max. Rollerbox Reaction Force . . . 34
  - 3.3.2 Max. Bending Angle Difference of the ILS . . . 34
  - 3.3.3 Python code . . . 35
- 3.4 Inline Structure Comparison . . . 36
  - 3.4.1 3D Non-linear Finite-Element ILS Model . . . 36
  - 3.4.2 Lumped-mass ILS Model . . . 37
- 3.5 Time-efficient ILS . . . 38
  - 3.5.1 Time-efficient Methods . . . 38

3.6 Python Script for Abaqus . . . 42

4 Results **43**
4.1 Pipeline Difference . . . 43
4.1.1 Material Consistency . . . 43
4.1.2 Solid Pipe Location . . . 45
4.1.3 Rollerboxes Configuration . . . 46
4.1.4 Result Conclusions . . . 48
4.2 Inline Structure Positions . . . 53
4.2.1 Max. Rollerbox Reaction Force Position . . . 54
4.2.2 Max. Bending Angle Difference Position . . . 55
4.3 Inline Structure Difference . . . 56
4.3.1 Max RB Reaction Force case . . . 56
4.3.2 Max Bending Angle Difference case . . . 61
4.4 Time-efficient ILS Results . . . 64
4.5 Implementation . . . 66

5 Discussion **68**
5.1 Modelling setup . . . 68
5.2 Homogeneous Pipeline . . . 68
5.3 Inline Structures . . . 69
5.4 Time-efficient Methods . . . 69

6 Conclusion **71**
6.1 Key conclusions . . . 71
6.2 Research questions . . . 73
6.2.1 Sub-questions and answers: . . . 73
6.2.2 Main question and answer: . . . 74

7 Recommendations **75**

References **77**

A Research Paper **79**

B Figures & Tables **88**
B.1 Figures . . . 89
B.1.1 Chapter 1: Introduction . . . 89
B.1.2 Chapter 2: Problem Formulation . . . 92
B.1.3 Chapter 3: Method . . . 93
B.1.4 Chapter 4: Results . . . 108
B.2 Tables . . . 113

C Coding **130**
C.1 Inline Structure Displacement Code . . . 130
C.2 Homogenous Pipeline Code . . . 133
C.3 Max Rollerbox Reaction Force Case Code . . . 135
C.4 Max Declination Angle Difference Case Code . . . 138

D Definitions **141**

List of Figures

1.1 Schematic diagram of S-Lay installation process [4] . . . 1
1.2 Rollerbox T2: a) Front-side view, not mounted on the stinger b) Top-side view, mounted on the stinger . . . 3
1.3 Rollerbox behaviour illustrated trough: a) Rollerbox drawing b) Real-life image . . . 4
1.4 Example of a subsea pipeline infrastructure [6]. . . . 4
1.5 Isometric view for the inline structure for Shenandoah . . . 5
1.6 Case example in OrcaFlex: Pipelay by *Solitaire* . . . 5
1.7 Drop-down menu of the types of components in OrcaFlex . . . 6
1.8 Lumped-mass model of a pipeline containing three sections . . . 6
1.9 Application of Abaqus within Allseas: **a)** Setup of the Abaqus model of a part **b)** Von Mises stress results for the individual components . . . 7

2.1 An ILS passing on the stinger during pipelay . . . 9
2.2 Inline Structure with the first four sections: A-A, B-B, C-C, and D-D . . . 10
2.3 Workflow: Global steps of Inline Structure (ILS) design . . . 11
2.4 Pipeline installation: a) Homogeneous pipeline b) Pipeline with an inline structure . . . 12
2.5 Visualization of the simplified inline structure installation problem - Assumption 11 . . . 16

3.1 Top view of the stinger during pipeline installation . . . 17
3.2 Full overbend model setup in Abaqus, before adding support rollers . . . 18
3.3 a) Rollerbox drawing b) Rollerbox modelled in OrcaFlex c) Rollerbox modelled in Abaqus 20
3.4 Full overbend model setup in Abaqus, after adding support rollers . . . 21
3.5 a) Rollerbox modelled in OrcaFlex c) Rollerbox modification in Abaqus . . . 21
3.6 Cross section of pipe section . . . 22
3.7 Overview of the pipeline in OrcaFlex . . . 24
3.8 Previous rollerbox model: a) Isometric view b) Side view . . . 25
3.9 Improved rollerbox model: a) Isometric view b) Side view . . . 26
3.10 Declination angle, the angle with respect to the horizontal (X) axis. . . . 26
3.11 Example of a pipeline comprising of both line elements *PIPE31H* and solid elements *C3D8R* . . . 27
3.12 Illustration of 1D, 2D, and 3D elements used . . . 30
3.13 Loads at element interfaces, for the case of a solid-line element pipeline . . . 32
3.14 Old rollerbox: a) Isometric view b) Side view . . . 33
3.15 Rollerbox reaction force breakdown . . . 34
3.16 Declination angle at the first and last node of the ILS . . . 35
3.17 Components of the 3D non-linear finite element Inline Structure: Rollerguide (Shell elements) and Inline Wye (Solid Elements) . . . 36
3.18 3D Non-linear finite element Inline Structure in the overbend section . . . 37
3.19 Side view of the inline structure with OrcaFlex sections . . . 38
3.20 Non-linear finite element inline structure with illustrative sections . . . 39
3.21 Non-linear finite element inline structure with illustrative sections compared to its CAD model . . . 40
3.22 Median bending moment and curvature calculation example for one section: a) node pairs and parts n lengths b) bending moment . . . 41
3.23 Average bending moment and curvature calculation example for one section: a) node pairs and parts n lengths b) bending moment . . . 41

4.1 The plastic area of the resulting bending moment-curvature relation curves . . . 44
4.2 Worst ZZ strain over: a) The overbend section b) Zoom-in for RB S1 and RB S2 . . . 45

4.3 a) Rollerbox terms in OrcaFlex b) Trigonometric calculation illustration for its coordinates 46
4.4 Rollerboxes configuration comparison for Abaqus and OrcaFlex 47
4.5 Declination angle comparison 48
4.6 Top and bottom tension comparison for all 4 pipelines 48
4.7 Comparison of the loads at the ILS interface a) Tension b) Bending Moment 49
4.8 Rollerbox reaction force comparison of OrcaFlex with Abaqus: a) Line-Elements b) Difference values 50
4.9 Axial strain components comparison for homogeneous line-element pipeline 51
4.10 Strain comparison: a) OrcaFlex, Abaqus line and solid elements b) Zoomed In 52
4.11 Change in the centreline by means of ovalization visualized: a) No ovalization b) Ovalization 52
4.12 Change in the centreline comparison of Abaqus and OrcaFlex: a) Centreline between top and bottom wall b) A zoom in into the centreline change 53
4.13 ILS displacement illustration: a) Start model b) End 53
4.14 Rollerbox reaction forces for RB V16, RB S1, and RB S2 during all displacements . . . 54
4.15 ILS position for the Max Rollerbox reaction force difference 54
4.16 Bending angle difference of the inline structure ends during all displacements 55
4.17 ILS position for the Max Bending angle difference 55
4.18 Max RB Reaction Force case: Rollerboxes configuration comparison 56
4.19 Max RB Reaction Force case: Overbend section Declination Angle comparison 57
4.20 Max RB Reaction Force case: Top & Bottom Tension comparison 57
4.21 Max RB Reaction Force case: ILS interface loads comparison a) Tension b) Bending Moment 58
4.22 Max RB Reaction Force case: Rollerbox reaction force comparison 58
4.23 Max RB Reaction Force case: Axial strain comparison 59
4.24 Inline Structure model comparison 59
4.25 Max RB Reaction Force case: ILS modelling differences 60
4.26 Max RB Reaction Force case: Axial strain comparison 60
4.27 Max Bending angle difference case: Rollerboxes configuration comparison 61
4.28 Max Bending angle difference case: Overbend section Declination Angle comparison . 61
4.29 Max Bending angle difference case: Top & Bottom Tension comparison 62
4.30 Max Bending angle difference case: ILS interface loads comparison a) Tension b) Bending Moment 62
4.31 Max Bending angle difference case: Rollerbox reaction force comparison 63
4.32 Max Bending angle difference case: Axial strain comparison 63
4.33 Comparison of the inline structures: a) Original version b) Alternative version 64
4.34 Rollerbox reaction forces for all four inline structures 64
4.35 a) Comparison of the axial strains b) Zoomed In 65
4.36 Detailed framework for the new ILS creation method 66

B.1 Images during pipeline installation of: a) Pipeline through a tensioner b) Vessel Support Rollers c) Pipe sections d) ILS on the stinger 89
B.2 Rollerbox setup drawing for Solitaire, with respective tags: a) Vessel rollerboxes b) Stinger rollerboxes 90
B.3 Stinger configuration: Stinger Length ($L_{Stinger}$), Stinger Radius ($R_{Stinger}$), Stinger Angle ($\beta_{Stinger}$) 91
B.4 a) Overview b) Pipeline after increasing tension c) Pipeline after reducing tension . . . 91
B.5 Actual pipeline and its discretized model in OrcaFlex, illustratin respective nodes and springs [21] 92
B.6 OrcaFlex inline structure section property sheet 92
B.7 Full OrcaFlex model: a) 2D view b) 3D view 93
B.8 Zoom into the vessel and stinger in the OrcaFlex model: a) 2D view b) 3D view 94
B.9 Zoom into a vessel rollerbox in the OrcaFlex model: a) 2D view b) 3D view 95
B.10 Zoom into a stinger rollerbox in the OrcaFlex model: a) 2D view b) 3D view 96
B.11 Zoom into the vessel rollerboxes V13-V16 in the OrcaFlex model: a) 2D view b) 3D view 97
B.12 Zoom into the stinger rollerboxes S1-S9 in the OrcaFlex model: a) 2D view b) 3D view . 98

B.13 Zoom into the stinger rollerboxes S10-S13 in the OrcaFlex model: a) 2D view b) 3D view 99
B.14 Zoom into the stinger rollerboxes S14-S17 in the OrcaFlex model: a) 2D view b) 3D view 100
B.15 Zoom into the stinger rollerbox S17 in the OrcaFlex model: a) 2D view b) 3D view . . . 101
B.16 Zoom into the vessel and inline structure in the OrcaFlex model: a) 2D view b) 3D view 102
B.17 Rollerbox properties, showing normal and shear stiffness, diameter, length and angle (from the horizontal axis), for: a) Support Roller b) Rollerbox V13 - V16 c) Rollerbox S1 - S9 d) Rollerbox S10 e) Rollerbox S11 f) Rollerbox S12 - S14 g) Rollerbox S15 h) Rollerbox S16 i) Rollerbox S17 . . . 103
B.18 Support rollers and vessel rollerbox dimension, mirror plane, and reference point: a) Support Roller 1, 2, 3 Front view b) Support Roller 1, 2, 3 Side view c) RB V13-V16 Front view d) RB V13-V16 Side view . . . 104
B.19 Stinger rollerbox dimension, mirror plane, and reference point: a) S1-S9 Front view b) S1-S9 Side view c) S10, S11, S15, S16 Front view d) S10, S11, S15, S16 Side view e) S12-S14 Front view f) S12-S14 Side view g) S17 Front view h) S17 Side view . . . 105
B.20 Interaction properties for frictionless sliding between the pipeline and the rollerboxes: a) Tangential Behaviour b) Normal Behaviour c) Damping . . . 106
B.21 Stress-Strain curve for the pipeline in Abaqus, for plastic behaviour . . . 107
B.22 Simplified Inline Structure in Abaqus, with respective point masses: a) Side view b) Isometric view . . . 107
B.23 Top view of the inline structure with OrcaFlex sections . . . 108
B.24 Material curves comparison, showing the old OrcaFlex and Abaqus material, and the new filtered material . . . 108
B.25 Declination angle for the rollerboxes used in both Abaqus and OrcaFlex, from RB S17 (leftmost point) to RB V13 (rightmost point) - For the comparison of the homogeneous pipeline . . . 109
B.26 Rollerbox reaction force comparison of OrcaFlex with Abaqus: a) Line-Elements b) Solid-Elements C) Longer Solid-Elements . . . 110
B.27 Inline structure illustration for the Max RB reaction force case: a) Abaqus b) OrcaFlex . 111
B.28 Declination angle for the rollerboxes used in both Abaqus and OrcaFlex, from RB S17 (leftmost point) to RB V13 (rightmost point) - For the Max RB reaction forces case comparison . . . 111
B.29 Inline structure illustration for the Max RB reaction force case: a) Abaqus b) OrcaFlex . 112
B.30 Declination angle for the rollerboxes used in both Abaqus and OrcaFlex, from RB S17 (leftmost point) to RB V13 (rightmost point) - For the Max bending angle difference case comparison . . . 112

List of Tables

2.1 Section properties sheet of the inline structure: first four sections of Figure B.6, illustrated in Figure 2.2 . . . 11

3.1 General comparison between Abaqus and OrcaFlex, for static simulation of a pipeline installation . . . 19
3.2 Values of the variables used in the case studied, for both Abaqus and OrcaFlex . . . 22
3.3 Pipe section properties: a) For the Abaqus model b) For the OrcaFlex model . . . 23
3.4 Rollerboxes modelled in Abaqus using Rollerbox T1, T2, T3, T4 and T5. . . . 26
3.5 Mesh used for in Abaqus, for all components . . . 29
3.6 Increments, in percentage of the final value, for the variables used in the Abaqus model for pipeline comparison, for each of the five steps. . . . 30
3.7 Rollerbox reaction force breakdown . . . 34
3.8 Breakdown of the meshes per component . . . 37

B.1 General comparison between Abaqus and OrcaFlex, for static simulation of a pipeline installation . . . 114
B.2 Results of the general comparison between Abaqus and OrcaFlex, for static simulation of a pipeline installation . . . 116
B.3 In-depth comparison between a pipeline in Abaqus and OrcaFlex, for static simulation of a pipeline installation . . . 118
B.4 Results of the in-depth comparison between a pipeline in Abaqus and OrcaFlex, for static simulation of a pipeline installation . . . 120
B.5 In-depth comparison between an inline structure in Abaqus and OrcaFlex, for static simulation of a pipeline installation . . . 121
B.6 Results of the in-depth comparison between an inline structure in Abaqus and OrcaFlex, for static simulation of a pipeline installation . . . 122
B.7 Values of the variables used in the Abaqus model for pipeline comparison, for the pretension step . . . 123
B.8 Values of the variables used in the Abaqus model for pipeline comparison, for the first contact step . . . 123
B.9 Values of the variables used in the Abaqus model for pipeline comparison, for the second contact step . . . 123
B.10 Values of the variables used in the Abaqus model for pipeline comparison, for the thirds contact step . . . 123
B.11 Values of the variables used in the Abaqus model for pipeline comparison, for the final step . . . 123
B.12 Relative distance from Rollerbox Main Hinge to each Hinge, given per rollerbox . . . 123
B.13 Pipeline comparison results: OrcaFlex declination angles for Hinge 1 and Hinge 2 . . . 124
B.14 Pipeline comparison results: OrcaFlex declination angles for Hinge 1 and Hinge 2 . . . 125
B.15 ILS Max RB reaction force results: OrcaFlex declination angles for Hinge 1 and Hinge 2 126
B.16 ILS Max RB reaction force results: OrcaFlex declination angles for Hinge 1 and Hinge 2 127
B.17 ILS Max bending angle difference results: OrcaFlex declination angles for Hinge 1 and Hinge 2 . . . 127
B.18 ILS Max bending angle difference results: OrcaFlex declination angles for Hinge 1 and Hinge 2 . . . 129

Nomenclature

Abbreviations

Abbreviation	Definition
CAD	Computer-Aided Design
DOF	Degree(s) of Freedom
ILS	Inline Structure
$ILSs$	Inline Structures
$OrcFxAPI$	OrcaFlex Application Programming Interface
RB	Rollerbox
RBs	Rollerboxes
$RB\ V$	Rollerbox on the vessel
$RB\ S$	Rollerbox on the stinger of the vessel
UI	User Interface

Symbols

Symbol	Definition	Unit
ϵ	Strain (Epsilon)	[%]
σ	Stress (Sigma)	[Pa]
ν	Poisson's ratio (Nu)	[$-$]
ρ	Density (Rho)	[kg/m^3]
ϕ	Declination Angle (Phi)	[$degrees$]
AS	Axial Stiffness	[N]
BS	Bending Stiffness	[Nm^2]
E	Young's Modulus	[Pa]
F	Force	[N]
ID	Inner Diameter	[m]
L	Length	[m]
M	Moment	[Nm]
m_L	Mass per unit Length	[kg/m]
MoI	Moment of Inertia	[mm^4]
OD	Outer Diameter	[m]
t_w	Wall Thickness	[m]
TS	Torsional Stiffness	[Nm^2]
U	Spatial displacement	[m]
$U1$	Spatial displacement in the direction of the X axis	[m]
$U2$	Spatial displacement in the direction of the Y axis	[m]
$U3$	Spatial displacement in the direction of the Z axis	[m]
UR	Rotational displacement	[$radians$]
$UR1$	Rotational displacement around the X axis	[$radians$]
$UR2$	Rotational displacement around the Y axis	[$radians$]
$UR3$	Rotational displacement around the Z axis	[$radians$]

1

Introduction

This chapter introduces the research topic, outlining the tools used throughout this research project. Section 1.1 provides a general introduction in to pipeline installation in order to build context. Section 1.2 discusses how pipeline installation is carried out within Allseas, touching upon the current simulation procedures and the software utilized. Section 1.3 summarizes the subsequent structure of the report, providing guidance on how to read the upcoming chapters for optimal understanding.

1.1. Pipeline Installation

The increase in the demand of hydrocarbons in recent decades has influenced the oil & gas industry to expand their resources in order to meet demand. Offshore pipelines are considered the most efficient and economically-viable means to transport these hydrocarbons, whether crude oil or natural gas, from subsea wellheads to offshore or coastal processing facilities [2][3]. In order to create such infrastructures, various techniques have been invented and developed. In general, the so-called S-lay and J-lay pipe installation techniques are the most recurring, especially in deep-water pipeline installations [3]. The names refer to the shape to which the pipeline bends once it is lowered into the sea, as seen in Figure 1.1 for S-lay.

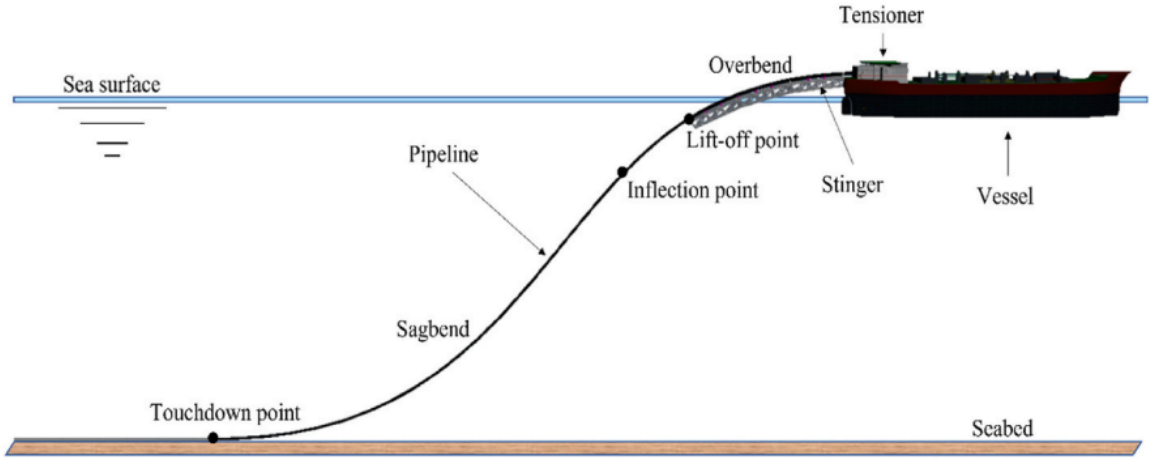


Figure 1.1: Schematic diagram of S-Lay installation process [4]

1.1.1. Installation process

Figure 1.1 depicts the standard setup of a pipeline installation - also referred to as pipelay - process: A pipelay vessel, where the pipeline is guided by the stinger on to the seabed. Pipeline installation can be divided into two main components: *the vessel* and *the pipeline*.

S-lay vessels generally consists of three sub-components for pipeline installation: *the tensioner*, which adjusts the tension in the pipeline whilst holding it in place, ensuring the pipeline behaves according to its design limits - Figure B.1a; *the stinger*, which guides the pipeline down towards the seabed, assuring that the pipeline enters the sea with a certain curvature - avoiding sharp turns in the pipeline; and *rollers*, both on the vessel and on the stinger, upholding and guiding the pipeline from within the vessel and via the stinger into sea - Figure B.1b.

The pipeline consists (but not limited to) of two sub-components: *pipe sections*, each measuring between $\pm$ 12 [m], which are continuously connected until a specified pipeline length is achieved - Figure B.1c; and *pipeline structures*, which can be integrated either between pipe sections or at the end of a pipeline, each serving a specific purpose within the entire pipeline infrastructure - Figure B.1d. In order to achieve pipelines of certain lengths required for the different projects, ranging from a few hundred meters to several kilometres, the aforementioned pipe section of $\pm$ 12 [m] are continuously connected. Consequently, to increase in length of the pipeline, the vessel will move in the opposite direction allowing the pipeline to lower into the sea and settle on the seabed.

As seen in Figure 1.1, once a part of the pipeline leaves the tensioner, it enters the *overbend* section, where its shape is induced by the rollerboxes on the stinger. The overbend section is the section from the tensioner up to the inflection point, and forms the top part of the “S” shape obtained during S-Lay installation. The *inflection point* is the point where pipe sections transition from the overbend section on to the *sagbend* section. The shape of the sagbend section is induced by the weight of the pipeline, sagging towards the seabed. This is the section starting from the inflection point, up to the touchdown point, and forms the bottom part of the “S” shape obtained during S-Lay installation. The touchdown point is the point where a pipe section first touches the seabed after passing through the sagbend section.

1.1.2. Overbend Section

As mentioned in Section 1.1.1, the overbend section spans from the tensioner up to the inflection point. Once a pipe section leaves the tensioner, it enters the overbend section upheld by rollerboxes on the vessel. These vessel rollerboxes guide the pipeline down towards the rollerboxes on the stinger.

By the pipeline being upheld by the vessel and stinger rollerboxes, its shape and curvature in the overbend section is determined by the stinger and its configuration, Figure B.2 and B.3 respectively. These are set and adjusted before the start of a project, and are not be adjusted during installation of the pipeline. Stinger geometry adjustments are typically made in order to modify and reduce strain peaks throughout the overbend section. The shape of the overbend section is therefore referred to as displacement-controlled: Controlling the shape - namely the strains - through adjusting the displacement of the pipeline by modifying stinger geometry.

1.1.3. Sagbend Section

As mentioned in Section 1.1.1, the sagbend section spans from the inflection point up to the touchdown point. The shape of the sagbend is mainly determined by the tension applied on the pipeline. By increasing the tension, either with the tensioner or by moving the vessel away from the touchdown point, the pipeline will sag less. By decreasing the tension, either with the tensioner or by moving the vessel closer to the touchdown point, the pipeline will sag more. This behaviour is depicted in Figure B.4b and B.4c, respectively. The shape of the sagbend section is therefore referred as load-controlled: Controlling the shape - namely the strains - through adjusting the tension on the pipeline.

1.1.4. Stinger

The stinger is a truss-like structure connected to the rear of the vessel, guiding the pipeline from the vessel into the sea. The name is derived from nature, due to its resemblance to the stinger of a bee or scorpion. The stinger consists of the following components: *Stinger hinges*, which connects the stinger itself to the vessel and allows for rotation in case that the stinger angle has to changed; *Rollerboxes*, upholding the pipeline and ensuring its intended shape; *Stanchions*, to which each rollerbox is connected to, and allows for height adjustment of individual rollerboxes; and *Sensors*, which are used in order to monitor different pipeline and stinger variables during pipelay. Figure B.2a shows a schematic drawing of the vessel rollerboxes. Figure B.2b shows a schematic drawing of the stinger rollerboxes. The stinger configuration, comprising of the stinger length, stinger radius, and stinger angle, is set and determined before the start of every project. These are illustrated in Figure B.3.

1.1.5. Rollerboxes

A rollerbox (RB) is a metal-box-shaped structure that forms an essential part of a stinger. The name “rollerbox” is derived from its appearance, as seen in Figure 1.2, which containing two V-shaped metal roller frames (boxes), each boxing in two roller pairs. These rollers sit in a V-shape formation, upholding the passing pipeline on two sides. Although there are a few types of rollerboxes (RBs) out there, the following explanation only discusses the type seen in Figure 1.2, as this type is the most occurring throughout this research project. Most rollerboxes are either type T1, T2, T3, or T4, where the main difference between these RB types is the distance between the rollers. Despite this length difference, the essence of their functionality, upholding the pipeline while distributing its load, stays the same.



(a) **(b)**

Figure 1.2: Rollerbox T2: a) Front-side view, not mounted on the stinger b) Top-side view, mounted on the stinger

Both boxes are connected via the main frame to each other, thus the term *rollerbox* refers not only to one box, but rather to the entire structure. For RB T1-T4, the RB contains a total of four roller pairs. Its behaviour is illustrated in Figure 1.3. The main frame itself has a rotational axis in the middle, referred to as Main Hinge. This allows both boxes to displace inversely. Additional to this behaviour, the individual boxes themselves also have a rotational axis, each referred to as RB Hinge. This RB Hinge allows for small additional movement, allowing the two rollers pairs of each box to better adjust to the pipeline it is upholding. These three hinges allow for this type of rollerbox to evenly distribute the load it endures when upholding a pipeline. The last RB of the stinger is of type T5, which contains three roller pairs and one rotational axis. These are later discussed in greater detail.

Each RB is specified and referenced using its corresponding abbreviation. The abbreviations for the RBs on the vessel are distinguished from those on the stinger by the addition of a *V* for vessel (RB V), or an *S* for stinger (RB S). To further distinguish the individual RBs, each RB is labelled with a unique number, e.g. *RB V13*. An overview of all RBs with respective numbers used throughout this report can be found in Figure B.2.

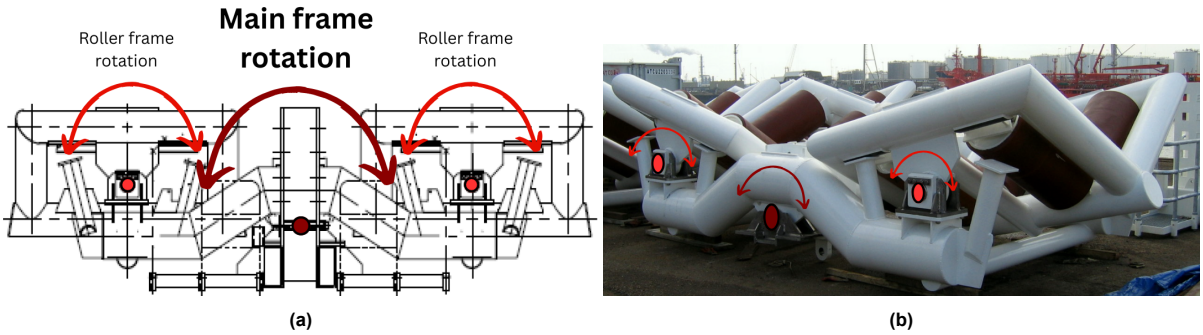


Figure 1.3: Rollerbox behaviour illustrated trough: a) Rollerbox drawing b) Real-life image

1.2. Pipeline Installation within Allseas

Allseas is a contractor company with its main activities in the global offshore energy industry, leading the way in subsea pipeline installation and offshore facilities removal. The pipelaying vessels under Allseas' fleet - *Audacia*, *Lorelay*, *Solitaire*, and *Pioneering Spirit* - all make use of the S-lay technology [5]. The S-lay technique has a better adaptability and workability around varying water depths and sea states, and has therefore a dominant position in the pipelay market [3].

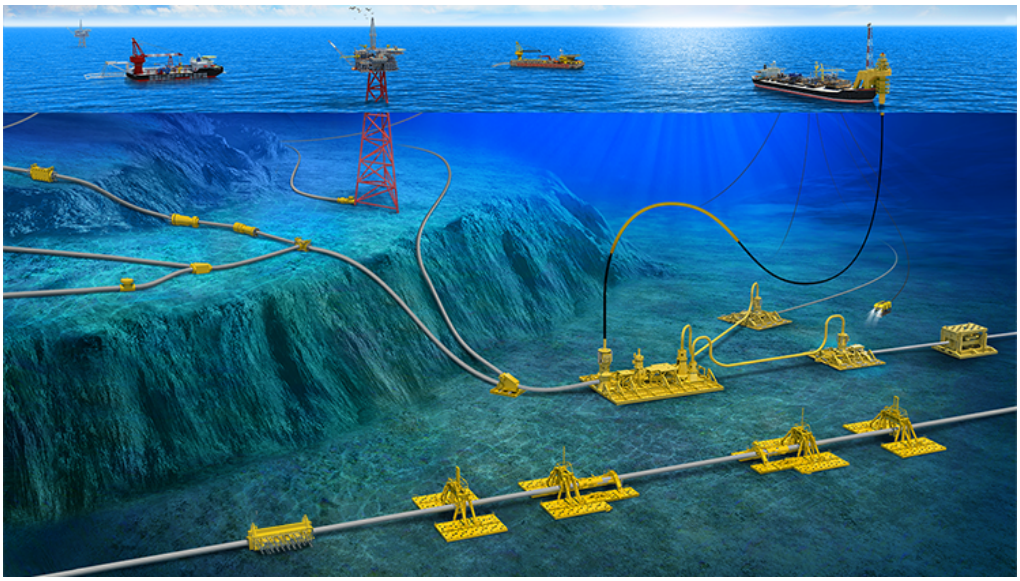


Figure 1.4: Example of a subsea pipeline infrastructure [6].

Offshore pipeline installation projects are inherently complex, characterized by non-linear dynamics and significant external influences, primarily environmental loads. These factors result in intricate component behaviours and reactions. To accurately predict how the pipeline components will respond to these external influences, extensive simulations are essential. Each pipeline installation project is unique, presenting its own set of challenges that require thorough simulation in order to ensure feasibility and success.

By conducting these simulations prior to production and installation, adjustments can easily be made if feasibility issues arise, avoiding the financial and environmental consequences incurred otherwise. Given the complexity and intricacy of these software, the information given is concise and tailored in order to maintain its relevancy to this research project, as well as to practices within Allseas.

1.2.1. Inventor

AutoDesk Inventor [7], referred to as Inventor, is a 3D Computed-Aided Design (CAD) software used for mechanical design and visualisation - Figure 1.5. Inventor provides tools for the creation of detailed drawings and designs, in order to create parts and assemblies for various projects. Inventor also provides the ability to simulate stress and strains, and get insights into their behaviour. Yet, within Allseas, Inventor is only used for detailed visualisation, from which technical drawings are developed.

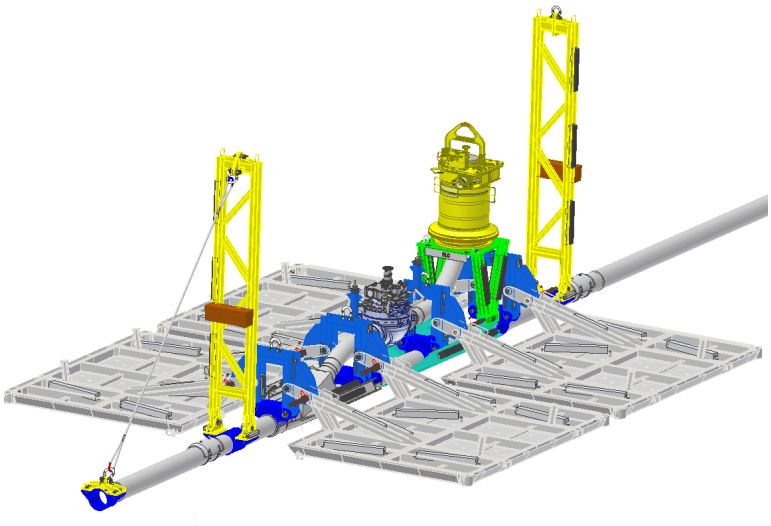


Figure 1.5: Isometric view for the inline structure for Shenandoah

1.2.2. Multi-body Modelling

In order to analyse and predict the behaviour of the pipeline during installation, global behaviour analyses of the pipeline are performed. This is done by means of multi-body modelling and analyses. Multi-body modelling entails recreating the different acting components, such as a vessel, stinger, seabed, and the pipeline, as seen in Figure 1.6. The stinger is modelled by the purple components, attached to the vessel. Furthermore, the seabed in this case is modelled as a flat horizontal plane. The pipeline is modelled by means of a lumped-mass representation, which is attached between the vessel (End A) and the seabed (End B). Having the different components, the actual behaviour of the pipeline during installation is predicted.

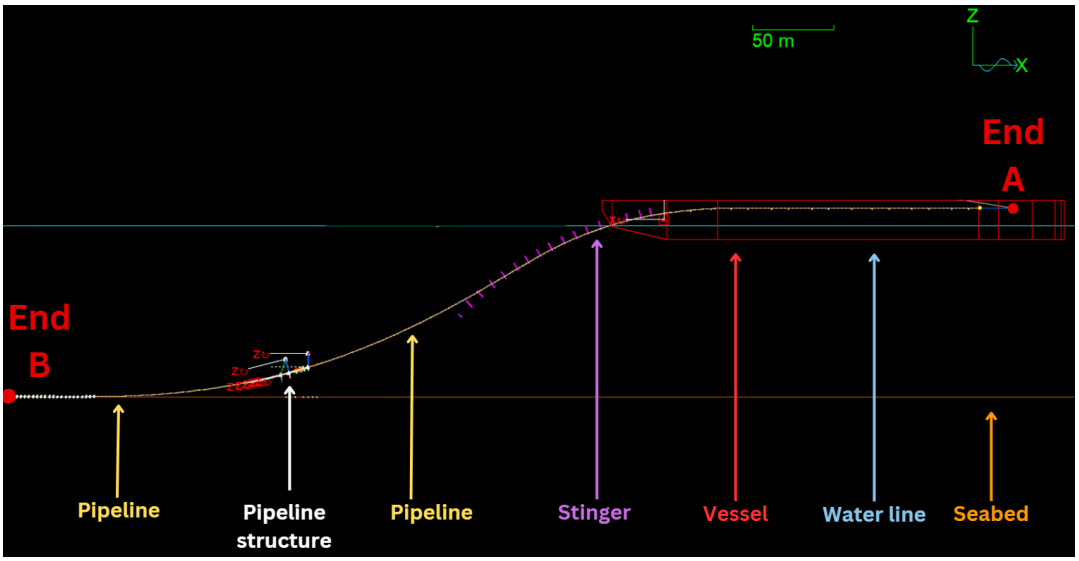


Figure 1.6: Case example in OrcaFlex: Pipelay by *Solitaire*

During this modelling, the acting components are not modelled in greater detail, as the goal is to recreate the global behaviour. The trade-off on this local level of detail of these components is the significantly small analyses converge time for global behaviour analyses. The software used within Allseas to construct these multi-body models and run respective analyses is *OrcaFlex*. OrcaFlex [8] is a Finite Element software package, primarily used in the offshore industry, in order to run complex static and dynamic analyses on (offshore) marine systems. Within Allseas, OrcaFlex is used for extensive modelling and simulation, where for pipeline installation practices, interactions of the vessel, stinger, pipeline, and pipeline structures are modelled. Allseas has standard base template models for each pipelay vessel, with their respective characteristics. Additional to this vessel, the seabed and sea are included, and environmental characteristics are edited per project and location. A pipeline is then attached to both the vessel and the seabed and its behaviour within the specified setup is analysed.

OrcaFlex offers a variety of (additional) building blocks in order for one to edit and construct their desired model. The drop-down menu for the different types of components in OrcaFlex can be seen in Figure 1.7. A brief description of the three main components used is listed below [9]:

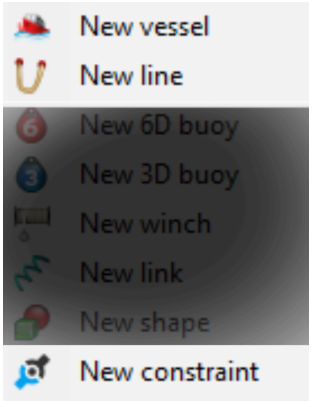


Figure 1.7: Drop-down menu of the types of components in OrcaFlex

Vessel
This component is used to create a boat-like structure in OrcaFlex, representing the vessel used for pipelay. The characteristics of this vessel component can be customized to accurately represent the specific vessel, including its size, shape, and dynamic properties. As previously mentioned, Allseas created OrcaFlex templates for each vessel, with the pre-configured vessel and its stinger.

Line
This component is used represent the pipeline, connected between end points A and B. OrcaFlex represents the pipeline by means of a lumped-mass model, as illustrated in Figure 1.8. This line can be sectioned into one or multiple line sections, representing different types of pipe sections, with respective section properties. These characteristics can vary from axial stiffness, either linear or non-linear, to inner and outer diameters. In the case that a pipeline structure needs to be included, the properties of sections at a certain location throughout the pipeline are modified accordingly to approximate its behaviour. Each pipe section can be modelled with a different amount of segments (and nodes). The greater the amount of segments, the higher the fidelity of its results. The lumped-mass model in OrcaFlex is a series of masses (nodes) are interconnected by means of massless springs (segments) like beads on a necklace, Figure 1.8. Most pipe section properties, such as diameter and mass, are “lumped” into the respective nodes accordingly. These nodes are then interconnected by means of massless segments, which only contain axial and torsional properties of the line, Figure B.5. By reducing the fidelity of such 3D pipeline (structure) using a lumped-mass model significantly reduces the convergence time of the model.

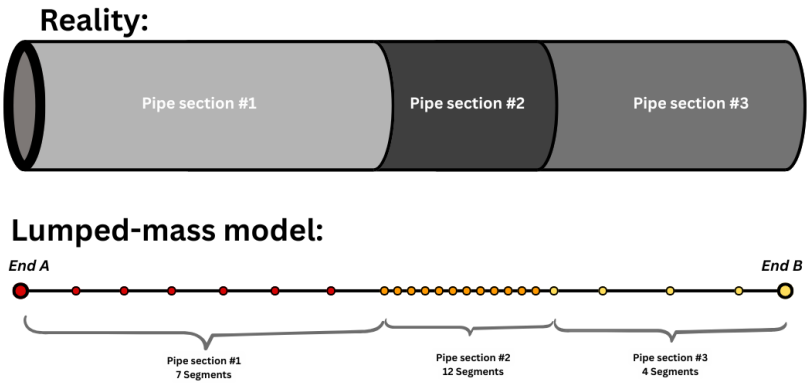


Figure 1.8: Lumped-mass model of a pipeline containing three sections

Constraint
This component allows one to restrict certain element behaviour, either fully or partially, in order to represent the real-world behaviour as close as possible. By using these aforementioned components together as building blocks, a multi-body model is achieved, serving as a close representation of an actual pipeline installation scenario. Figure 1.6 illustrates the typical appearance of a complete multi-body model within Allseas, which is similar to the depiction of S-Lay shown in Figure 1.1.

1.2.3. Non-linear Finite Element Modelling

In order to model components in greater detail and obtain high-fidelity results, 3D Non-linear Finite modelling and analyses are performed. These models are mostly used to run (local) high-fidelity analyses, in order to ensure structural integrities are maintained.

Additionally, these models are also used as benchmark. Project components such as pipelines and structures can not be created beforehand and tested to ensure feasibility. These components are therefore modelled in greater detail and used to benchmark the multi-body models and their results.

The software used within Allseas to construct these 3D Non-linear Finite Element models and run respective analyses is *Abaqus*. Abaqus CAE [10], referred to as Abaqus, is a Finite Element analysis software used to model and simulate structures and components under various conditions. It provides the possibility to create all types of parts and components, apply respective loads and constraints, in order to approximate reality as much as possible. Within Allseas, Abaqus is used to model pipeline structures, pipeline structure parts and their components, simulate their behaviour and monitor stresses and strains without having to model the complete scenario.

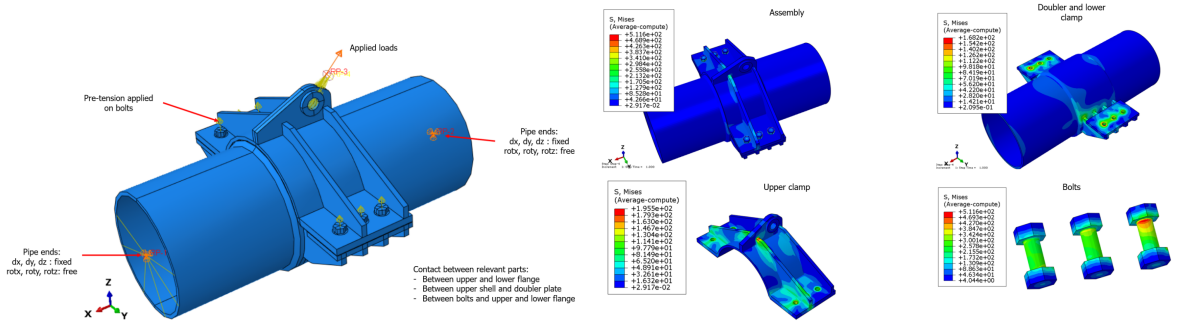


Figure 1.9: Application of Abaqus within Allseas: **a)** Setup of the Abaqus model of a part **b)** Von Mises stress results for the individual components

Figure 1.9 gives a brief example of this. Figure 1.9a shows a small section of the pipeline, where a wire clamp is attached to it by means of bolts. The simplification in this example is as follows: The wire attached to the clamp is substituted by a pulling at the eye on top of the clamp; Instead of modelling the entire pipeline, a small section of less than a meter is modelled. Both ends are set to only rotate, allowing the pipe section to behave as if it was connected to the pipeline.

Figure 1.9b shows the stresses on the different components after applying the pulling force. Additional to stresses, strains, displacements, rotations, and reaction loads can also obtained from simulations. The goal of such simulation is to get insights whether the designed structure, parts, and their components behave within their design limits for specific scenarios. If not within the limits, corresponding adjustments are made in order to reinforce the design for it stay within these limits.

Additional to modelling a structure and/or components with corresponding influences, Abaqus is also used to create models of a certain pipeline installation instances. These models are referred to as *full overbend models*, and are similar to aforementioned OrcaFlex models. A full overbend model contains: Rollerboxes, both from the vessel and from the stinger, with specific locations; Pipeline, comprising from pipe sections and respective pipeline structure to be laid; Additional features, such as forces and boundary conditions, in order to approximate realistic conditions and environments.

1.3. Report Structure

To effectively read, follow, and understand this report, it is recommended that the reader first reviews the following report structure, and subsequently proceeds through the sections as outlined below. The aforementioned sections contain information deemed crucial in order to understand the subsequent chapters. An introduction into pipeline installation (at Allseas) was provided, along with description of key procedures.

Following this, Chapter 2 will elaborate more on the current process of pipeline installation, highlighting its short-comings, in order to pave the way for the research questions.

Chapter 3 will outline the methods used to answer the aforementioned research questions. These methods are introduced and explained according to each research question.

Chapter 4 will then provide and discuss the findings resulting from the previous chapter. This provision and discussion will comprise of numeric results, with corresponding graphs and tables. Additional illustrations will be added in order to compliment the results.

Lastly, Chapter 5 will discuss the aforementioned findings which are later concluded in Chapter 6, followed by further recommendations given in Chapter 7.

The remaining pages of this report will comprise of the reference list, followed by the appendix. The appendix comprises of additional content which was deemed less relevant to the content found in the chapters, but still provides valuable information to this research project. Appendix A will contain the same contents of this report, but in a different format. Appendix B will contain additional Figures and Tables, Appendix B.1 and B.2 respectively. Appendix C will contain the Python scripts used throughout this research project.

Additionally, a glossary of terms and their definitions used throughout this report is provided in Appendix D. The reader encountering unfamiliar terminology is encouraged to consult this list for further clarification. A revision of this list prior to reading this report is suggested to non-technical readers, in order to enhance comprehension.

2

Problem Formulation

This chapter will build on the introduction given in Chapter 1. Section 2.1 will go into greater detail on the current process of pipeline installation, in order to pave the way for the next section. Section 2.2 will then highlight the current issues with the current process, in order to show the relevance of this research project. Section 2.3 will tackle the approach by providing the research question and sub-questions, followed by an overview of boundaries set and assumptions taken.

2.1. Current Process

As mentioned in Section 1.1, the pipeline itself can be split into two main components, namely *pipe sections* and *pipeline structures*. Pipeline structures can be categorized, varying from their function, to their installation method. For this research project, only pipeline structures which are installed in between pipe sections - called an Inline Structure (ILS) - are considered. This name is derived from the fact that they are installed in between pipe sections, thus lie in line with the pipeline. Other pipeline structures are usually installed by means of cranes lowering these on to the seabed. As these do not pass over the stinger thus encounter other types of loads, they are not relevant for this research project. An example of an ILS installation can be seen in Figure 2.1. Figure B.1d provides this same figure, including corresponding terminologies used throughout this report.



Figure 2.1: An ILS passing on the stinger during pipelay

Only ILSs are considered due to the fact that these experience high external loads as well as vertical displacements whilst passing through the overbend section. Since these ILSs much stiffer than pipe sections in general, sudden displacements can significantly impact the integrity of its adjacent pipe

sections, which have to be monitored and analysed during pre-installation analyses. These external loads are mainly the reaction forces of the RBs on the pipeline which is adapting to the curvature of the stinger.

As mentioned in previous sections, the global behaviour analyses of pipeline installations are performed by means of multi-body model analyses in OrcaFlex. This method is typically chosen in the offshore industry as it allows for low convergence times, usually between 20-60 seconds, despite modelling an entire pipeline installation scenario [4] [11]. The modelling of the regular pipe sections which make up (most of) the pipeline, are relatively straightforward. They are represented by means of the aforementioned lumped-mass line elements during multi-body analyses in OrcaFlex. These elements themselves will not be at the centre of attention for this research, as they are a trusted method to simplify yet closely approximate the behaviour of pipe sections [2][3].

In contrast to pipe sections, the modelling of an ILS for multi-body analyses lacks this fidelity. As these multi-body systems take into account various components and their interaction, a trade-off in localized details is made to achieve the globalized behaviour, thus detailed representation of an ILS is not feasible. As mentioned in Section 1.2, ILSs are then modelled by means of changing specific line element properties of a certain section, for multiple sections in this pipeline, Figure B.5. Due to the fact that a 3D ILS body can not be directly constructed within a pipeline by means of the aforementioned trade-off, an approximation of its behaviour is achieved. In other words, one can solely create and analyse the behaviour of this pipeline containing an ILS by means of adding, removing, and altering lumped-masses (nodes) and springs (segments) properties in such way, that the resulting behaviour of the modified nodes and segments embody an ILS. This is done by e.g. increasing the inner and/or outer diameter of a pipe section, or by increasing its bending stiffness (BS). By approximating its behaviour, the respective analyses results approximate those of the 3D non-linear finite element analyses.

2.1.1. Inline Structure Design

Before the execution of every installation, it is crucial to analyse an ILS and its effects on the pipeline. As its installation entails the ILS passing over the entire overbend section, this displacement would technically need to be replicated by displacing the ILS in significantly small increments and performing analyses at every position. However, this installation process is not mimicked in practice. The ILS is only positioned and analysed at certain position deemed critical for each specific installation scenario. Only critical positions, i.e worst-case scenarios, are analysed as this saves a lot of labour, money, and behaviour outside of these critical positions have less/to no effect on the integrity of the pipeline. Whether a certain position is deemed critical or not is scenario-dependent, and its definition within this research project is later defined in Chapter 3.

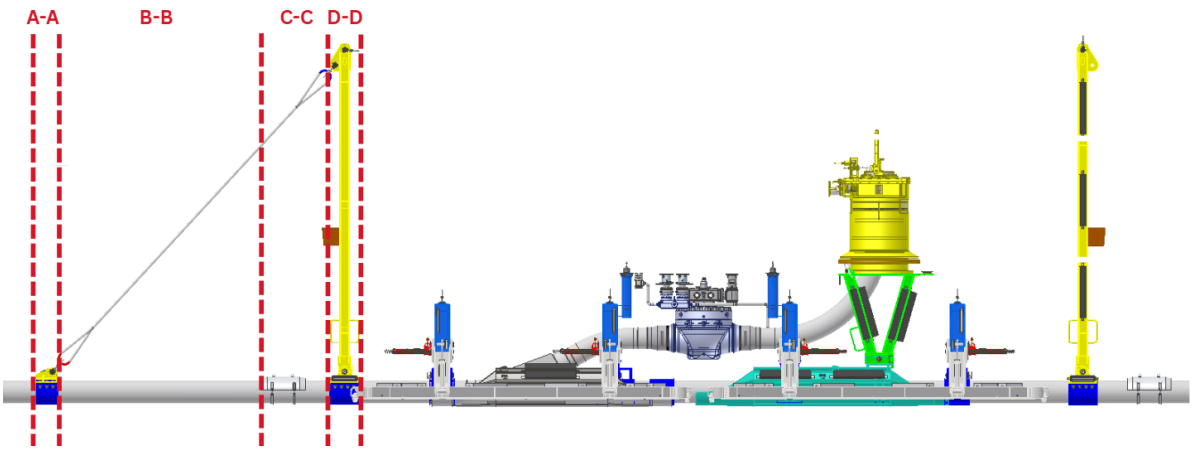


Figure 2.2: Inline Structure with the first four sections: A-A, B-B, C-C, and D-D

	Clamp	Pipe	Transition	Yoke Clamp	...
Section	A-A	B-B	C-C	D-D	...
Length $[mm]$	490	5072	630	235	...
OD $[mm]$	571,5	508	571,5	508	...
ID $[mm]$	446,18	446,18	446,18	446,18	...
Area $[mm^2]$	100166,3	46328,6	100166,3	46328,6	...
Moment of Inertia $[mm^4]$	3,29 $\cdot 10^9$	1,32 $\cdot 10^9$	3,29 $\cdot 10^9$	1,32 $\cdot 10^9$	...
...	...	...	...	...	...

Table 2.1: Section properties sheet of the inline structure: first four sections of Figure B.6, illustrated in Figure 2.2

In order to analyse the behaviour of an ILS at every critical position, pre-,post-, and mid installation simulations are built using both multi-body models in OrcaFlex and 3D non-linear finite element models in Abaqus. The 3D non-linear finite element models are mainly used to get a more in-depth (localized) analysis on (a component of) an ILS, whereas the multi-body models paints a broader picture of the pipeline installation. The roles of Abaqus, OrcaFlex, and Inventor when creating an ILS are depicted in Figure 2.3.

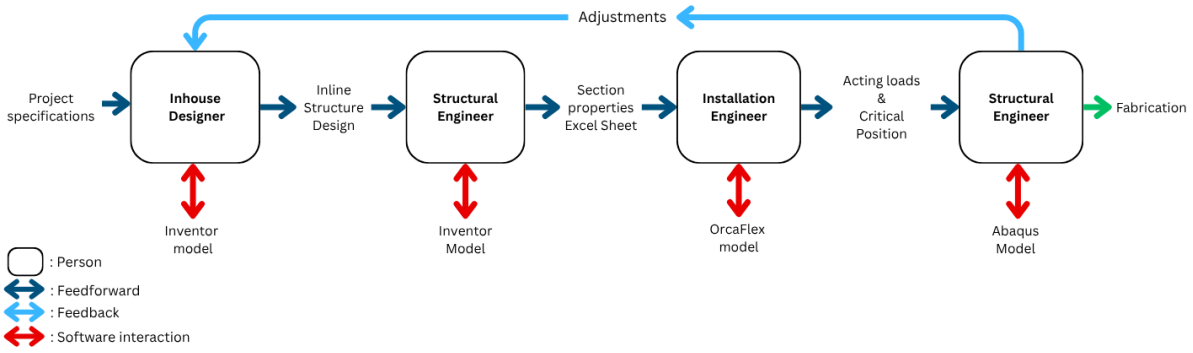


Figure 2.3: Workflow: Global steps of Inline Structure (ILS) design

1. *In-house Designer:* An ILS is initially created by designer by means of Inventor, tailored to project specification, such as its function and location.
2. *Structural Engineer:* A structural engineer then partitions this ILS into different sections, where each section has specific properties such as length and moment of inertia (MoI). These sections and their properties will then be used as pipe section properties in OrcaFlex as mentioned before. Figure 2.2 visualizes this sectioning by giving an example of an ILS with its first four sections. For each section, a few properties are extracted and placed into the Section Properties Excel Sheet. Table 2.1 shows a section of the corresponding Section Properties Excel Sheet, only showing the first five properties (table rows) of the first four section (table columns). In total, this ILS was sectioned into 20 different sections, with 17 different properties, and is later discussed in Chapter 3.
3. *Installation Engineer:* An installation engineer then recreates this ILS in OrcaFlex. This is done by altering the pipe section properties of a specific regions within the pipeline. By modifying the properties of some pipe sections within the pipeline to those from the Section Properties Excel Sheet, a representation of the ILS can be created, allowing its actual behaviour to be approximated. The installation engineer will then runs static and dynamic analyses on the pipeline containing this ILS representation. The ILS itself is very stiff when compared to the pipeline itself. One of the main objectives is to thus ensure that the ILS does not affect the integrity of the adjacent pipe sections. Additionally, the installation engineer checks whether the RB reaction force limits are not exceeded, in order to ensure that the integrity of the RBs are kept. The global behaviour of the ILS can be approximated by modelling it using different pipe sections, but this will inevitably

lack local behaviour. Therefore, detailed behaviour of the ILS has to still be studied by means of 3D non-linear finite element analyses in Abaqus.

4. *Structural Engineer:* A structural engineer then performs a more detailed structural analysis on the ILS with input from the installation engineer. By means of 3D non-linear finite elements modelling, the ILS can be created and analysed in greater detail. Once created, the input received from the installation engineer can be put to use. This input usually contains loads, such as reaction forces and bending moments where the ILS is connected to the pipeline (pipeline interface). By doing so, insights on whether the structure itself can withstand the acting loads are obtained, and required adjustments can be performed accordingly.

2.2. Current Issue

When recreating an ILS in OrcaFlex, the CAD model of the ILS has to first be sectioned in order to retrieve sectional properties. As illustrated in Figure 2.2, an ILS consists of multiple components in every direction, with a changing geometry, each having their own stiffness. Taking the first section as an example, Section *A-A* contains more than one component: Regular pipe, Pipe clamp, Clamp head, and the Clamp hook. These components have been sectioned as one, but could have also been sectioned in to two or three sections. Moving more towards the centre of the ILS in Figure 2.2, the amount of components, and their geometry and stiffness difference become more varied. Having to do this every time when modelling structures in OrcaFlex subsequently leaves the engineers with question marks regarding the actual (minimal) number of sections needed to approximate the behaviour of a structure, and whether having to do the extra work in order to have more sections is actually beneficial.

Following this tedious procedure, the question arose whether it is viable to recreate an ILS and approximate its behaviour in OrcaFlex by means of a less-tedious time-efficient method, thus simplifying the modelling method without sacrificing the quality of the results. This research project has been thought of in order to first analyse the current pipeline installation analysis method, and subsequently challenge this ILS creation approach.

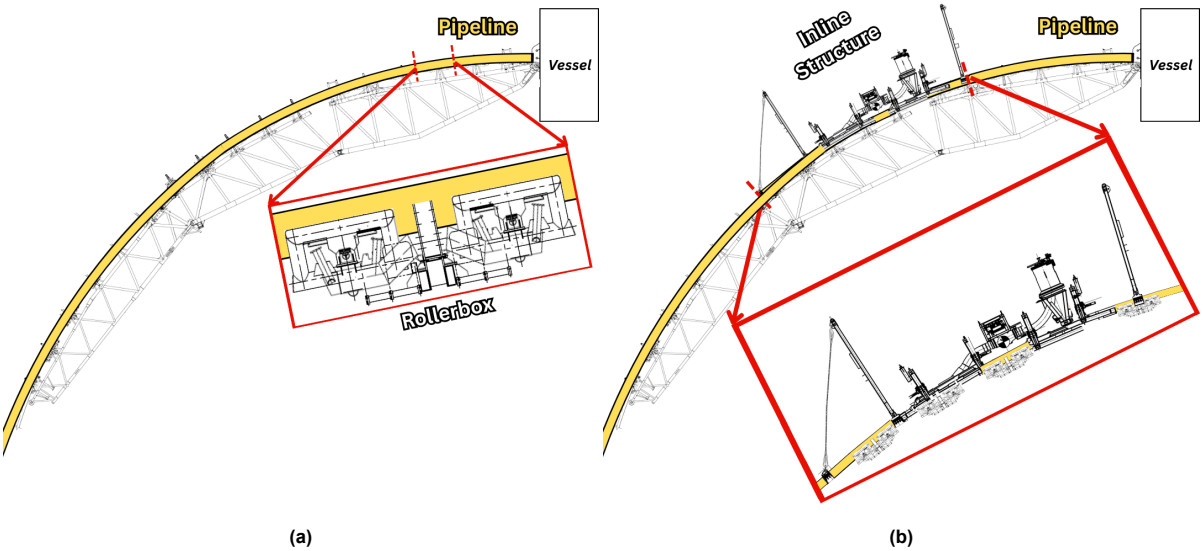


Figure 2.4: Pipeline installation: a) Homogeneous pipeline b) Pipeline with an inline structure

2.3. Approach

The objective of this research project is to provide an answer to the following research question:

How can the structural integrity of an inline structure be analysed in the overbend section?

In order to do so, the following research sub-questions are first investigated and answered:

1. *How are pipeline installations currently analysed?*
2. *What are the differences between using lumped-mass and non-linear finite-element representations for the modelling of homogeneous pipelines?*
3. *What are the critical positions of an inline structure in the overbend section?*
4. *What are the differences between using lumped-mass and 3D non-linear finite-elements representations for the modelling of inline structures?*
5. *What is a viable time-efficient method for the modelling of inline structures?*

2.3.1. Research Sub-question 1

The first sub-question addresses the overall pipeline installation analyses procedure. To tackle this question, both a global and in-depth comparison is done between the multi-body model setup and 3D non-linear finite elements model setup. While some components might be modelled differently, their difference might be deemed irrelevant for this research project.

The global comparison will look at the different components taken into account and how these are modelled. The in-depth comparison will focus on the most critical components - namely the pipeline and the ILS itself - and assess how different parts of these are modelled. For every identified difference deemed relevant for this research project, a solution is proposed in order to improve consistency between both methods.

The specifics on how this sub-question will be addressed and its results can be found in Section 3.1 and Tables B.1, B.2, B.3, B.4, B.5, and B.6.

2.3.2. Research Sub-question 2

Building on the aforementioned comparison, comparable case models will be created. This is to further illustrate the differences identified in the first sub-question, assess their relevance, and uncover any additional software-related discrepancies that may arise after the analyses are completed.

The comparison will be between a multi-body model and a non-linear finite element model of the installation of a homogenous pipeline - a pipeline where each pipe section has the same properties. Unlike for an ILS, both methods are suited for modelling homogeneous pipelines. By comparing similar pipelines, their behaviour and static analysis results can be directly compared, and possible systematic differences could be identified.

The specifics on how this research sub-question will be addressed can be found in Section 3.2, and the results can be found in Section 4.1.

2.3.3. Research Sub-question 3

Due to time constraints, only one specific ILS is considered and analysed during this research project. During actual pipeline installation, this ILS passes on majority of the RBs before it transitions from the overbend section to the sagbend section. Modelling every single position is very labour intensive, time consuming, and practically unnecessary. Following standard practices, the amount of positions modelled will be boiled down to just a (few) critical position(s).

For this research project, the critical positions are determined as to where the RB reaction force on the ILS is the highest, and where the ILS bends the most. The reasoning of these two will later be discussed in Section 3.3. In order to determine the exact location of these two positions, static analyses of the ILS over the overbend section has to be performed. This will be done through OrcaFlex, as the average converging time for one static analysis typically lies between 20-60 seconds. Additionally,

OrcaFlex allows modelling possibilities with Python through OrcFxAPI (OrcaFlex Application Programming Interface), where automated modelling, editing, execution of analyses, and data extraction can be implemented.

Through a Python script, the ILS will be displaced over the stinger recreating its movement during actual installation. For each position, a model will be created. Static analysis will be performed for each model, and the results from all models will be combined and exported into an Excel sheet. From this Excel sheet, the position for both the highest RB reaction force on the ILS and where the ILS bends the most, will be determined.

The specifics on how this research sub-question will be addressed can be found in Section 3.3 and the results can be found in Section 4.2. The Python script used can be found in Appendix C.

2.3.4. Research Sub-question 4

Following the aforementioned two positions determined in OrcaFlex, the ILS in Abaqus will be placed accordingly. This results in four models, an Abaqus and OrcaFlex model for the position with the highest RB reaction force on the ILS, and an Abaqus and OrcaFlex model for the position where the ILS bends the most. The results of static analyses for the Abaqus and OrcaFlex models will be compared respectively, in order to get insights into the current fidelity of the ILS defined in the multi-body model in OrcaFlex.

The specifics on how this research sub-question will be addressed can be found in Section 3.4, and the results can be found in Section 4.3.

2.3.5. Research Sub-question 5

Following the aforementioned analyses on homogeneous pipelines and inline structures, an alternative time-efficient method for creating an ILS is proposed, tested, and discussed. The idea of this method is to provide the previously-used ILS, which is recreated in a multi-body model, with a lower fidelity. After running a static analysis on this ILS, the results are compared to the previous models in order to assess the trade-off between modelling and result fidelity.

The specifics on how this research question will be addressed can be found in Section 3.5, and the results can be found in Section 4.4.

2.3.6. Research Question

After concluding the five aforementioned research sub-questions, the main research question itself can be tackled. For this, a list of steps are described and illustrated using a flow chart, in order to explain how the structural integrity of an inline structure can be analysed. Depending on the results of the last research question, these steps will entail an alternative time-efficient method for the creation of an inline structure than used currently.

The specifics of these steps can be found in Section 4.5.

2.3.7. Research Scope

The following outlines the key boundaries set and assumptions taken at the start of this research project, in order to shape its scope. Some of these have been mentioned in the preceding sections, and were added to the list for overall clarity. Respective reasoning is denoted below each corresponding assumption:

1. The pipelay vessel *Solitaire* has been chosen for this research.

   Solitaire has been chosen as pipelay vessel due the availability of previous research done, e.g. for its stinger [12] and its rollerboxes [13] [14] [15].

2. Inline Structures (ILSs) are the only type of structures of interest.

   ILSs are installed in between pipe sections in the pipeline, and have direct impact on the integrity of the pipeline. Other types pipeline structures are installed by means of cranes, thus are not of interest for pipeline installation.

3. Only one ILS will be considered and analysed: the ILS of choice is that of the previous project *Shenandoah*.

   Due to time constraints for this research project, only one ILS can be considered and analysed. The ILS of choice is that of the previous project Shenandoah, with corresponding vessel Solitaire. The choice for this ILS is based on the fact that the corresponding vessel had to be Solitaire - following assumption #1. When comparing the different ILSs installed by Solitaire, Shenandoah offered a well-developed set of documentations and simulations in order to build upon.

4. The stinger, stinger components, and rollerboxes are assumed to be fully rigid.

   The stinger and its components are assumed to be rigid, as proposed by previous research [12] [13] [14] [15] done for the respective components.

5. Other components and their properties, such as vessel and stinger dimensions, are assumed to be accurate.

   The main focus of this research project is to analyse ILSs. Other components and their dimensions, such as vessel and stinger dimensions, are assumed to be measured correctly. These have been set in the past by other employees, and have been deemed correct ever since. Additional research into these will not provide any relevant outcome, especially not for the goal of this research project. The measurements within the models themselves are checked in order to assure and maintain consistency.

6. Only the overbend section is of relevance. The overbend section starts at the 13th rollerbox on the vessel (RB V13), and can span up to the 17th rollerbox (RB S17) of *Solitaire*, depending on the shape of the pipeline.

   The pipeline is upheld by RBs on both the vessel and the stinger in the overbend section, before entering the sagbend section. ILSs differ in geometry and stiffnesses when compared to pipe sections, leading to a different behaviour when passing on these RBs. This behaviour is influenced by various factors, and can have significant effects on the integrity of the pipeline. The behaviour of the pipeline in the sagbend is mainly controlled by controlling the tension, making it irrelevant for this research. The length of the pipeline considered the overbend section is determined by the position of the inflection point, as seen in Figure 1.1. The inflection point is the point where the overbend section transitions into the sagbend section. Figure 1.1 shows that the inflection point takes place after the stinger, thus the overbend section covers the entire stinger - RB V13 up to RB S17. In the case that the inflection point lies on the stinger, i.e. the pipeline has no contact with the end of the stinger, the overbend section is reduced up to the last rollerbox in contact with the pipeline.

7. The focus will be only on static analyses.

   Only statics have been chosen as it is assumed that these analyses provide enough insights into the fidelity of the ILS representation. Looking at dynamics will propose an additional set of challenges, increasing the amount of time needed to complete this research project.

8. The full overbend models using non-linear finite elements in Abaqus will be used in order to benchmark the multi-body models in OrcaFlex.

   When analysing pipeline structures, simulation models are not physically pre-built. Instead, corresponding models are developed using non-linear finite elements beforehand, incorporating specific boundaries, contacts, and loads to approximate real-world conditions. Modelling by means of these elements allow for high fidelity, and are considered to be the most accurate representation of reality achievable. These are therefore used as benchmark

9. The interaction between the pipeline and the rollers of the rollerboxes is assumed to be frictionless.

   In reality, these rollers spin as the pipe sections passes on them. Their bearings are meticulously maintained, and spin freely. Additionally, these rollers can be seen spinning freely for a few seconds after losing contact with the pipeline. Therefore, the interaction between the pipeline and the rollerboxes is assumed to be frictionless.

10. The water density is assumed to be constant in the vertical direction.

    This is an assumption used for all multi-body models, and is also adopted for this research project.

11. The installation of an ILS is simplified to be a 2D problem as seen in Figure 2.5, where torsion is neglected.

    This is an assumption used frequently for ILS installation analyses. ILSs have buoys which enforces them to stay upright, avoiding sudden twists and reduces torsional loads. Additionally, assuming the installation to be a 2D problem and neglecting torsion simplifies the mathematical problem, limiting the scope of this research project, thus limiting the time needed for its completion.

12. The strains within the ILS itself are deemed irrelevant, and only the effects of the ILS on adjacent pipe sections are relevant for this research project.

    The main goal of pipeline installation analyses is to capture and analyse the global behaviour of the pipeline. In the case of installation of an ILS, pipeline installation analyses performed using multi-body models are thus done to check the global behaviour of the pipeline when containing an ILS. The multi-body models provide the ability to perform global analyses in a small amount of time, but have therefore a trade-off in the ability to model detailed structures and in the ability to capture localized results. ILS themselves are very stiff when compared to the pipeline itself. Such a change in stiffness throughout the pipeline, especially when experiencing high external loads, can have direct impact on its integrity and has to be analysed in order to ensure that this pipeline integrity is kept. When comparing multi-body models and non-linear finite element models for pipeline installation, the results of the ILS itself is not compared due to inherent lack of detail for ILS modelling within multi-body models compared to non-linear finite elements. The structural integrity of the ILS itself is assessed by means of additional localized analyses using 3D non-linear finite elements, using interface loads provided from the global analyses as seen in Figure 2.5.

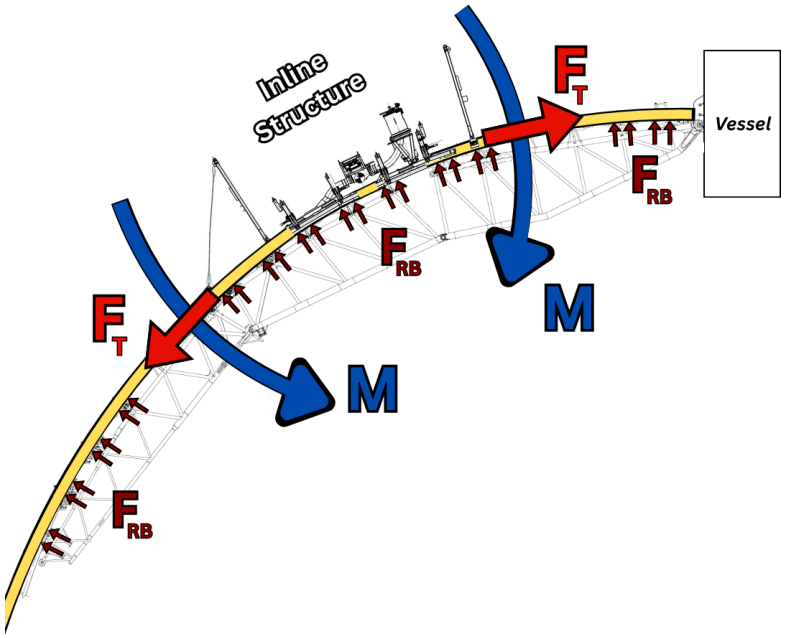


Figure 2.5: Visualization of the simplified inline structure installation problem - Assumption 11

3

Method

This chapter will provide the methods used in order to answer the aforementioned research questions. Section 3.1 will dive into how pipeline installations are analysed, tackling the first research sub-question. Section 3.2 will give a breakdown of both setups of the multi-body model in OrcaFlex and the non-linear finite element model in Abaqus for a homogeneous pipeline. The following two sections will tackle the comparison when having an ILS within the pipeline: Section 3.3 will give a breakdown on how the critical positions of the ILS are determined, and Section 3.4 will give a breakdown on the comparison of the lumped-mass model and the 3D non-linear finite element model of the ILS. Section 3.5 will give a breakdown on an alternative time-efficient method for the modelling of ILSs. Section 3.6 briefly discusses Python scripts used to automate steps in Abaqus, in order to significantly reduce modelling time.

Tables B.1 and B.3 contain a list of a global and in-depth comparison between pipeline installation analyses for the case of a homogeneous pipeline. Tables B.5 in Appendix B contains a list of an in-depth ILS comparison. Tables B.2, B.4, and B.6 contain the respective differences and possible solutions for the aforementioned comparisons, respectively. Appendix C contains all Python scripts used throughout this research project. Appendix D contains a list of definitions of terms used throughout this chapter.



Figure 3.1: Top view of the stinger during pipeline installation

3.1. Pipeline Installation Analysis

Before any pipeline installation can take place, thorough installation analyses are performed in order to assess the behaviour and integrity of the pipeline and the feasibility of its installation using a certain pre-determined setup. Each project has its own characteristics such as water depth, pipeline structure types and the amount of structures to be installed. Following these characteristics, an installation setup is pre-determined. This setup entails determining variables such as the pipelay vessel(s), the pipe-section type, the applied pipeline tension, and stinger configuration. Once pre-determined, each installation setup with respective variables is replicated in a multi-body model in order for its global behaviour to be analysed. The results illustrate whether the pre-determined installation setup is feasible or if variables need to be modified accordingly: Having high sagbend pipeline strains can be reduced by applying more tension on the pipeline, yet high overbend strains can be reduced by changing the stinger configuration.

OrcaFlex provides the ability to construct multi-body models, in order to analyse the global behaviour of the pipeline during installation and assess its feasibility. As seen in Section 1.2.2, these multi-body models replicate the pipeline installation scenario containing the sea, seabed, pipeline, vessel, and its stinger. The pipeline itself is modelled by means of a lumped-mass model, and is represented by means of nodes interconnected by springs - Figure 1.8. For homogeneous pipelines, close approximation of its actual behaviour using these 1D elements can be achieved. However, sudden changes in diameter, e.g. when modelling an ILS, can lead to discrepancies in the results. Despite discrepancies that might arise, these multi-body models are inherently used due to their ability to significantly low convergence time.

Abaqus provides the ability to construct detailed models of component(s) by means of using non-linear finite elements. As mentioned in Section 1.2.3, this can be to either analyse local or global behaviour. For pipeline installation analysis, non-linear finite element models are created to obtained higher-fidelity behaviour of the pipeline, and are used to benchmark some behaviour results obtained from the multi-body models. By means of using more complex elements during modelling, the result fidelity increases; however, so does the computational intensity and consequently the convergence time.

In order to understand how these full overbend models are set up using both the multi-body model in OrcaFlex and the non-linear finite element model in Abaqus, a comparison analysis is done. This will highlight model similarities and differences. The differences per component are highlighted, and eventual solutions to minimise these differences are proposed. Some components might not be modelled identically, yet this difference might not be relevant for the scope of this research project. The extended tables showing pipeline comparison, and in-depth ILS comparison can be found in B.1 & B.2, B.3 & B.4, and B.5 & B.6 respectively.

3.1.1. Comparison Results

The following is a condensed version of Table B.1, B.3, and B.5, showing the only components relevant to the subsequent conclusions in Section 3.1.2. The vessel modelled in OrcaFlex has representations of the additional support rollers which uphold the pipeline horizontally whilst passing towards the tensioner and the first RB as in seen Figure B.1a and B.1b. By replicating these support rollers in OrcaFlex, the pipe sections following the first node - which is fixed to the environment - has room to settle horizontally before coming into contact with the RBs. On the other hand, these support rollers are not modelled in the full overbend models in Abaqus. The first node of the pipeline sits right above the first rollerbox as illustrated in Figure 3.2. By slightly missing the correct coordinates of this first node, the pipeline can be either too close or too far from the first RB, resulting in inaccurate behaviour over the following RBs. To allow the pipeline to also settle horizontally in Abaqus and avoid these boundary condition effects, these additional support rollers will be added and discussed in Section 3.1.2 for further analyses.

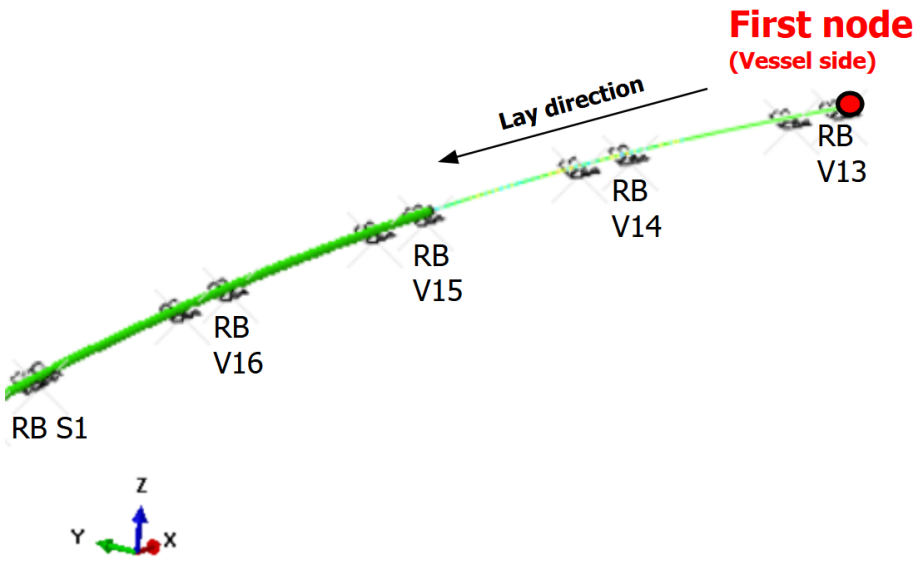


Figure 3.2: Full overbend model setup in Abaqus, before adding support rollers

Component	Abaqus	OrcaFlex
Vessel	Not modelled.	Modelled, only used as attachment points for rollers supports, vessel RBs, the stinger, and one end of the pipeline.
Rollerboxes		
Vessel rollerbox body	Modelled as one rigid body per rollerbox, with respective dimension and declination angle.	Modelled as eight rollers, lying in pairs in a V-shape configuration with V angle of 150 degrees (between rollers). The rollers are fixed on top of two separate flat supports, where each support has two roller pairs.
Stinger rollerbox body	Modelled as one rigid body per rollerbox, with respective dimension and a pre-fixed declination angle.	Most stinger RBs have eight rollers, lying in pairs in a V-shape configuration with V angle of 120 degrees (between rollers). The rollers are fixed on top of a flat support, where some RBs have two flat supports with two roller pairs, and some have one flat support with all four roller pairs.
Degree of Freedom	The RBs are modelled as on rigid body, and are completely pre-fixed. There coordinates and declination angles are set and fixed during the creation of the model. The coordinates and declination angle is set via a reference point in the middle of each rollerbox body, where all corresponding roller pairs modelled in a straight line. All roller pairs will thus have the same declination angle and will not translate nor displace during analyses. During modelling for some previous project, the rollerbox which caused the highest contact stress on the structure has been modified to allow just rotation in the plane of bending in order to alleviate these contact stresses. To do so, a 1D rotational spring is added with an arbitrary spring stiffness with a rotational freedom in the plane of bending.	All RBs have rotational freedom in the plane of bending. The amount of rotational points (hinge) depends on how the behaviour of a rollerbox is in reality. Most RBs are modelled with two supports, where these RBs have three hinges. The main hinge in the middle of the supports allow for these to translate synchronously. Additionally, each support will have its own hinge, allowing each support to rotate individually. The RBs modelled with only one support will have just a main hinge, placed in the middle of this support.

Table 3.1: General comparison between Abaqus and OrcaFlex, for static simulation of a pipeline installation

Additionally, OrcaFlex provides the ability to model the RB geometries and respective behaviours very close to actuality. Figure 1.3 illustrates how the geometry and behaviour of this type of RB is in reality. Figure 3.3b illustrates how its geometry and behaviour is modelled in OrcaFlex, replicating the three rotational axis (hinges). This allows for the RBs in OrcaFlex to also evenly distribute the load of the pipeline. Figure 3.3c illustrates how this is done in Abaqus. Here, an RB is modelled as a single rigid body with a reference point (RP) in the middle of it. The RBs are fixed at a certain location with a certain angle before any analyses are executed. Having the RB behave as a single rigid body does not coincide with its actual behaviour. Additionally, having one fixed rigid body results in the RBs not distributing the load of the pipeline evenly, causing for irregular contact behaviour between the RB and the pipeline. To improve the distribution of the load of the pipeline in Abaqus, each RB will be modelled as two separate rigid bodies, and discussed in Section 3.1.2, for further analyses.

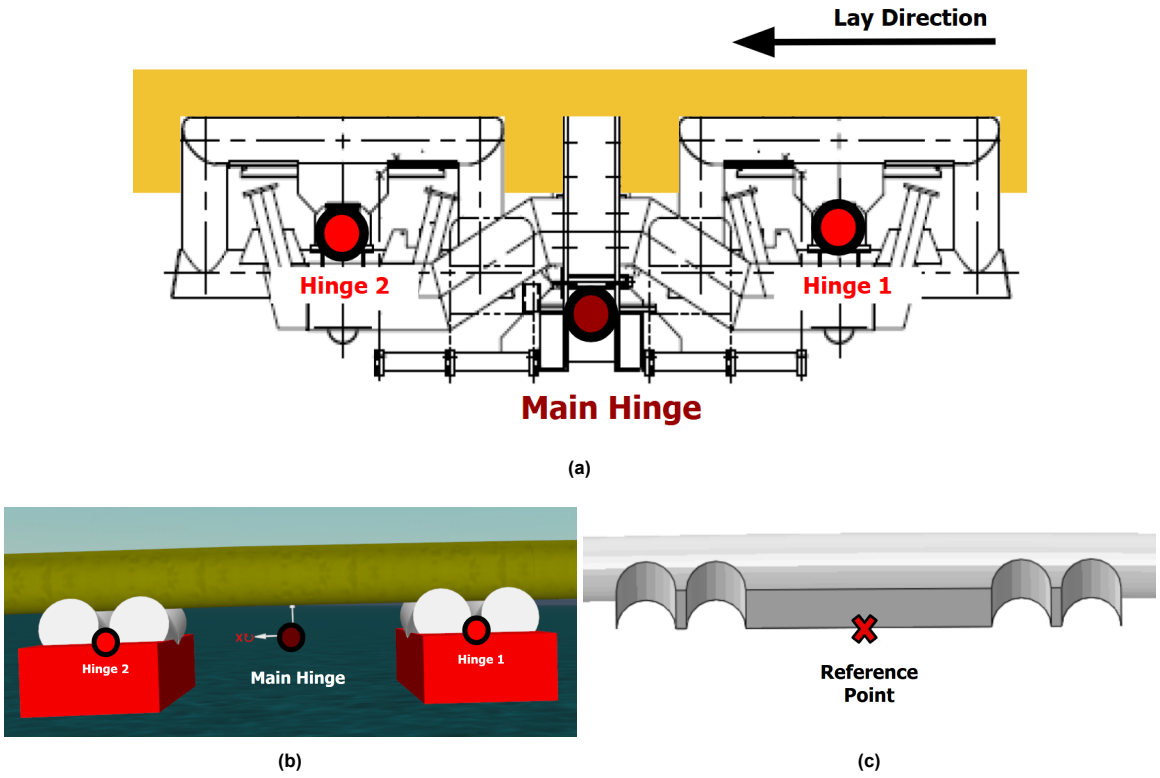


Figure 3.3: a) Rollerbox drawing b) Rollerbox modelled in OrcaFlex c) Rollerbox modelled in Abaqus

3.1.2. Setup Modifications

Following the aforementioned discrepancies in Abaqus, the subsequent two modifications to the Abaqus setup are done:

1. Support rollers are added before RB V13.
2. Rollerbox geometry is modified, from one rigid body to two.

Support Rollers

The first node of the pipeline is fixed to the environment at a certain location, and following nodes of the pipeline start coming into contact with the following RBs. In the case that the first node is located right before and/or above RB V13, having a slight discrepancy in its coordinates will cause for the contact to be inaccurate - if there is even contact - affecting following pipeline-rollerbox interactions. Therefore, as discussed, three support rollers are added preceding RB V13 as illustrated in Figure 3.4. These support rollers will minimize the boundary condition effects on the results of the static analyses by allowing the first few meters of pipeline to settle, ensuring pipeline-rollerbox interaction from RB V13 onwards.

The amount of support rollers is chosen as three after looking upon previously-used OrcaFlex models. The vertical displacements of the pipeline in these models did not take place until after the second-to-last support roller. Modelling only two support rollers, right before vertical displacements start occurring, might still result in slight boundary condition effects. It is therefore decided to model the (minimal) amount of three support rollers. Naturally, more support rollers can be modelled. However, the more support rollers modelled, the bigger the mathematical problem becomes and thus the larger the amount of convergence time.

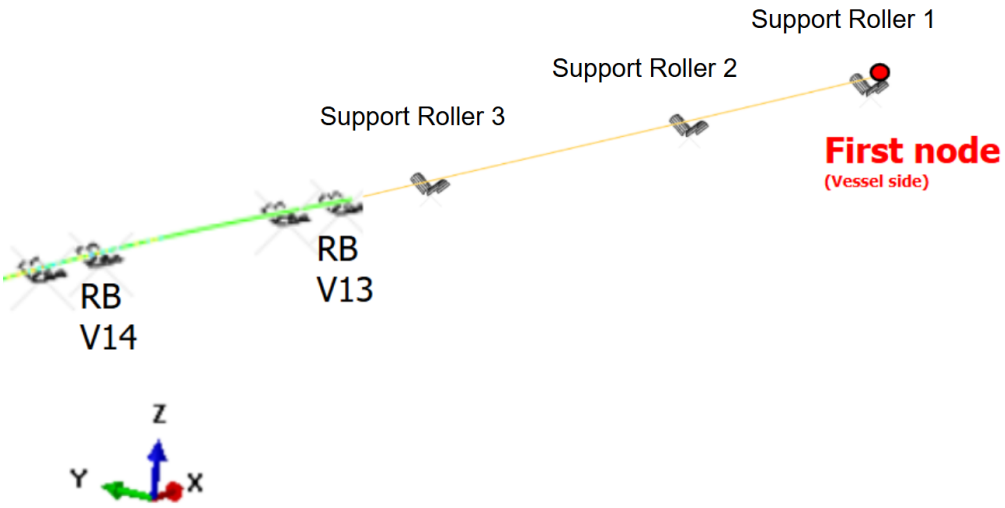


Figure 3.4: Full overbend model setup in Abaqus, after adding support rollers

Modified Rollerboxes

Until now, RBs have been modelled by means of one rigid body at a certain location with a certain angle. The angle of a RB at the Main Hinge in OrcaFlex would be replicated in Abaqus by rotating the rigid body around its RP accordingly. Yet, the relative angle of the Hinge with respect to its Main Hinge would not be accounted for due to the entire RB being one rigid body, as previously seen in Figure 3.3c. Therefore, Abaqus RBs had a discrepancy with respects to reality and OrcaFlex. To solve this, the rigid body is separated into two, making each roller frame its own rigid body with its respective RP. Having its individual RP allows for the Hinge declination angles to be replicated. Figure 3.5b now illustrates this modification, and how it compares to that of OrcaFlex in Figure 3.5a. The corresponding method to set these RBs at identical locations and angles will be discussed in Section 3.2.2.

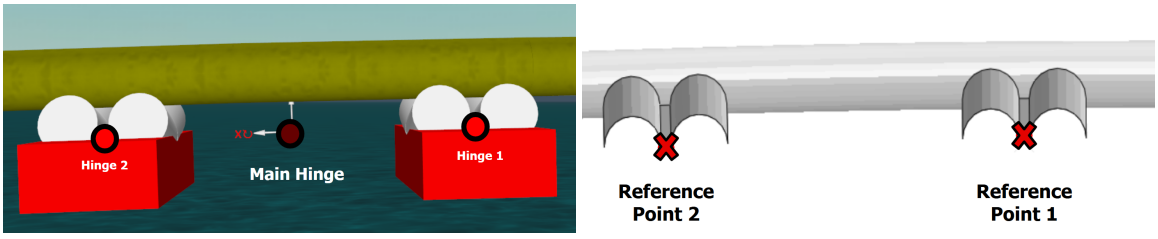


Figure 3.5: a) Rollerbox modelled in OrcaFlex c) Rollerbox modification in Abaqus

3.2. Pipeline Comparison

The aforementioned comparison showed how both setup in Abaqus and OrcaFlex model certain components, and how certain differences were mitigated.

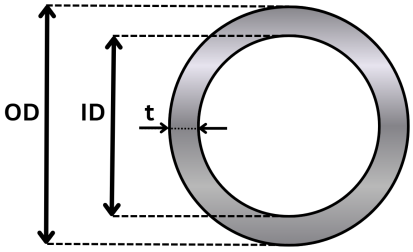


Figure 3.6: Cross section of pipe section

To further gain understanding of the differences in the results of static analyses between a pipeline constructed using non-linear finite elements in Abaqus and using a lumped-mass model in OrcaFlex - and assess whether these differences are relevant - a comparative analysis is conducted between a homogenous pipeline case model. To ensure the validity and relevance of this comparison, both model setups have to be identical. This includes matching parameters as: Rollerbox dimensions; Stinger radius; Material; Environment; Boundary conditions; and loads. By aligning these parameters, any observed discrepancy can be attributed to the inherent difference between the non-linear finite elements and the lumped-mass model, rather than to the setup itself.

The details listed in Table 3.2 correspond to the pipeline case studied, which variables originates from the project Shenandoah.

Variable	Value	Unit
Pipe section OD	508	$[mm]$
Wall thickness	$30,98$	$[mm]$
Water depth	1326	$[m]$
Bottom Tension	1200	$[kN]$
Water density	1025	$[kg/m^3]$

Table 3.2: Values of the variables used in the case studied, for both Abaqus and OrcaFlex

Defining a pipeline with the same pipe section in Abaqus and OrcaFlex differs slightly. Some of these pipe section properties are defined identically, e.g. outside diameter. Other properties have to be modified in order to be used. An example of this is material density. Abaqus has a specific material density input, but OrcaFlex does not. OrcaFlex rather has an input for the mass per unit length. Therefore, the density has to be multiplied by the area of the cross section in order to have a quantity for the mass per unit length. On the other hand, Abaqus does not automatically factor in buoyancy, thus the density of the pipeline that is submerged has to be modified accordingly. This is usually done by creating two material with different densities, a dry and submerged variant.

The following properties are input for the pipe sections in Abaqus: Outside diameter (OD); Wall thickness (t_w); Dry steel density ($\rho_{steel,dry}$); Submerged steel density ($\rho_{steel,subm.}$); Young's modulus (E); Poisson ration (ν). The corresponding values can be seen in Table 3.3a.

The following properties are used as input for the pipe sections of the lumped-mass pipeline: Outside diameter (OD); Inner diameter (ID); Mass per unit length (m_L); Bending stiffness (BS); Axial stiffness (AS); Poisson ration (ν). The corresponding values can be seen in Table 3.3b. Despite the assumption of not including torsional effects for this research project, the calculation of torsional stiffness is still provided as this is still a material input in OrcaFlex. This stiffness is filled, but the assumptions of not including torsional effects is maintained during static analyses. This is done by turning off the "Include torsional effects" feature of OrcaFlex.

Abaqus		
Property	**Value**	**Unit**
OD	0,508	$[m]$
t_w	0,03098	$[m]$
$\rho_{steel,dry}$	7850	$[kg/m^3]$
$\rho_{steel,subm.}$	3375,9	$[kg/m^3]$
E	2,07 $\cdot 10^{11}$	$[Pa]$
ν	0,3	$[-]$

(a)

OrcaFlex		
Property	**Value**	**Unit**
OD	0,508	$[m]$
ID	0,46604	$[m]$
m_L	364,4496	$[kg/m]$
BS	See Figure B.24	$[Nm^2]$
AS	9,61 $\cdot 10^9$	$[N]$
TS	2,11 $\cdot 10^8$	$[Nm^2]$
ν	0,3	$[-]$

(b)

Table 3.3: Pipe section properties: a) For the Abaqus model b) For the OrcaFlex model

The bending stiffness (BS) consists of a bending moment-curvature relationship, where data points of this relationship curve serve as input for the material. How this curve is obtained is later discussed in Section 4.1.1.

The axial stiffness (AS) found in Table 3.3b is calculated using the following equation:

$$AS = E \cdot A \tag{3.1}$$

where E is the Young's modulus, found in Table 3.3a, and A is the cross-sectional area of this pipe section, calculated using Equation 3.2. The axial stiffness input is given in [N]. This allows for OrcaFlex to automatically adjust the actual axial stiffness [$\frac{N}{m}$] for segments, which simplifies changing segment lengths.

The cross-sectional area (A) used above is calculated using the following equation:

$$A = \frac{\pi}{4} \cdot (OD^2 - ID^2) \tag{3.2}$$

where OD is the outside diameter and ID is the inside diameter of the respective pipe section, found in both Table 3.3a and Table 3.3b

The torsional stiffness (TS) found in Table 3.3b is calculated using the following equation:

$$TS = G \cdot J \tag{3.3}$$

where G is the modulus of rigidity and J is the polar moment of inertia of this pipe section, calculated using Equation 3.4 and Equation 3.5 respectively.

The modulus of rigidity (G) used above is calculated using the following equation:

$$G = \frac{E}{2(1+\nu)} \tag{3.4}$$

where E is the Young's modulus and ν is Poisson's ratio, found in both Table 3.3a and Table 3.3b.

The polar moment of inertia (J) used above is calculated using the following equation:

$$J = \frac{\pi}{32} \cdot (OD^4 - ID^4) \tag{3.5}$$

where OD is the outside diameter and ID is the inside diameter of the respective pipe section, found in both Table 3.3a and Table 3.3b

3.2.1. Lumped-mass Pipeline Model

For the OrcaFlex model, the following is modelled: Solitaire vessel; 18 support rollers; All four vessel RBs; All 17 stinger RBs; The pipeline; The sea and seabed. Figure B.7 shows the entire model, showing all aforementioned components. Figure B.8 zooms into the vessel, rollers, and RBs attached to it. All of these components are broken down (in greater detail) in the following parts.

Solitaire:
The vessel is present in the OrcaFlex model but does not contribute to the simulation in any way as only static analyses are being performed for this research project. It only serves as an attachment point for the pipeline. Therefore, no further details are discussed.

Support Rollers:
The support rollers uphold the pipeline horizontally along the vessel. These are modelled as a pair of rollers in a V shape, at an angle of $120°$ between rollers. Their properties can be seen in Figure B.17a. Despite having a normal and shear stiffness of each $70 \cdot 10^6$ $[N/m]$, no friction coefficient is set. Therefore, the interaction with the pipeline will be frictionless, as mentioned in Section 2.3.7.

Vessel rollerboxes:
The vessel RBs uphold the pipeline in the transition from the vessel to the stinger. These have three rotating points (hinges), illustrated in Figure B.9a, where the middle hinge is referred to as the main hinge. These hinges allow the RBs to rotate and adapt to the pipeline, mimicking their behaviour in reality. Their properties can be seen in Figure B.17b. Despite having a normal and shear stiffness of each $70 \cdot 10^6$ $[N/m]$, no friction coefficient is set. Therefore, the interaction with the pipeline will be frictionless, as mentioned in Section 2.3.7.

Stinger rollerboxes:
The stinger RBs uphold the pipeline from the vessel, down towards the sagbend section. These also have three rotating axis (hinges) except for the last rollerbox, RB S17. This RB S17 is smaller since it only contains three roller pairs and therefore has only one hinge as illustrated in Figure B.15a. Figure B.10a illustrates the three hinges, where the middle one is referred to as the main hinge. These hinges allow the RBs to rotate and adapt to the pipeline, also mimicking their behaviour in reality. Their respective properties can be seen in: Figure B.17c for RB S1-S9; Figure B.17d for RB S10; Figure B.17e for RB S11; Figure B.17f for RB S12-S14; Figure B.17g for RB S15; Figure B.17h for RB S16; and Figure B.17i for RB S17. Despite all RBs having a normal and shear stiffness of each $70 \cdot 10^6$ $[N/m]$, no friction coefficient is set. Therefore, their interaction with the pipeline will be frictionless, as mentioned in Section 2.3.7.

Pipeline:
The pipeline is constructed using a lumped-mass model as mentioned in Section 1.2.2, and its dimensions are given in Table 3.3a. Specifics such as total length, sections, section lengths, and target segment lengths are illustrated in Figure 3.7. The pipeline in constructed using 13 sections, and can be split up into three different parts: The start of the pipeline, with a total length of $206,43$ $[m]$, sections #1 and #2; The middle of the pipeline, with a total length of 200 $[m]$, sections #3 to #12; And the end of the pipeline, with a total length of $3035,5$ $[m]$, section #13.

Sections 13 Total length = 3441,93m

No.	Line type	Section length (m)	Expansion factor	Target segment length (m)	Number of segments	Clash check	Cumulative values Length (m)	Cumulative values Segments
1	20-inch 30,98mm WT OEP	20,0	~	20,0	1	☐	20,0	1
2	20-inch 30,98mm WT OEP	186,43	~	2,5	75	☐	206,43	76
3	20-inch 30,98mm WT OEP1	20,0	~	0,2	100	☐	226,43	176
4	20-inch 30,98mm WT OEP2	20,0	~	0,2	100	☐	246,43	276
5	20-inch 30,98mm WT OEP3	20,0	~	0,2	100	☐	266,43	376
6	20-inch 30,98mm WT OEP4	20,0	~	0,2	100	☐	286,43	476
7	20-inch 30,98mm WT OEP5	20,0	~	0,2	100	☐	306,43	576
8	20-inch 30,98mm WT OEP6	20,0	~	0,2	100	☐	326,43	676
9	20-inch 30,98mm WT OEP7	20,0	~	0,2	100	☐	346,43	776
10	20-inch 30,98mm WT OEP8	20,0	~	0,2	100	☐	366,43	876
11	20-inch 30,98mm WT OEP9	20,0	~	0,2	100	☐	386,43	976
12	20-inch 30,98mm WT OEP10	20,0	~	0,2	100	☐	406,43	1076
13	20-inch 30,98mm WT OEP	3035,5	~	10,0	304	☐	3441,93	1380

Figure 3.7: Overview of the pipeline in OrcaFlex

The start of the pipeline is the part that is upheld by the support rollers, lying horizontally. This part is right before the overbend section, and is not relevant for this research project.

The middle of the pipeline is the part that is upheld by the vessel and stinger RBs. The distance between the first vessel rollerbox and last stinger rollerbox is ± 200 $[m]$. One can extract results such as strain in terms of a position throughout the length of the pipeline, but it becomes hard to visually interpret and coordinate which rollerbox is causing e.g. a specific strain peak. To ease this process, the middle part is split into 10 different sections, each of 20 $[m]$ in order to colour these differently, each serving a visual purpose. By matching the result and its position throughout the length of the pipeline, with the cumulative length and colour of the section, one can visually narrow down at which rollerbox it takes place. To further enhance the results, the target segment lengths are set to $0,2$ $[m]$, since only the results of this part of the pipeline are relevant for this research project.

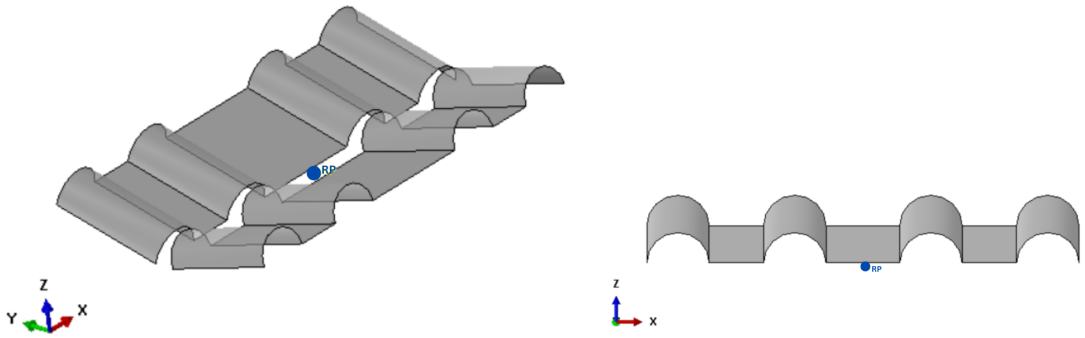
The end of the pipeline is the part of the pipeline after the stinger, also not relevant for this research project.

The sea and seabed:
The density of the sea water and the depth of the flat seabed are given in Table 3.2.

3.2.2. Non-linear Finite-Element Pipeline Model

For the Abaqus model, the following is modelled: The pipeline; All 17 stinger RBs; All four vessel RBs. All of these components are broken down (in greater detail) in the following parts:

Rollerboxes:
The RBs modelled start at RB V13, up to the last one, RB S17. These 21 RBs are modelled using five different rollerbox types, referred to as rollerbox T1, T2, T3, T4 and T5. By adding these five types of RBs according to Table 3.4, the same vessel and stinger RBs configuration as OrcaFlex can be achieved. Their dimensions can be seen in Figure B.18 and B.19. They are created as two rigid 3D shell bodies, with respective dimensions and RPs, as seen in Figure 3.9. Rollerbox type T1, T2, T3, and T4 are similar, where each rollerbox contains four pairs of rollers, one Main Hinge, and two Hinges, as discussed in Section 1.1.5 and 3.1.2. Their differences lies in their geometry, as seen in Figure B.19. The modification from the previous RB model to the improved RB model is illustrated once more in Figure 3.8 and 3.9, respectively. RB T5 on the other hand differs completely, having three roller pairs and only one rotational axis. RB T5 is used only for the last RB on the stinger, RB 17. Figure B.19h illustrates how RB S17 in Abaqus is also modelled different from the other RBs, having only one hinge (one RP). This stays consistent with the modelling of RB S17 in OrcaFlex as previously discussed and illustrated in Figure B.15a.



(a) (b)

Figure 3.8: Previous rollerbox model: a) Isometric view b) Side view

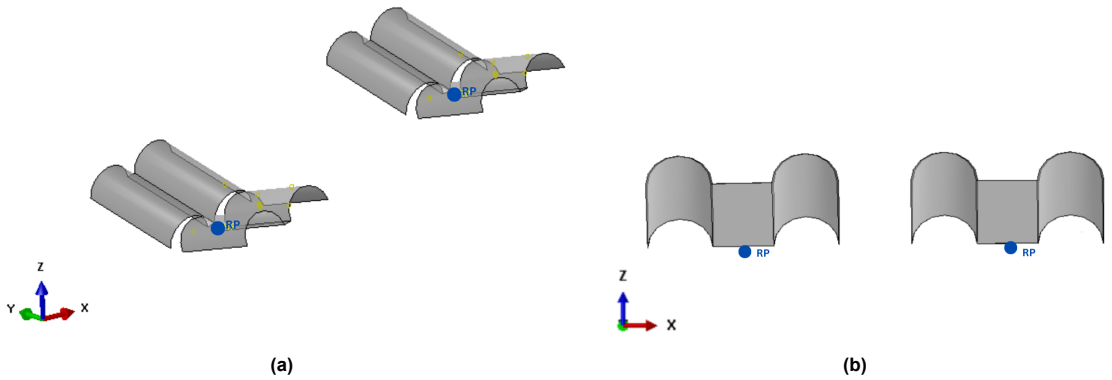


Figure 3.9: Improved rollerbox model: a) Isometric view b) Side view

These RBs are used in the following configuration seen in Table 3.4, which coincides with the RB order and respective geometries in OrcaFlex. Additionally, the coordinates and resulting declination angles of the RBs will be extracted from OrcaFlex. The static analysis is first performed in OrcaFlex, and the resulting declination angles are copied in Abaqus to match the rollerboxes configuration in Abaqus to that in OrcaFlex. The declination angle is the angle which the rollerbox makes in the plane of bending with respects to the horizontal (X) axis, illustrated in Figure 3.10. Further details on this are later discussed in Section 4.1.2.

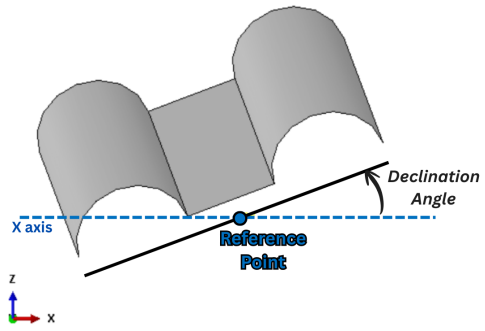


Figure 3.10: Declination angle, the angle with respect to the horizontal (X) axis.

Rollerbox Type	Rollerbox Modelled
T1	$V13,\ V14,\ V15,\ V16$
T2	$S1,\ S2,\ S3,\ S4,\ S5$ $S6, S7,\ S8,\ S9$
T3	$S10,\ S11$
T4	$S12,\ S13,\ S14$
T3	$S15,\ S16$
T5	$S17$

Table 3.4: Rollerboxes modelled in Abaqus using Rollerbox T1, T2, T3, T4 and T5.

Pipeline:
For the purpose of comparison, the pipeline will be modelled in two different ways:

1. Only using line elements $PIPE31H$
2. Combining both line elements $PIPE31H$ with solid elements $C3D8R$

Firstly, the pipeline is modelled using only line elements as this replicates the lumped-mass pipeline, allowing for a better comparison and understanding of their similarities and differences. Line elements $PIPE31H$ is used as it is the most-common line element type found in respective literature as it allows smoother convergence for large-translation non-linear problems such as a full overbend model [16] [17]. In addition, it was also recommended by Structural Engineers at Allseas.

Secondly, the pipeline is modelled using a combination of line elements and additional solid elements. This combination is done to simplify majority of the model by using line elements in sections detailed results are not relevant, reducing the convergence time. Yet, solid elements are used to have a higher-fidelity section within the pipeline, providing more realistic results and allowing for the lumped-mass model to be benchmarked against results deemed to be realistic.

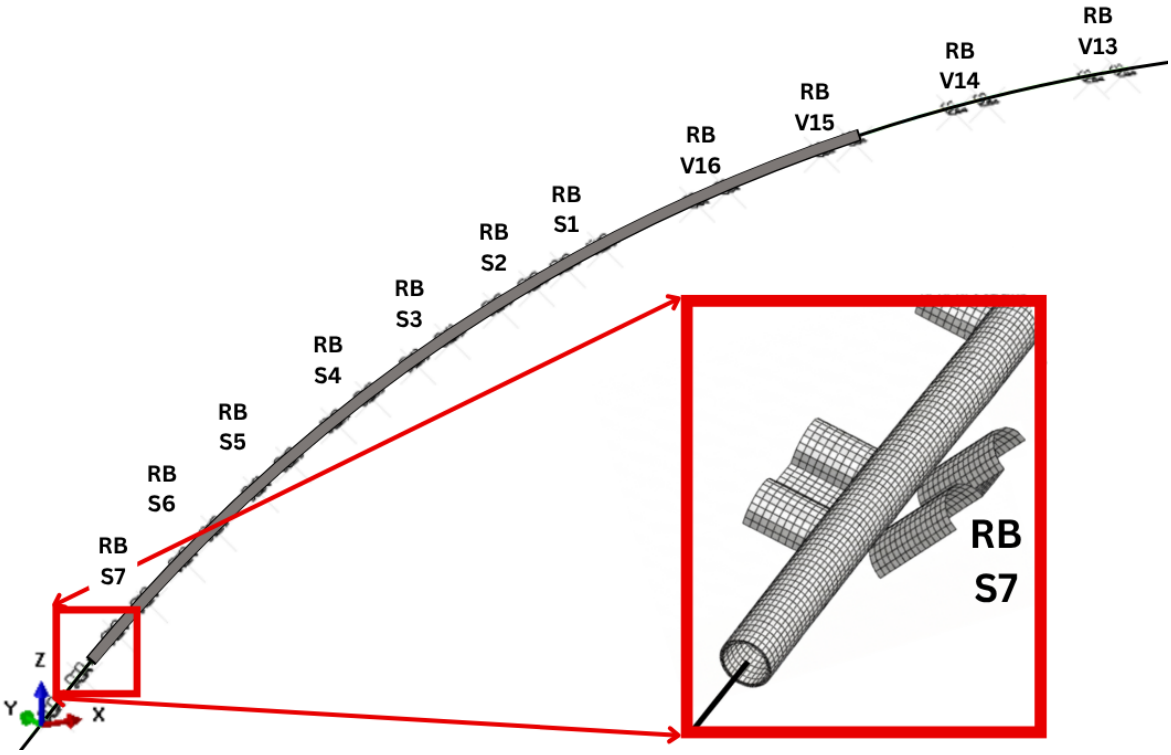


Figure 3.11: Example of a pipeline comprising of both line elements *PIPE31H* and solid elements *C3D8R*

The area which will be modelled with solid elements will be (most of) the overbend section, as only this section is relevant for this research project. The exact location where this solid part will start and end, will be later determined and discussed in Section 4.1. The remaining length of the pipeline will then be modelled using line elements. The pipe section properties used can be seen in Table 3.3a.

The line-to-solid elements connection can be made using either a Kinematic Coupling constraint or Multi-Point constraint (MPC) Beam type. Compared to a Kinematic Coupling, an MPC Beam type does not allow selective constraint of individual DOF and directly creates a rigid connection, transferring all six DOF directly. For cases where all DOF need to be transferred - such as for these full overbend analyses - MPC Beam type is chosen as it provides a rigid connection with a straight-forward application. This direct transfer of all of the DOF can promote quicker convergence in Abaqus when compared to a Coupling constraint.

Material:

Density

Abaqus does not automatically take buoyancy into account. Buoyancy will be accounted for by changing the material density of the section of the pipeline that is submerged. The exact location of this is approximated by using the waterline in OrcaFlex as a reference, and sectioning the pipeline at this location in Abaqus. The length from start of the pipeline, up until this waterline section, will be assigned a regular material density (ρ_{steel}). The length from this waterline section, down to the end of the pipeline will have a submerged material density ($\rho_{submerged}$).

Submerged material density is calculated as follows. Firstly, the dry weight (w_{dry}) of a specific pipe section per unit length is calculated using Equation 3.6. Secondly, the submerged weight ($w_{submerged}$) of this pipe section per unit length is calculated by subtracting buoyancy effects from w_{dry}, Equation 3.7. Lastly, $\rho_{submerged}$ is calculated by multiplying ρ_{steel} by the ratio between $w_{submerged}$ and w_{dry}.

$$w_{dry} = g \cdot \rho_{steel} \cdot \frac{\pi \cdot (OD^2 - ID^2)}{4} \tag{3.6}$$

$$w_{submerged} = w_{dry} - g \cdot \rho_{water} \cdot \frac{\pi \cdot OD^2}{4} \tag{3.7}$$

$$\rho_{submerged} = \rho_{steel} \cdot \frac{w_{submerged}}{w_{dry}} \tag{3.8}$$

Stress-Strain curve
When entering values for a material property for the calculation of finite strain, stress is refers to true (Cauchy) stress and strain refers to logarithmic strain [18]. As mentioned before, the Abaqus model is used as benchmark, thus the material initially defined is assumed to represent reality. The corresponding stress-strain curve is illustrated in Figure B.21. The elastic part of the curve is created by entering both Young's modulus E and Poisson's ratio ν - Table 3.3a, and its plastic part is created by entering stress-strain relation data points.

Interaction:
The interaction between the pipeline and the RBs is set to be a frictionless sliding. This is done by adding the following as a contact property, where all of the values are illustrated in Figure B.20. These values are adopted from previous projects and are not discussed, as their exact derivation is outside of the scope of this research project.

Tangential Behaviour
Friction formulation is set to frictionless as mentioned in Section 2.3.7 - Figure B.20a.

Normal Behaviour
The pressure-overclosure is set to "hard" contact and the constraint enforcement method is set to penalty (standard), as this will ensure that the pipeline does not penetrate the RBs, and that the corresponding reaction forces are transferred.

The contact stiffness is set to non-linear with the specifications seen in Figure B.20b, allowing for an increase in contact stiffness the more the pipeline tries to penetrate and pass through the RBs, mimicking reality and allowing for quicker convergence.

Damping
The values used for damping can be seen in Figure B.20c.

Loads:
The following two loads are set for this model and applied in increments: gravity and tension. These are applied in incremental steps as this allows for a smoother contact behaviour between the pipeline and the RBs as the pipeline gets lowered, allowing for a quicker convergence. The amount of increments, thus the incremental value for each step, is based solely on experience. From past projects, having three so-called Contact steps in which the values of the loads are increased in each step before the final step, assured convergence of the model. During this research project, convergence issues have been experienced when following other incrementation steps, e.g. 50 % of the final values in the first contact step. It has been therefore decided to maintain the incrementation sequence used within Allseas for full overbend analyses in Abaqus. The exact magnitudes of the loads at each steps are given in the Steps description.

Gravity is set in increments of 25 % of the final value, until the final value of $9,80665$ [m/s^2] is reached. The incremental value of 25 % is arbitrary, and has no specific reasoning. This load is applied using a *Gravity load* feature with its acceleration set vertically downwards (negative Z axis displacement), as accustomed for gravity. The gravity load feature applies this to the entire model.

Tension is set in increments of 25 % of the final value, until the final value of $1,2 \cdot 10^6$ [N] is reached. This load is applied using the *Concentrated Force* feature with its direction set to pull the pipeline (negative

X axis displacement), as this is applied to the last node at the end of the pipeline. This way, the force applied mimics bottom tension.

Boundary Conditions:
To ensure the pipeline and RBs behave as expected, the following boundary conditions are set:

Pipeline start:
The first node of the pipeline is limited to all DOF but rotation in the plane of bending (Y axis rotation). Displacements of the first node are limited to ensure the pipeline does not fly away, as the end of the pipeline is being pulled. Only rotation in the plane of bending is allowed as the exact position of the first node of the pipeline with respect to the first rollerbox is unknown. This rotation will allow the pipe to slightly adjust and rotate once contact is made with the first few rollerbox, avoiding excessive displacement and strain at the start of the pipeline due to misalignments.

Pipeline end:
The last node of the pipeline is limited to all DOF but horizontal movement (X axis displacement). Only this DOF is set to be free, as there will be a horizontal force applied to pull on the pipeline to apply tension. In addition to having the pipe end move freely in horizontal direction, it is also displaced downwards until the depth of the seabed is reached. This is done by changing the vertical position of this boundary condition incrementally, until the final value of $1,336$ [m] is reached. This value is greater than the specified water depth of $1,326$ [m], due to the starting position of the pipeline end being 10 [m] above the waterline.

Rollerboxes:
As mentioned in previous sections, the RBs are set at a certain declination angle, which has to be maintained throughout the simulation. Therefore, a boundary condition is applied to each RB part at its RP, constraining all six DOF.

Extra:
As the full overbend models are a 2D problem, extra boundary conditions are set to constrain the following unwanted displacement and rotation: Torsion (X axis rotation); Out-of-plane displacement (Y axis displacement); Out-of-plane rotation (Z axis rotation).These are set along the pipeline, both at the line elements section, and at the solid elements section.

Mesh:
Both the pipeline and RBs have to be meshed before running. The mesh descriptions and number of elements per components can be seen in Table 3.5. These elements are illustrated in Figure 3.12

Component	Mesh	Mesh Type	Number of Elements
Rollerboxes	Linear Quadrilateral Elements	R3D4	37687
Detailed Pipeline Section	Linear Hexahedral Elements	C3D8R	92160
Simplified Pipeline Section	Linear Line Elements	PIPE31H	12501

Table 3.5: Mesh used for in Abaqus, for all components

The RBs are all modelled as a discrete rigid body. These are meshed using $R3D4$ linear quadrilateral elements. The mesh type used for these RBs is the same as that used for rigid bodies in past researches [19] and also for other full overbend models made for past projects at Allseas. Additionally, the mesh size for these RBs is also kept the same, with a total of 37687 elements between all RBs.

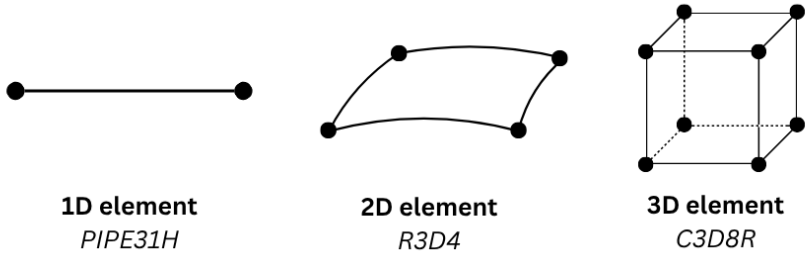


Figure 3.12: Illustration of 1D, 2D, and 3D elements used

The pipeline comprises of the detailed section, modelled using solid elements, and the simplified section, modelled using line elements. The detailed pipeline section is meshed using $C3D8R$ linear hexahedral elements, as $C3D8$ elements are commonly used, and has a so-called reduced version $C3D8R$, which reduced computational time whilst maintaining the same level of results. The simplified pipeline section is meshed using $PIPE31H$ linear line elements, as $PIPE31$ elements provide good approximation of pipes with tube profile, modelled through line elements. Additionally, its Hybrid variation $PIPE31H$ is used. The hybrid variants of line elements are recommended for extremely large slenderness - length to thickness ratio is in the order of 10^3 or more, e.g. pipelines - as geometrically non-linear analyses will have poor convergence otherwise. The hybrid variants treat axial forces as independent degrees of freedom, instead of deriving axial forces from the displacement of nodes, simplifying the mathematical equations thus allowing a quicker convergence [20].

Steps:
The steps in which the static simulation of the full overbend model is performed, are split into five steps. This is done by using: The first step, where initial bottom tension is applied to pretension the pipeline; Three subsequent contact steps, where tension, gravity and the displacement of the pipeline are incrementally increased; And a final step to increase bottom tension, gravity and the displacement of the pipeline to their final values. This sequence is implemented to ensure smooth pipeline-rollerbox interactions, consequently reducing the convergence time of the model. As mentioned above, this sequence is chosen due to its successful and proven application in previous full overbend models. Table 3.6 shows for each step, the increment in percentage of the final value for each variable. The final value of each variable is also shown.

Variable	Pre-tensioning	Contact #1	Contact #2	Contact #3	Final	Final Value
Bottom Tension	50%	50%	75%	75%	100%	1200 [kN]
Gravity	0%	25%	50%	75%	100%	$9,80665$[m/s^2]
Pipeline-End Displacement	0%	25%	50%	75%	100%	1334[m]

Table 3.6: Increments, in percentage of the final value, for the variables used in the Abaqus model for pipeline comparison, for each of the five steps.

Pre-tensioning
The pipeline starts horizontally, where the first node is fixed and the last node can only translate in the horizontal direction. This horizontal freedom will then allow for the pipeline to be tensioned. An axial force is applied to the last node to pretension it. This axial force is the bottom tension force F_{BT}. During this step, F_{BT} is set to 50% of the final F_{BT} value. The F_{BT} will be increased up to its final value in the following steps. Table B.7 shows the exact value for each variable.

Contact #1, #2, and #3
The goal of these steps is to lower the pipeline incrementally, in order to progressively initiate pipeline-rollerbox interactions, and avoid high contact stresses. Following the pre-tensioning step, the pipeline will be horizontal at the start of these steps sequence. Following this, the vertical position of the end of

the pipeline will be lowered, as the magnitude of the bottom tension and gravity applied is increased. The values are increased in equal increments of 25 %. Table B.8, B.9, and B.10, show the exact value for each variable, respectively.

Final
The goal of this step is to finalize the shape of the pipeline. After the previous step, the pipeline is now in contact with the corresponding rollers. The values for gravity, tension, and displacement is increased in order for the sagbend section of the pipeline to finalize its shape. Table B.11 shows the exact value for each variable.

3.2.3. Extracted Result Variables
Once the static analysis in completed, the following result variables are extracted:

1. Declination angles of the pipeline
2. Top and Bottom tension
3. Tension and bending moment at the element interfaces
4. RB reaction forces
5. Strain

Declination angles of the pipeline
The declination angles of the pipeline relate to the angle every node of the pipeline makes with respect to the vertical or horizontal axis. For this research project, the declination angle is defined as the angle with respect to the horizontal axis. By plotting the declination angles of the pipelines against each other, their behaviour and shape can be compared. Having identical declination angles proves consistency between the modelling setups, namely the rollerboxes setup and how they uphold the pipeline, and is therefore checked first thing.

Top and Bottom tension
The top and bottom tension refer to the tensions at the start of the overbend section (top) and the tension at the seabed (bottom). As mentioned in Section 1.1.3, tensions control the shape of the sagbend section. In practice, top tension is applied by either pulling/releasing the pipeline by means of tensioners located on the vessel as seen in Figure 1.1. Subsequently, bottom tension results from pulling suspended pipe sections from the pipe sections laying on the seabed. During analyses, the top and bottom tensions are mainly monitored to control the shape of the pipeline, and also to estimate how much the vessel and its tensioner will have to pull on the pipeline. The more pulling on the pipeline, the more energy and/or fuel the vessel consumes thus the higher the cost. Unlike for the multi-body models, the modelling method using non-linear finite elements requires the bottom tension to be applied as a force. The resulting bottom tension is therefore extracted from the multi-body models. By applying a force at the bottom of the non-linear finite element pipeline, a top tension will result. To ensure identical modelling setup, namely the resulting top tension is check between models.

Tension and bending moment at the element interfaces
During ILS analyses, tensions and bending moments are extracted from the global analyses and implemented in further localized analyses of ILSs, as mentioned in Section 2.1.1 and illustrated in Figure 2.3. The tensions and bending moments are now also extracted and compared at element interfaces. The element interfaces refers to the locations where there is a transition in the type of elements used. When modelling solid-elements pipe section in between a line-elements pipeline, this interface is where the pipeline transitions from line elements to solid elements, and vice versa - illustrated in Figure 3.13. When modelling an ILS, this interface refers to the start and the end of this ILS. These interface loads are subsequently compared to check whether those deriving from the multi-body models are a close approximation of those deriving from the non-linear finite element models.

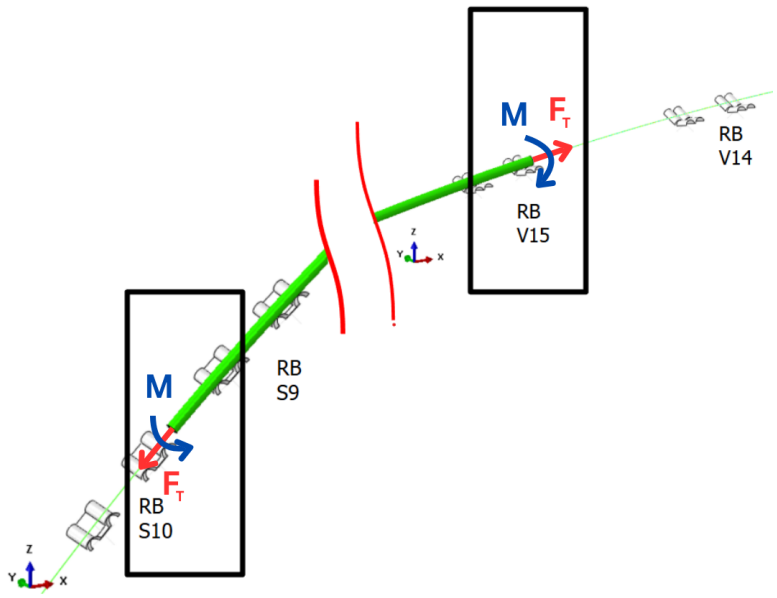


Figure 3.13: Loads at element interfaces, for the case of a solid-line element pipeline

RB reaction forces

RB reaction forces are typically extracted during analyses in order to ensure that these do not exceed the force limits of the RBs themselves, ensuring that their structural integrity is kept. For this research project, the RB reaction forces are rather extracted and compared to assess the differences in magnitude and in contact behaviour between the RBs in OrcaFlex and the lumped-mass pipeline, and between that of the RBs in Abaqus and the non-linear finite element pipeline. The non-linear finite element pipeline results themselves are deemed as reality for this research project, however the behaviour of the RBs in OrcaFlex are closer to reality as these can rotate accordingly during analyses. Additionally, the RBs in Abaqus are set to the declination angles of those in OrcaFlex. This comparison will illustrate the viability of replicating these declination angles in Abaqus using the newly modified RBs, further improving the distribution of the load of the pipeline for each RB.

Strains

The resulting strains is the main variable to be checked. In a full overbend model, these resulting strains are used to check the integrity of the pipeline and its respective components, and the overall feasibility of the installation procedure that is being assessed. The strains are thus compared to see whether the strains in the lumped-mass pipeline approximate those resulting from the non-linear finite element pipeline. Additionally, as there are many strain measures available to extract, it is important to first determine which strains measure(s) will be compared between these pipelines.

3.3. Inline Structure Positioning

During actual installation, the ILS will travel over the vessel and stinger RBs before exiting the overbend section. To accurately replicate and simulate this installation process, the ILS must be analysed at every possible position within the overbend section. This scenario would require displacing the inline structure across a span of approximately $160\ [m]$ - From right before RB V13, up to RB S13 - in increments in the order of at least $10^{-3}\ [m]$. The smaller the increment, the more realistic these analyses would be. However, this level of detail over such a distance, a total of at least 160.000 models would have to be created and analysed, for both Abaqus and OrcaFlex.

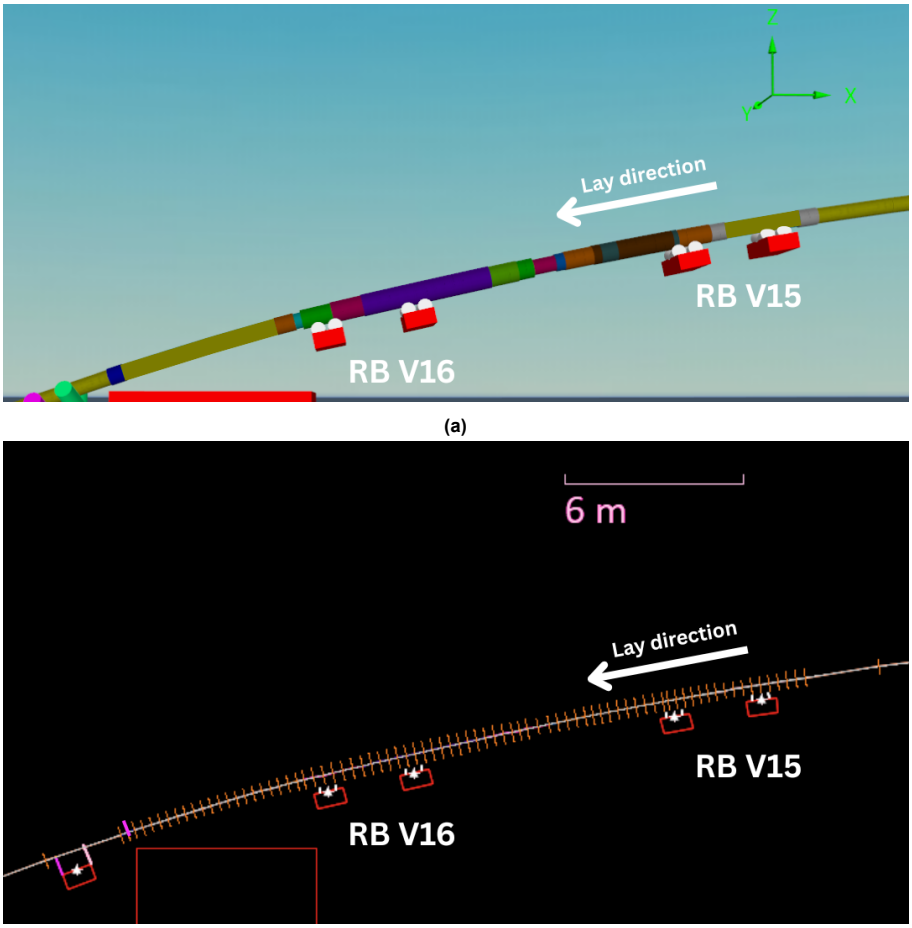


Figure 3.14: Old rollerbox: a) Isometric view b) Side view

For a static analysis, a multi-body model of a pipeline installation including an ILS can take up to a minute to converge, whereas the equivalent 3D non-linear finite element model can take a few hours as mentioned in Section 3.1. Therefore, due to time constraints, the positions in which the ILS is compared for both 3D non-linear finite element model and the multi-body model are limited to just two (critical) positions, which will be determined using OrcaFlex. These are determined by looking at critical positions in the vicinity of the governing rollerbox - the rollerbox affecting the pipeline the most - for the stinger configuration used in this research project. As long as the stinger configuration is not changed, the rollerbox that is governing also stays the same, regardless of whether pipe sections or an ILS is being installed. From the multi-body model used in Section 3.2, the location with the highest strain in the pipeline is extracted. This location relates to determining which of the RBs is governing, and thus affecting the pipeline the most. Since this rollerbox will also be governing when installing an ILS, the amount of simulations to be done are reduced. Now, the ILS has to be only simulated while passing over this one rollerbox, instead of the whole overbend section.

To determine the exact two positions in which the ILS will be placed on this governing rollerbox, a Python script is written. This Python script will displace the ILS over the aforementioned governing RB, and run a static analysis each time to extract corresponding data. These positions are then used in the following section. The positions where the ILS will be compared are:

1. Maximum rollerbox reaction force.
2. Maximum bending angle difference of the ILS.

3.3.1. Max. Rollerbox Reaction Force

The shape of the pipeline in the overbend is determined by the stinger. However, the pipeline does not lay on the stinger directly, but is being upheld by both vessel and stinger RBs. The position where the RB reaction force is the highest is deemed as crucial.

The RBs themselves have a force limit they can endure in order to ensure that their structural integrity is kept, and the resulting RB reaction forces are therefore checked. Ensuring that the RB enduring the highest reaction force maintains its integrity, the integrity of all other RBs are consequently also maintained. This RB reaction force check is therefore a standard procedure within Allseas, ensuring RB integrity and project feasibility. For this research project, this position however will not be used to eventually check whether the exact limit is surpassed or not. As these limits change per vessel, this position is rather used to compare consistency of the RB Reaction Force output values between the lumped-mass pipeline and the non-linear finite element pipeline, in order to take this into account in the (new) analyses procedure that will eventually be proposed.

In OrcaFlex, the values of the RB reaction forces can be extracted from either the individual support pairs ($F_{support}$), the hinges (F_{hinge}), or from the main hinge ($F_{mainhinge}$) of each rollerbox - Figure 3.15. The force at the main hinge is an addition of the normal resultant forces of both hinges, which is a resultant force of the roller support forces - 3.7. The magnitude of the reaction force at the main hinge of each rollerbox will be checked, as this is the standard procedure. Additionally, checking the highest RB reaction force by looking at $F_{mainhinge}$ will result in the same governing RB as checking the highest RB reaction force by looking at F_{hinge}, due to the fact that the RBs in OrcaFlex evenly distributes the load of the pipeline.

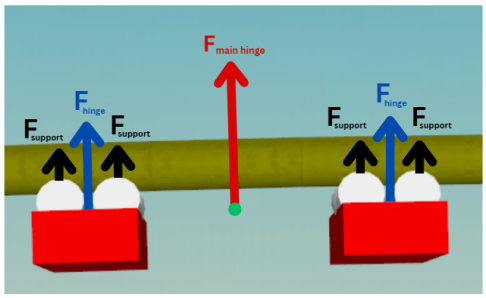


Figure 3.15: Rollerbox reaction force breakdown

Variable	Value [kN]
Hinge1rollerConstr-V13 In-frame connection force	113,8
Hinge2rollerConstr-V13 In-frame connection force	113,8
MainHingeRB-V13 In-frame connection force	227,6

Table 3.7: Rollerbox reaction force breakdown

3.3.2. Max. Bending Angle Difference of the ILS

The ILS placed within the lumped-mass pipeline in OrcaFlex is constructed differently than the ILS constructed using 3D non-linear finite elements in Abaqus. The lumped-mass model tries to approximate the behaviour of that using 3D non-linear finite elements, as these allows for modelling in greater detail. This approximation is mainly done by adjusting the bending stiffness (BS) - the relationship between bending moment and curvature - of different pipe sections throughout the lumped-mass pipeline. Since this approximation is based on the bending that the ILS makes, it is relevant for this research project to compare how both ILSs bends when passing over the overbend section. The method of illustrating how (much) the ILS bends is done by looking difference between the declination angle of the ILS start and ILS end. The more the ILS bends, the greater this bending angle difference will be.

In OrcaFlex, this is determined by extracting the variable node declination angle. The declination angle for the first node (vessel side) is subtracted from that at the last node (sea side) of the ILS - Figure 3.16.

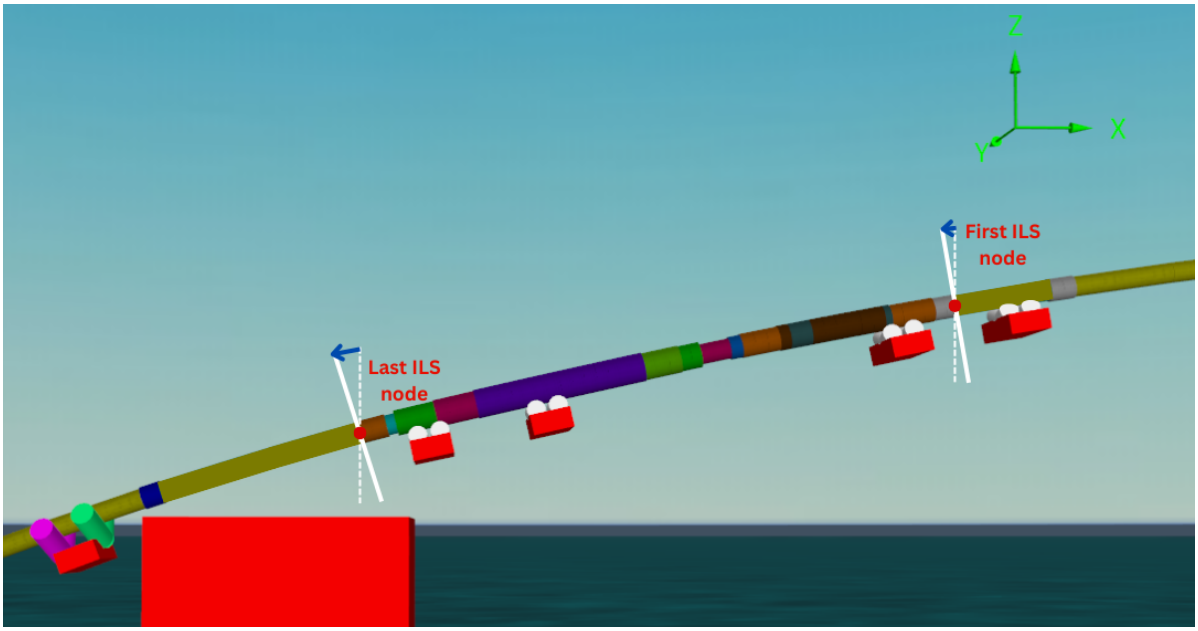


Figure 3.16: Declination angle at the first and last node of the ILS

3.3.3. Python code

To find these two positions, a Python script is written to displace the ILS in increments and run static analysis at different positions.

The ILS will be placed right before coming into contact with the governing rollerbox, and will be displaced until after this rollerbox. The displacements will be in increments of $0.1m$, smaller than any section length, thus ensuring enough contact between the rollerbox and the ILS. Having this displacement increments too small will lead to an excess of (irrelevant) models with similar results.

The Python code written to displace the ILS in OrcaFlex can be found in Appendix C.1. This Python code works as follows:

Code structure

Line 13 defines the entire code as a function called $payout$ with four variables: $sectionlength$, the length of each increment over which the ILS is moved each time; $totalmovement$, the total distance over which the ILS has to be moved; $pipebeforeily$, the length of the pipeline before the ILS; $pipeafterily$, the length of the pipeline after the ILS; $adjacentpipes$, a certain distance of adjacent pipe section on both sides of the ILS, which will be monitored.

Line 23 defines a *for*-loop, which iterates over the amount of files i that need to be created. This amount is determined in line 14, and is the total length to be passed over the length of each increment.

Model definition

Lines 25 and 26 locates and defines the base model. The ILS is first moved in the following code, after which a static analysis is perform on line 71. Subsequently, results are extracted and each model is then saved following line 111.

ILS movement

Line 35 adds a specific length $sectionlength$ to the start of the pipeline and line 36 removes $sectionlength$ from the end of the pipeline, for each model i.

Results

Line 38-68 defines the position of the ILS and its adjacent pipe sections. Additionally, the vessel and stinger RBs are defined in order to extract the rollerbox reaction forces at their main hinge for each model i. These results are exported to an Excel sheet in line 130-134. Line 128 does this for every 50

models that are created. Since this Python code takes a few hours to complete, a backup is created every 50 files in order to prevent losing all data in the case that an OrcaFlex license is lost during execution.

Error mitigation
It might occur that a few models will not converge by only adjusting the position of the ILS in the base model. Therefore, line 70 and 113 provide a $try/expect$ function where, in the case that no errors pop up the code following the $-$ indentation is executed. In the case that there is an error, the code following the $except$ indentation is executed.

3.4. Inline Structure Comparison

The inline structure used for this research project, as mentioned in Section 2.3.7, is that used for a past project referred to as “Shenandoah”. Every ILS is tailored to the project in order to meet its requirements. These requirements are usually determined by e.g. the functionality the ILS will have, and by the surrounding environment. In addition to these project-specific requirements, there are also design requirements that have to be met.

A 3D model of the ILS is built in Inventor in order to start the workflow seen in Figure 2.3. Besides the typical workflow represented by arrows going from left to right, there is also a feedback arrow from the end to the start of the workflow. This represent additional adjustments that have to be made to the ILS. Adjustments to the initial design of the ILS usually take place either due to the design, e.g. not meeting the design requirements or to reduce production cost due to the design being over engineered. As mentioned in Section 2.3.7, the non-linear finite element models of the full overbend will be used to benchmark the multi-body models containing a lumped-mass pipeline, and are assumed to represent reality. Therefore, no further modifications will be performed on the ILS modelled using 3D non-linear finite elements. Figure 1.5 illustrates the final design for the ILS that will be used for this research project.

3.4.1. 3D Non-linear Finite-Element ILS Model

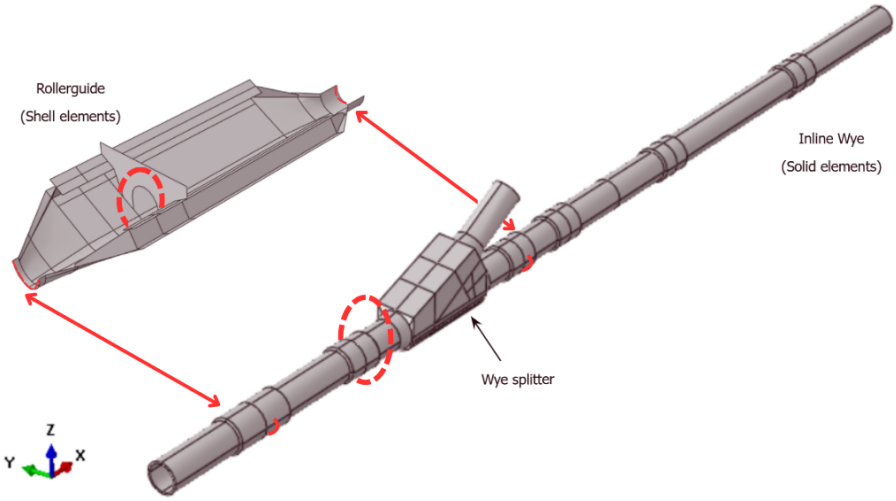


Figure 3.17: Components of the 3D non-linear finite element Inline Structure: Rollerguide (Shell elements) and Inline Wye (Solid Elements)

This ILS has been built for its original project by means of creating two components: The inline wye and one rollerguide. The inline wye is the main body of the ILS, consisting of the pipe section where the medium will flow through and also a wye-splitter component. The rollerguide is placed right beneath the wye component, and is connected through a pin connection on each side. Both the rollerguide and the inline wye are illustrated separately in Figure 3.17. Some components, such as the second rollerguide, the table, and yokes, are not modelled yet still taken into account as point masses illustrated in Figure

B.22a and B.22b. The full overbend models containing an ILS is built upon the previously used Abaqus model for pipe comparison. The pipeline now also consists mainly of line elements. The ILS is placed at a certain location(s) within these line elements as illustrated in Figure 3.18. Further details remain the same and are not discussed again, and can be found in Section 3.2.2

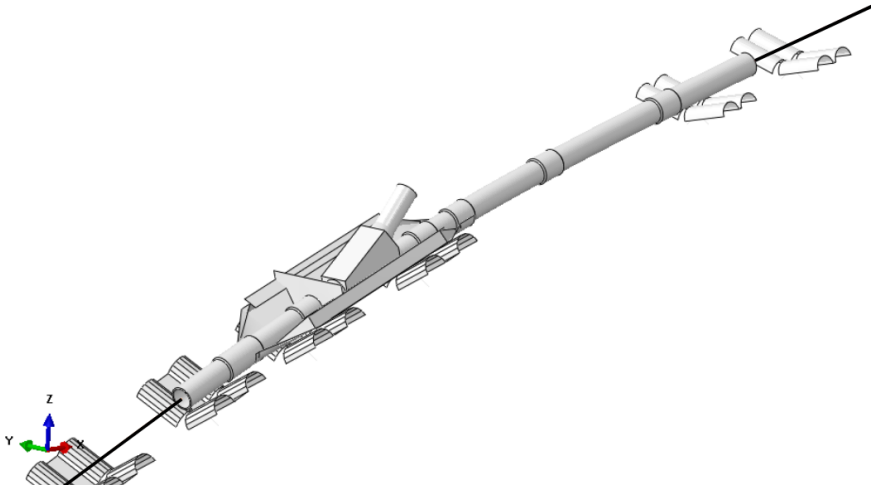


Figure 3.18: 3D Non-linear finite element Inline Structure in the overbend section

When computing full overbend models in general, the objective is global rather than local structural behaviour. In the case of this ILS, the effects of having a jump in stiffness - the wye splitter component - within the pipeline is the most relevant. Local results, such as checking if the pin connectors of the rollerguide can withstand loads, are obtained by doing more precise analyses with smaller and more detailed (localized) models. Therefore, the simplification of this ILS reduces the computational time whilst still delivering relevant results.

Component	Mesh	Mesh Type	Number of Elements
Simplified Pipe Sections	Linear Line Elements	PIPE31H	1165
Rollerboxes	Linear Quadrilateral Elements	R3D4	51835
Rollerguide	Linear Triangular Elements	S4R	26673
Inline Wye	Linear Quadrilateral Elements	C3D8R	225590

Table 3.8: Breakdown of the meshes per component

Similarly to the solid pipe sections in Section 3.2, the majority of the pipeline in this full overbend model is modelled using line elements. In this case, instead of having a section of solid elements representing pipe sections, the ILS modelled by means of mainly solid elements is placed. The ILS is connected to the line elements in the same way, through an MPC connection on both sides, transferring all loads throughout the pipeline. The exact location in which the ILS is found in the pipeline will be determined in Section 4.2 following Section 3.3. Variables such as the tension applied are also discussed in the following sections.

3.4.2. Lumped-mass ILS Model

This ILS has been built for its original project, following the flow chart in Figure 2.3. Due to the limited modelling possibilities using a lumped-mass model for pipeline and pipeline structures, the construction of the ILS is limited to only pipe sections and respective properties modifications. The ILS thus consists of different types of pipe sections, differing in diameter and bending stiffness. Due to the fact that this ILS was used for the previous project, the ILS itself is not altered. The properties for the pipe sections used for this ILS representing its various components have been determined by engineers and are assumed to be correct. Figure 3.19 further extends the previously shown sectioning of the ILS in Section 2.2. The corresponding section properties can be found in Figure B.6.

The lumped-mass pipeline containing an ILS is built upon the previously used lumped-mass pipeline model for homogeneous pipeline comparison. Properties of certain pipe sections are now altered in order to create the ILS and approximate its behaviour. Further details remain the same and are not discussed again, and can be found in Section 3.2.1.

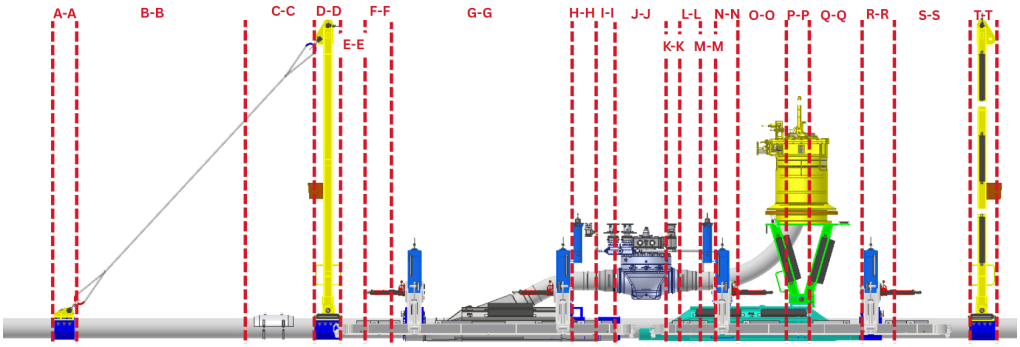


Figure 3.19: Side view of the inline structure with OrcaFlex sections

3.5. Time-efficient ILS

In order to tackle the problem about the ILS fidelity mentioned in Section 2.2 whilst still performing a structural integrity check, two new methods to create an ILS using the lumped-mass model are proposed, tested, and later discussed in Section 4.4.

As mentioned, an ILS is replicated in the multi-body models by means of modifying various pipe section properties within the lumped-mass pipeline in such way that the resulting behaviour of the ILS modelled using 3D non-linear finite elements is approximated. Naturally, the more sections used to discretize a structure, the closer it is approximated. Yet, the downside of this is the amount of tedious work every additional section adds, with the aforementioned question whether these additional sections enhance the simulation results or not. Therefore, alternative time-efficient methods for the creation of the ILS used for this research project is additionally discussed, tested, and analysed, resulting in two simplified ILSs. As seen in Table 3.3, materials in OrcaFlex are defined by their stiffness values, where the main driver is the bending stiffness. Torsional stiffness does not play a role, as this research project is simplified to a 2D problem, and axial stiffness only changes if the cross-sectional area is changed. The bending stiffness in OrcaFlex is defined by entering values of a Bending Moment - Curvature curve. Values lying between those entered and linearly interpolated, and values lying outside are linearly extrapolated [21].

For these new methods, the ILS modelled using 3D non-linear elements will first be divided into sections. For each section, a new material is created in OrcaFlex, and corresponding curvature and bending moment is then entered as a bending moment - curvature relation value. The exact procedure used to extract this relationship value is what differs between the two methods. Therefore, only the extraction of bending moment - curvature relation value is explained separately for both methods. The remaining steps for both methods are the same and thus are explained just once.

3.5.1. Time-efficient Methods

The following is a step-by-step on how a simplified ILS is created, supported by additional illustrations: *Firstly*, a static analysis of the full overbend model containing a 3D non-linear finite element ILS model has to be completed in Abaqus. This completed analysis will be used to extract both the resulting bending moment and curvature of the later-specified sections, which will later form a new material in OrcaFlex.

Secondly, the level of discretization of the ILS has to be determined, which subsequently determines the amount of sections to be created within the lumped-mass pipeline in OrcaFlex. In other words, a decision has to be made on the fidelity of the ILS in OrcaFlex. This step refers back to the main issue discussed in Section 2.2, where the importance of the level of discretization, i.e. the number of sections representing the ILS in the lumped-mass pipeline, is at question.

To further tackle the issue on whether a high level of discretization is needed, a new ILS will be created represented by only three sections in the lumped-mass pipeline. The resulting behaviour of this ILS is subsequently benchmarked with the original ILS in OrcaFlex, and also that modelled using 3D non-linear finite elements in Abaqus. To further simplify this new ILS, only the bending stiffness of the pipe sections within the pipeline will be altered. Naturally, there are multiple ways of recreating the ILS within the lumped-mass pipeline, containing different combinations of amount of sections, bending stiffness, and varying section diameters. For this research project, the amount of sections is chosen as three. Figure B.22a illustrates a side-view of the ILS in Abaqus, where it becomes evident that its components can be split into a minimal of three sections - the left section containing the Yoke tower, Rollerguide, Wye splitter, and a Mudmat, and the right section containing the Hub tower, Rollerguide and the other Mudmat - and naturally an additional section connecting these two. Figure 3.20 illustrates this by means of three blocks covering their respective components. Figure 3.21 illustrates these sections now compared to the CAD model of the ILS.

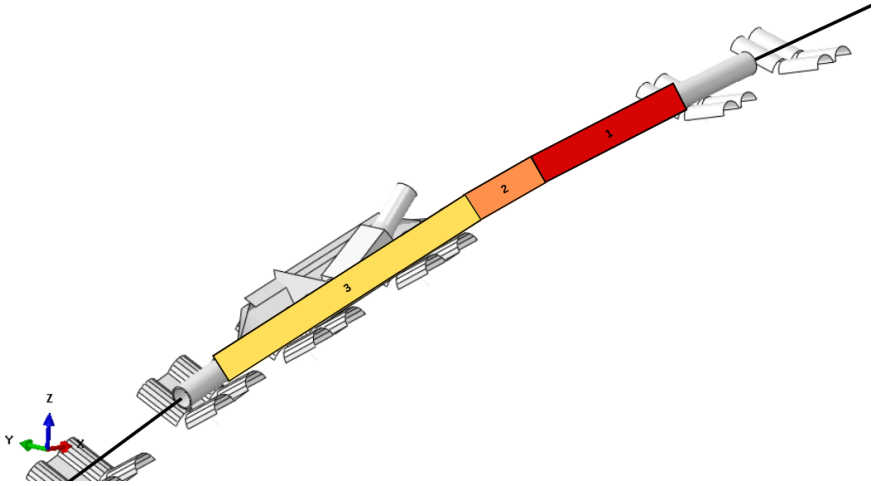


Figure 3.20: Non-linear finite element inline structure with illustrative sections

Thirdly, the bending moment - curvature relationship for each section has to be extracted. As mentioned before, two new methods are proposed which differ in the way the way that this relationship is extracted. The first method will extract and use the median bending moment-curvature value of each section. This is done by extracting the bending moment and curvature from the middle cross-section of each section. The second method will extract and use the mean (average) bending moment - curvature value of each section. This is done by extracting the bending moment and curvature at a few cross-sections within the section, and averaging them out. Unlike the lumped-mass model, solid elements do not provide the ability to directly extract their rotation and curvature [22]. The work around created for this also refers to trigonometry and is explained in the subsequent steps listed below, using the following equations:

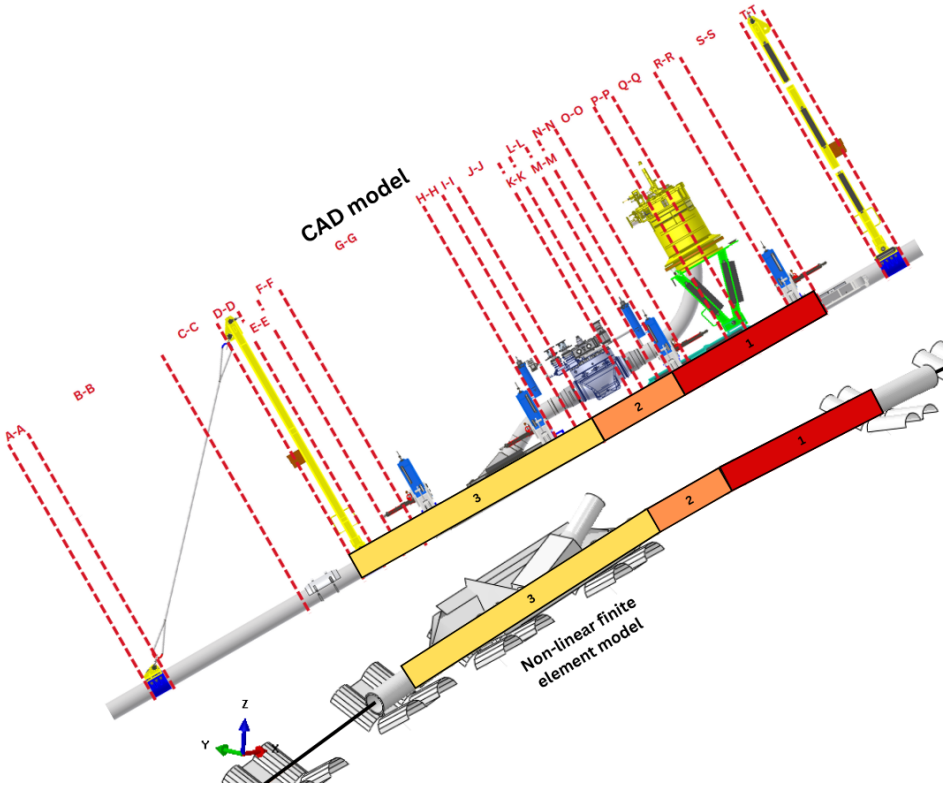


Figure 3.21: Non-linear finite element inline structure with illustrative sections compared to its CAD model

Equation 3.9 is used to calculate the angle between cross section of part n with respect to the vertical axis using trigonometry, where $X_{n_{differnece}}$ and $Z_{n_{differnece}}$ are the horizontal and vertical differences between the top and bottom nodes of cross section n respectively, and ϕ_n is the declination angle of cross section n with respects to the vertical axis.

$$\phi_n = \arctan(\frac{X_{n_{differnece}}}{Z_{n_{differnece}}}) \tag{3.9}$$

Equation 3.10 is used to calculate the curvature, using the length of part n with the difference in cross-sectional angle between its declination angle ϕ_n, and that of the previous section ϕ_{n-1}. This angle difference relates to how each specific part n is bending.

$$Curvature = \frac{1}{\frac{L_n}{\phi_n - \phi_{n-1}}} \tag{3.10}$$

To calculate the median bending moment and curvature for a section, the following steps are taken:

a. Divide the section in two equal parts of length $L = \frac{L_{section}}{2}$.
b. Find the node pairs at cross section L, consisting of the top-most node $N_{1\ top}$ and the bottom-most node $N_{1\ bottom}$, Figure 3.22a.
c. Use the Abaqus Query tool to find the distance values for each node pair, and specifically extract the Deformed Distance difference for each node pair.
d. Input the $X_{n_{differnece}}$ and Z value $Z_{n_{differnece}}$ in Equation 3.9 to get the angle of the cross section ϕ_n with respects to the vertical axis (declination angle).
e. Insert the declination angles in Equation 3.10.
f. Extract the bending moment by showing the out-of-plane reaction-moment component for each cross section, illustrated in Figure 3.22b

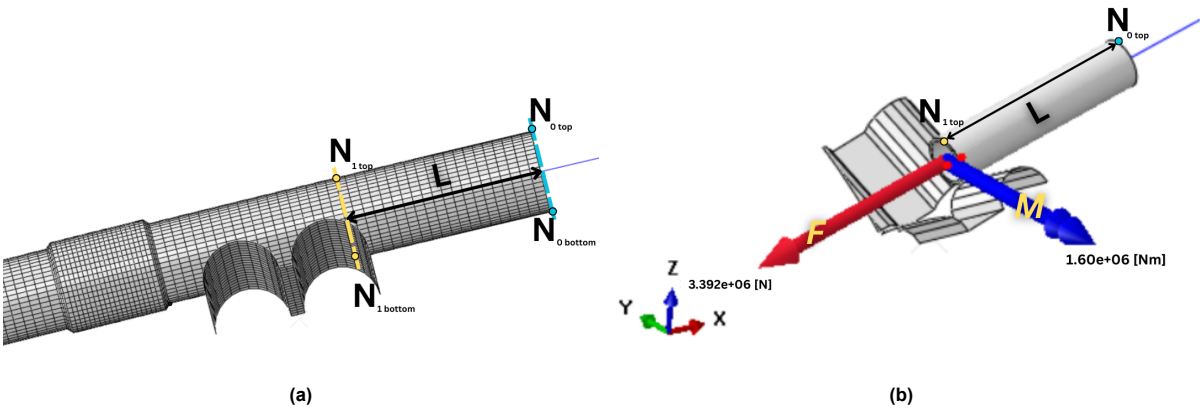


Figure 3.22: Median bending moment and curvature calculation example for one section: a) node pairs and parts n lengths b) bending moment

To calculate the mean bending moment and curvature for a section, the following steps are taken:

a. Divide the section of length $L_{section}$ into equal parts of length $L_n = \frac{L_{section}}{n_{parts}}$.
b. Find the node pairs at each cross section L_n, consisting of the top-most node $N_{n\ top}$ and the bottom-most node $N_{n\ bottom}$, Figure 3.23a.
c. Use the Abaqus Query tool to find the distance values for each node pair, and specifically extract the Deformed Distance difference for each node pair.
d. Input the X $X_{n_{differnece}}$ and Z value $Z_{n_{differnece}}$ in Equation 3.9 to get the angle of the cross section ϕ_n with respects to the vertical axis (declination angle).
e. Insert the declination angles in Equation 3.10.
f. Extract the bending moment by showing the out-of-plane reaction-moment component for each cross section, illustrated in Figure 3.23b
g. Repeat for each part n of the section
h. Average out the bending moment and curvature for each section.

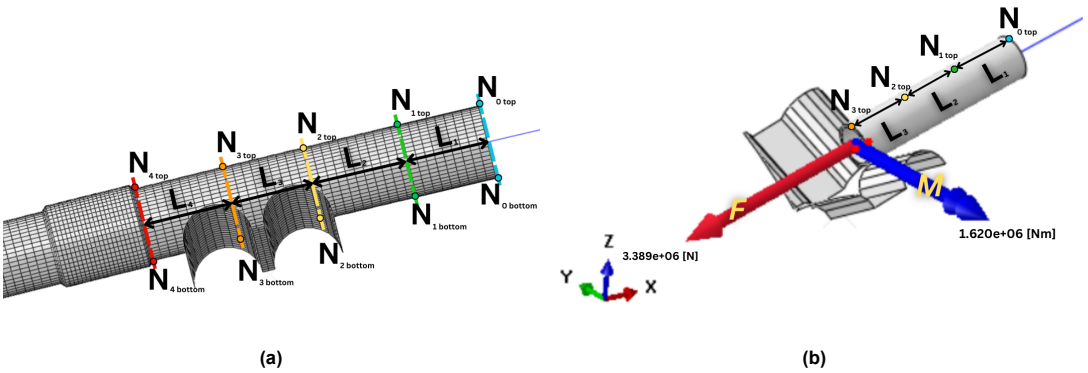


Figure 3.23: Average bending moment and curvature calculation example for one section: a) node pairs and parts n lengths b) bending moment

Lastly, after calculating bending moment and curvature for each section of the ILS, these are implemented into the lumped-mass pipeline in OrcaFlex. Once in OrcaFlex, a new Bending Stiffness input is created for each section. The first point for each Bending Stiffness is a bending stiffness of 0 $[Nm]$ for a curvature of 0 $[rad/m]$. The second point entered is the value calculated for the respective section. After creating a Bending Stiffness input for each section, these are assigned to a pipe section of length $L_{section}$ with its corresponding bending stiffness. Section 4.4 further implements these steps.

The following two ILS will be created: ILS #1 will be created by by means of the median the bending moment and curvature of each section; ILS #2 will be created by means of the mean bending moment-curvature value of each section. These two ILSs, and also the original ILS, will be compared with the 3D non-linear finite element ILS in Abaqus.

3.6. Python Script for Abaqus

To be able to recreate the rollerboxes configuration resulting from OrcaFlex in Abaqus, each RB in Abaqus has to go through the following steps: Firstly, the rigid body of the RB has to be imported into the assembly. Secondly, once imported, the rigid body has to be displaced so its RP lies on the origin. This is due to the fact that the RB coordinates extracted from OrcaFlex are naturally relative to the origin. To ensure consistency and overall correctness, rotation and translation of each RB takes places at and from the origin respectively. Thirdly, once the RP lies on the origin, the rigid body can be correctly rotated to its respective declination angle. Lastly, once rotated, the rigid body can be translated to its position using the respective coordinates. These steps have to be repeated for a total of 44 rigid bodies: Three support rollers and RB S17, each containing one rigid body; Four vessel RBs and 16 stinger RBs, each containing two rigid bodies.

In order to automate this procedure, the latter three Python scripts shown in Appendix C are used. Initially a Python script, written for a previous project, was given by a Structural Engineer at Allseas to automate the process of placing RBs. This script was a project draft, thus is completely modified and tailored to the models of this research project. The script seen in Appendix C.2 is used for the homogeneous pipeline case discussed in Section 3.2. The script seen in Appendix C.3 and C.4 are used for the Max RB Reaction Force and Max Bending Angle Difference cases discussed in Section 3.4, respectively. The essence of these three scripts are the same, the only differences are the resulting RB coordinates and declination angles, which are case-specific. All three of these Python scripts follow the following structure:

- The first 25 lines of code start of by defining the model, the RB coordinates, the RB declination angles, the amount of RBs, and their names.
- The following sixty-two lines of code start the iteration process (loop). The loop iterates per roller-box, and it starts off by identifying exactly which RB has to be placed, also defining its type. As each type has a different dimension, it is important to first know which RB is going to be set.
- The last few lines of code extract the respective RB coordinates and declination angle, rotating and translating the RB in such sequence in order to correctly place each one.

4

Results

This chapter will provide the results and findings after running static analyses on the models proposed in Chapter 3. Section 4.1 will elaborate in greater detail on the differences between a homogeneous lumped-mass pipeline and non-linear finite element pipeline, providing an answer to the second research sub-question. Section 4.2 will elaborate in greater detail on the positions in which the ILS will be placed for the subsequent section, providing an answer to the third research sub-question. Section 4.3 will elaborate on the differences between modelling the ILS using a lumped-mass model and 3D non-linear finite element model, providing an answer to the fourth research sub-question. Section 4.4 will present an alternative method for the modelling of ILSs, providing an answer to the fifth research sub-question. Section 4.5 will then synthesize aforementioned steps into one framework, serving as an answer to the main research question. Appendix D contains a list of definitions of terms used throughout this chapter.

4.1. Pipeline Difference

As mentioned in Section 3.2, the pipeline in Abaqus is modelled in two ways: Using only line elements $PIPE31H$; Using a combination of mainly line elements $PIPE31H$, with solid elements $C3D8R$ at a certain location. The specific length and location of this detailed pipeline section is determined by first completing the static analysis in OrcaFlex with a homogeneous pipeline mentioned in the previous section. The location for the detailed pipeline section in Abaqus is determined to cover the length where the highest strain peaks occur.

The strain in the pipeline is looked upon due to the fact that, as mentioned in Section 1.1.2, the shape of the overbend section is displacement-controlled and the effects on the pipeline are induced by the pipeline bending on the stinger. Looking upon the location with the highest strain will then result into the location where the pipeline is being bent the most. This section is thus seen as the most critical area, and is therefore represented with solid elements in Abaqus in order to have more detailed results. To ensure simplicity, tidiness, and consistency, the resulting graphs and curves corresponding to the lumped-mass pipeline will be coloured red, and labelled as “OrcaFlex”. Those corresponding to the non-linear finite elements pipeline will be coloured (a shade of) blue, and labelled as “Abaqus”.

4.1.1. Material Consistency

As mentioned in Section 3.2, the material input of both OrcaFlex and Abaqus differs. For this research project, the Abaqus models are deemed to be realistic and used as benchmark, thus the material defined is also used to benchmark that of OrcaFlex. In order to maintain consistency between pipelines, the properties have to be checked. Consistency between pipeline geometry is relatively easy to ensure, however the material definition itself requires more than cross-referencing numbers. The material in Abaqus is defined by a stress-strain curve: For the elastic part of this curve, the Young’s modulus E and Poisson ratio ν is entered; For the plastic part of this curve, data points are entered following a plastic stress-strain curve illustrated in Figure B.21. The material in OrcaFlex is defined using data points from a bending moment-curvature curve, thus a bending moment-curvature relation has to be

established and extracted from Abaqus to be used as material input in OrcaFlex in order to ensure that the same pipeline is being used. The following method is thus proposed and employed in order to ensure material consistency between pipeline models. By modelling the pipeline using line elements in Abaqus, it is possible to directly extract both the bending moment $NFORCESO5$ and curvatures $SK1$ results [22], and input this into OrcaFlex.

Firstly, a static analysis is performed on an OrcaFlex model with the stinger configuration used throughout this research project. It is noteworthy to highlight that for this initial static analysis, the only discrepancy is the pipeline material, thus geometrical consistency has already been ensured. Despite the material not being identical yet, the resulting rollerboxes configuration is close enough, and does not have to be identical to get correct material curve from Abaqus. This is due to the fact that the resulting material data is based on bending moment and curvature, thus having slightly different rollerbox declination angles will still result in the same material curve.

Secondly, once the static analysis is done, the procedure to replicate the rollerboxes configuration mentioned in Section 4.1.3 is performed. This will replicate the rollerboxes configurations in order run a static simulation.

Thirdly, once the static simulation in Abaqus is completed, both the bending moment $NFORCESO5$ and curvatures $SK1$ are plotted against each other, and the resulting Abaqus curve illustrated in Figure 4.1 is obtained. Before being able to input this curve into OrcaFlex, the curve has to be filtered by first removing data points outside of the overbend section. Additionally, any recurring values have to be removed, as the OrcaFlex material input cannot contain repeating and/or descending values. This will result in the following (filtered) curve, illustrated in Figure 4.1. Additionally, the initial OrcaFlex material is also shown in order to illustrate the difference in material.

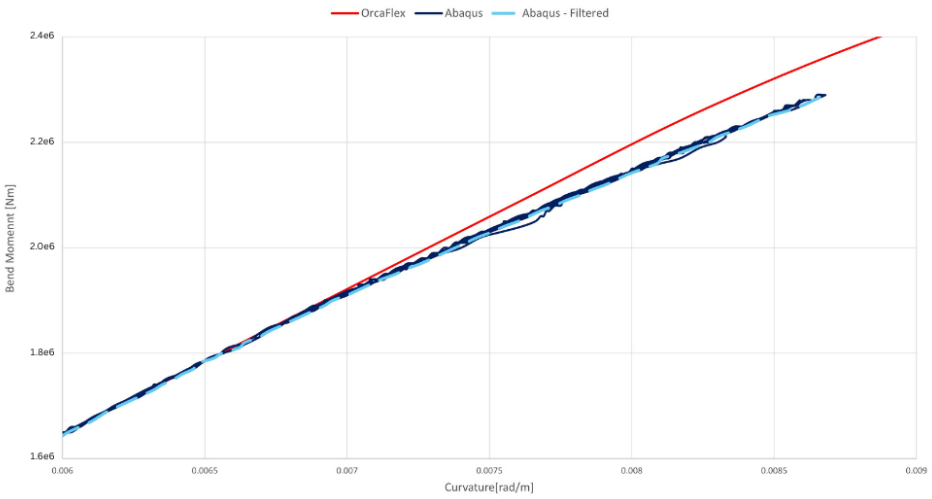


Figure 4.1: The plastic area of the resulting bending moment-curvature relation curves

Figure 4.1 illustrates a zoom-in graph of that in Figure B.24, as the elastic region of the curves are consistent and the real difference takes place in the plastic regions. The OrcaFlex curve is the original material curve that was used in order to extract the rollerboxes configurations, as mentioned in the first two steps. This material curve in OrcaFlex has an identical elastic part to those in Abaqus as seen in Figure B.24, however differs for the plastic part as seen in 4.1. When directly extracting the bend moment-curvature relation, a noisy curve is obtained. This initial resulting curve is subsequently filtered by means of removing recurring values, in order to be used as material input in OrcaFlex. This is illustrated in Figure 4.1, as the Abaqus curve has a flattened equivalent version.

Lastly, once the correct material curve is imported into OrcaFlex, the static analysis for a homogeneous pipeline has to performed again. Additionally, the previously-mentioned rollerboxes configuration has to also be recreated. This is due to the fact that the material from OrcaFlex has now changed, thus the

RB declination angles will slightly differ. The differences might be in the decimals (for a homogenous pipeline), yet everything has to be performed again in order to be as consistent as possible.

4.1.2. Solid Pipe Location

After running the static analysis on the homogeneous lumped-mass pipeline, the Worst ZZ strain curve illustrated in Figure 4.2a is obtained.

The strain peaks are labelled with their corresponding rollerbox. Figure 4.2a shows the highest axial strain peak at RB S1, and the second-highest strain peak at RB S7. Figure 4.2b zooms into the highest axial strain peak, showing the strain curve at both RB S1 and RB S2 to further illustrate how to visualize the strain "bumps" caused by the RBs. By means of Figure 4.2a, it is concluded to place the section of the pipeline represented with solid elements over the length of RB S15 up to RB S7, with a total length of solid-elements pipe of 80 $[m]$.

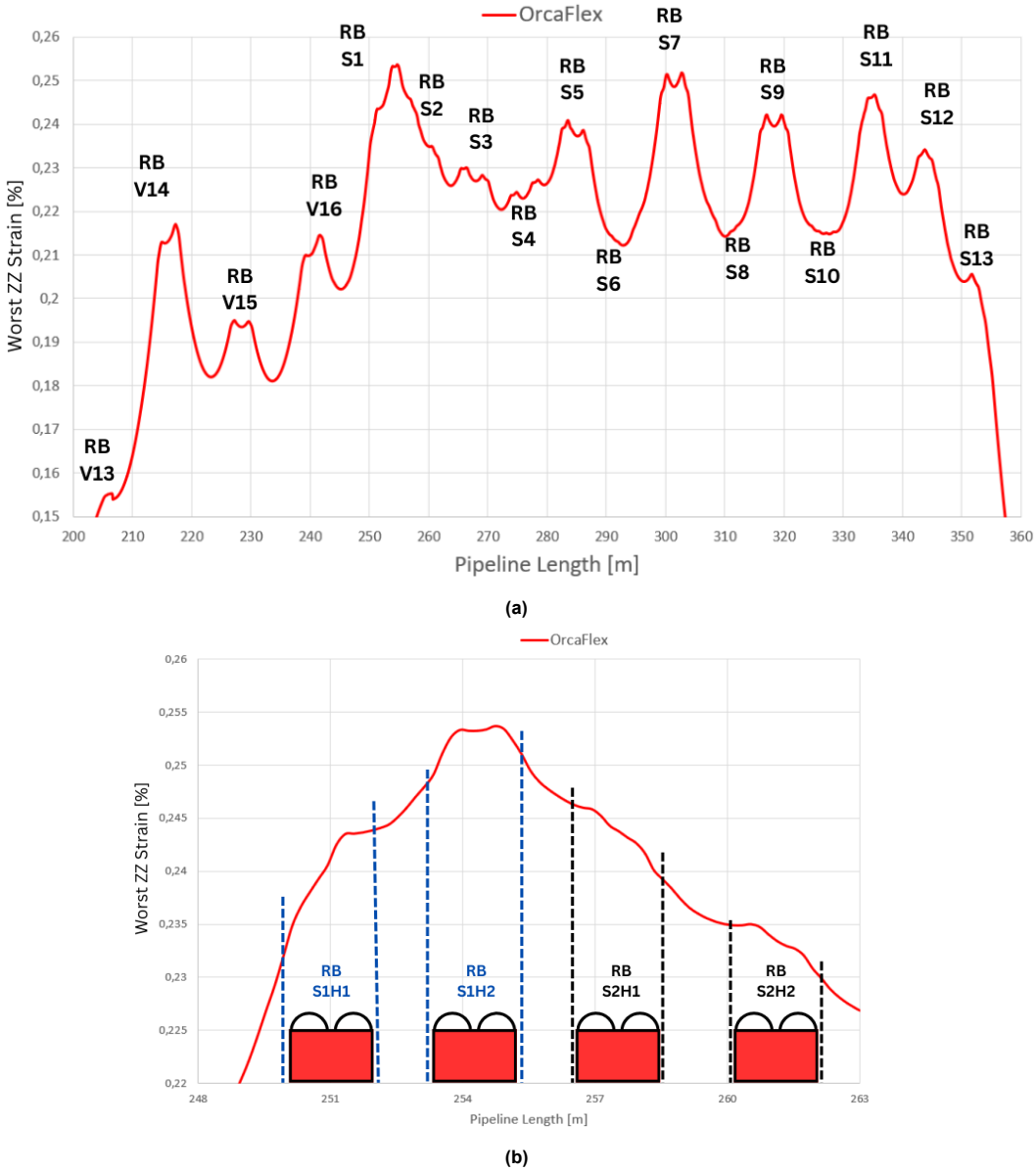


Figure 4.2: Worst ZZ strain over: a) The overbend section b) Zoom-in for RB S1 and RB S2

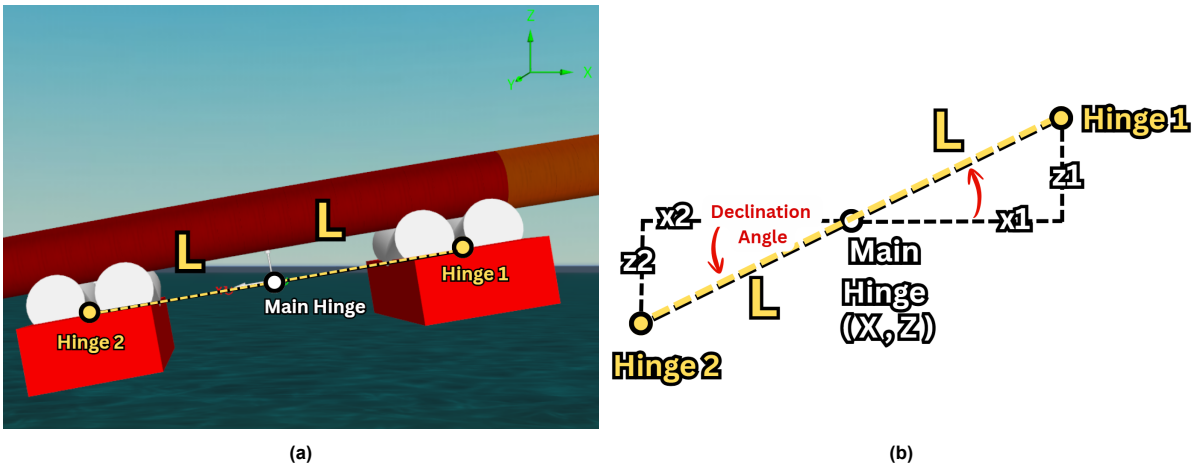
An additional pipeline is modelled, with a total length of solid-elements pipe of 95 $[m]$, in order to mitigate boundary-condition effects. The pipeline of 80 $[m]$ transitions to line elements right at RB S7, where the second-highest strain peak lies on the curve. By having a longer replica of this pipeline, boundary-condition effects on a strain peak can also be studied.

4.1.3. Rollerboxes Configuration

In order to match the rollerboxes configurations in both Abaqus and OrcaFlex, the declination angles and coordinates of the RBs are needed. The RBs in OrcaFlex are able to rotate and distribute the load of the pipeline evenly during static analyses. In addition, OrcaFlex provides the ability to extract the resulting angles that these RBs have once an analyses is complete. It is therefore decided to directly extract the declination angles from OrcaFlex in order to replicate these in Abaqus. Matching the rollerbox configurations will ideally lead to identical setup.

The declination angle for each RB Main Hinge and respective hinges - Hinge 1 and Hinge 2 - can be directly extracted from OrcaFlex. Yet, the respective coordinates of the hinges can not. This is due to the fact that the RB Hinges are constraint features in OrcaFlex, which are linked to their respective RB Main Hinge constraints. Therefore, the RB Hinge positions are relative positions to their RB Main Hinge. In order to still get the coordinates for the RB Hinges, the following trigonometric procedure is done for each rollerbox:

1. Extract X and Z coordinates of the RB Main Hinge.
2. Extract the declination angle of the RB Main Hinge, $\phi_{MainHinge}$.
3. Extract the shortest distance L of both RB Hinge 1 and RB Hinge 2 from RB Main Hinge, Figure 4.3a. These distances L are set per RB type, and are all listed in Table B.12
4. Calculate the x and z distance components of both RB Hinge 1 (x_1, z_1) and RB Hinge 2 (x_2, z_2) from RB Main Hinge using Equation 4.1 and 4.2 respectively, illustrated in Figure 4.3b.
5. Calculate the coordinates of both RB Hinge 1 and RB Hinge 2 using Equation 4.3 and 4.4, respectively, illustrated in Figure 4.3b.
6. Input all RB Hinge and their respective declination angles into the Python script mentioned in Section 3.3.3.
7. Run this code in the Abaqus Command Line window to automatically recreate place all stinger and vessel RBs.



(a) (b)

Figure 4.3: a) Rollerbox terms in OrcaFlex b) Trigonometric calculation illustration for its coordinates

Both Equation 4.1 and 4.2 make use of trigonometry, illustrated in Figure 4.3b, in order to calculate x_1, x_2, and z_1, z_2, respectively. These equations make use of different hinge distances L and main hinge declination angles $\phi_{MainHinge}$ - Table B.12 and B.13 respectively.

$$x_n = L \cdot \arccos(\phi_{MainHinge}) \tag{4.1}$$

$$z_n = L \cdot \arcsin(\phi_{MainHinge}) \tag{4.2}$$

Equations 4.3 and 4.4 are used to calculate the RB Hinge x and z coordinates respectively. The $\pm$ icon in both equations refers to the fact that, as seen in Figure 4.3b, Hinge 1 and Hinge 2 lie in opposite direction from the Main Hinge. The calculation to get their coordinates are technically the same, yet Hinge 1 lies more towards the positive X-axis and Y-axis than Hinge 2.

$$x_{Hinge} = X \pm x_n \tag{4.3}$$

$$z_{Hinge} = Z \pm z_n \tag{4.4}$$

This procedure is performed for all RBs expect RB S17, due to the fact that RB S17 has only a main hinge, thus its coordinates can be extracted directly. After performing this procedure correctly, the following rollerboxes configuration illustrated in Figure 4.4 is obtained. This figure graphs only the position of the RB Main Hinges in OrcaFlex, the position of the respective RB Hinges in Abaqus, and illustrates that both rollerboxes configurations are identical forming the same stinger shape. The corresponding RB Main Hinge declination angles are found in Table B.13, the RB Hinge declination angles are found in Table B.14, and these are all illustrated in Figure B.25. The Python scripts used to recreate the rollerboxes configuration in Abaqus can be found in Appendix C.2.

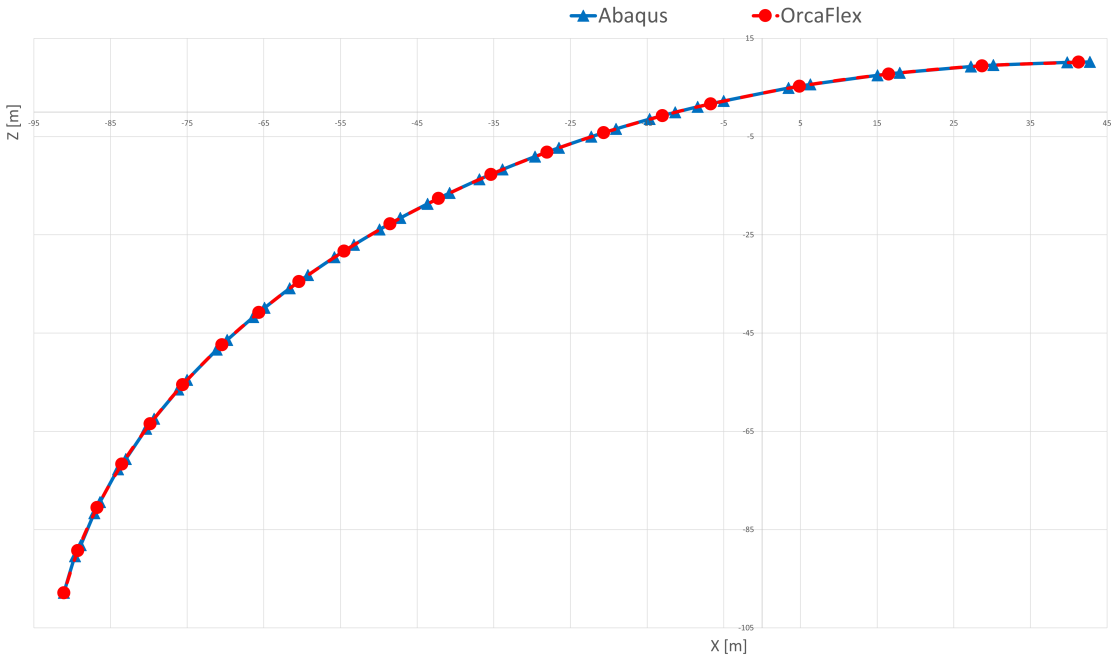


Figure 4.4: Rollerboxes configuration comparison for Abaqus and OrcaFlex

4.1.4. Result Conclusions

After running static analyses with the same stinger configuration in both Abaqus and OrcaFlex, the following results are obtained:

Identical Declination Angle

As mentioned in Section 3.2.3, the declination angles of the pipelines are compared in order to ensure that these bending behaviours are the same. Figure 4.5 illustrates the comparison of the declination angle for OrcaFlex with that of line and solid elements in the overbend section. Only the overbend section is shown, as the rest of the pipeline is not relevant for this research project. This figure illustrates that all pipelines have the same declination angle, ensuring similar setups.

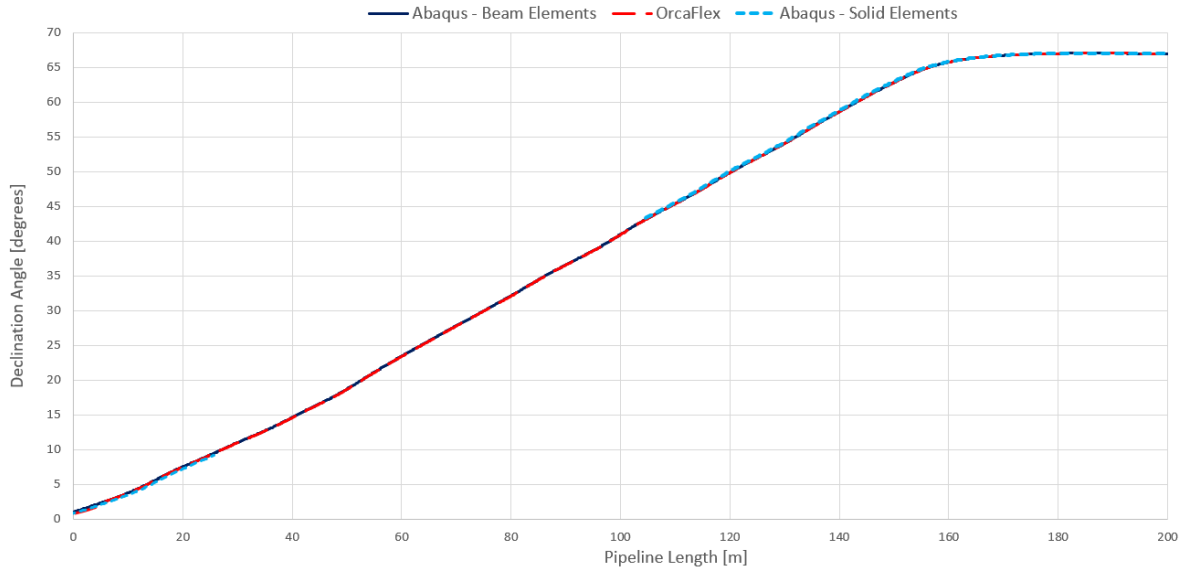


Figure 4.5: Declination angle comparison

Identical Tension

Figure 4.6 illustrates the comparison of both top and bottom tension for all four models. The different model setups have proven to replicate the top and bottom tension identically, thus is not investigated in further cases.

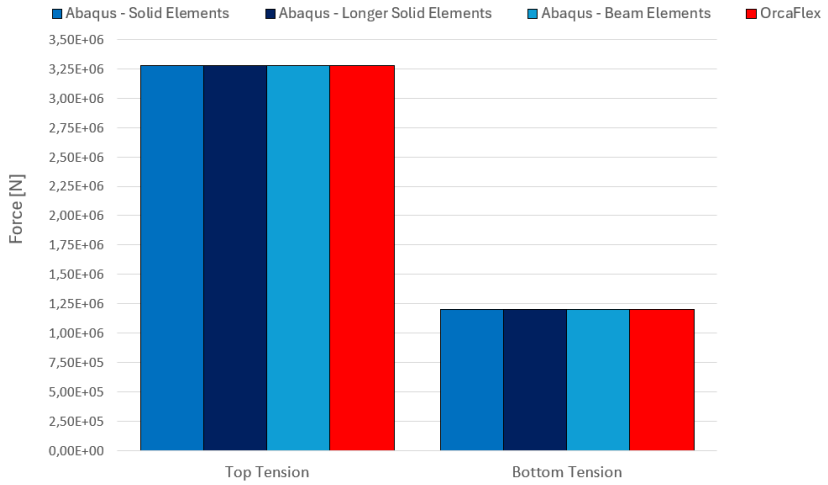


Figure 4.6: Top and bottom tension comparison for all 4 pipelines

Loads at the solid pipe interfaces

Figure 4.7 illustrates load comparison at the solid pipe interface, where the pipeline transitions from line elements to beam elements (Start) and vice versa (End) - Figure 3.13. The loads at the interfaces are also checked for the homogeneous pipeline, and not just for ILSs, as the overall results for the lumped-mass pipeline and non-linear finite element pipeline should be similar. In the case of any discrepancy, these difference could be attributed for example to a boundary condition effect at the element transition for the non-linear finite element pipeline.

Figure 4.7a illustrates the tension at the interfaces which, following the aforementioned figure, also are identical. Figure 4.7b shows the bending moments at the interfaces, which are slightly higher in the lumped-mass pipeline.

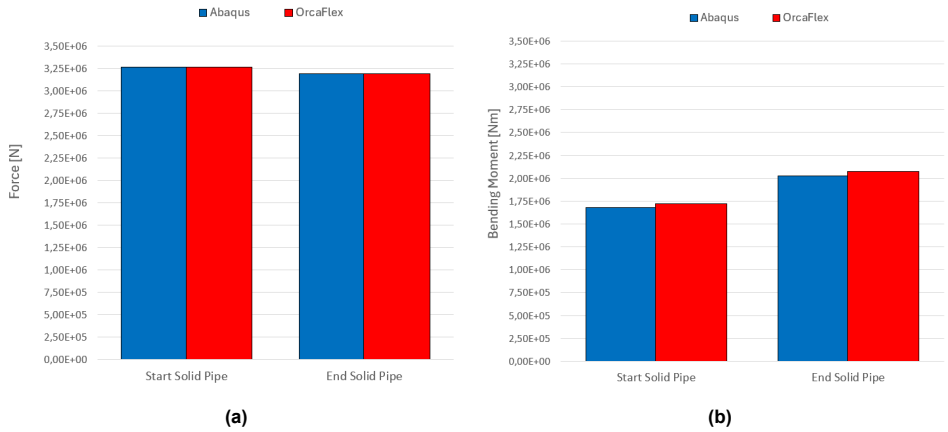


Figure 4.7: Comparison of the loads at the ILS interface a) Tension b) Bending Moment

Reaction Forces

As mentioned in Section 3.2.3, the RB reaction forces are usually check in order to ensure that the integrity of the RBs are maintained. For this research project, these are rather compared between models in order to assess the overall pipeline behaviour and whether the modified RBs in Abaqus approximate the behaviour of those in OrcaFlex. Figure 4.8 now illustrates a comparison of the RB reaction forces in OrcaFlex to that of the pipeline modelled using line elements, with additional difference values. Figure B.26 shows additional comparison of OrcaFlex with both pipelines containing solid elements.

Overall, the RB reaction forces pattern stays the same. Due to the RBs in OrcaFlex having the ability to rotate, the load of the pipeline is perfectly distributed and their reaction forces per RB, e.g. RB S1H1 and RB S1H2, are identical. Despite replicating the rollerboxes configurations from OrcaFlex into Abaqus, the load of the pipeline is not perfectly distributed. For RBs where the first hinge H1 has a lower reaction force than OrcaFlex, the second hinge H2 typically has a higher reaction force - and vice versa.

Figure 4.8a shows all of the RBs in the overbend section. After RB S13H2, the pipeline loses contact with the stinger. For RBs such as RB V15H1 and RB V15H2, the reaction forces are identical in both Abaqus and OrcaFlex. For most RBs this is not the case.

Figure B.26b shows the reaction forces of the RBs up until where the solid pipe of 80 $[m]$ ends. Unlike for the previous pipeline, these reaction forces barely distribute the load of the pipeline evenly for each RB. The reaction force at RB S7H2 is surprisingly high. This is due to boundary condition effects, as it is where the pipeline transitions from solid to line elements.

Figure B.26c shows the reaction forces of the RBs up until where the solid pipe of 95 $[m]$ ends. Following the previous shorter solid pipe section, the reaction force at RB S7H2 has reduced since the transition area has moved. Now, the effects of this can be seen at RB S9H1, also having an effect the neighbouring RBs.

As most H1 hingers are lower than their respective H2 hinges, it seems that the set declination angles in Abaqus are slightly smaller - the RB is more horizontal - than needed for the load of the pipeline to

be evenly distributed. Due to the reaction force of hinge H2 being bigger than that of hinge H1, the pipeline is sitting more on H2 than on H1. If its declination angle is increased - rotating the RB closer to the vertical axis - the reaction force at H1 would increase, and that at H2 would decrease.

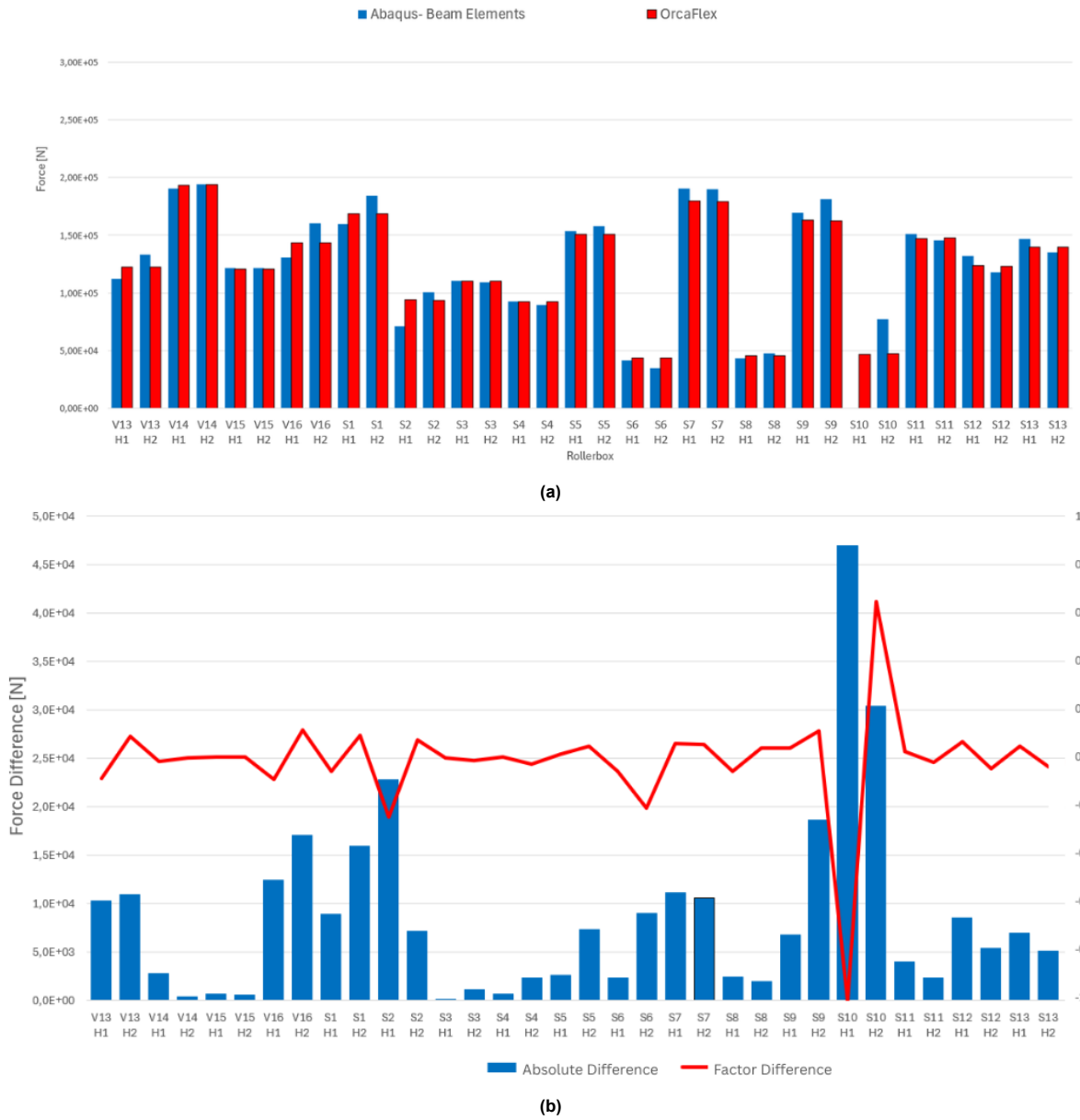


Figure 4.8: Rollerbox reaction force comparison of OrcaFlex with Abaqus: a) Line-Elements b) Difference values

Following these, despite modifying the RBs in Abaqus in order to closely replicate the rollerboxes configuration from OrcaFlex in Abaqus, their reaction forces do not completely align. The rollerboxes in Abaqus might need to rotate a few extra fractions of a degree in order to properly distribute the load of the pipeline, but can not do so due to the boundary conditions set. Therefore, slight differences in the reaction forces occur.

Strain in the pipeline

The main result that is looked upon after completing analyses are the strains. The strain gives a measure on how much relative elongation is taking place, and has to be monitored in order to maintain integrity. Strains can be calculated and plotted in many different ways, and it is therefore crucial to know which strain will be compared between the pipelines in Abaqus and OrcaFlex. The standard

practice for pipeline installation in OrcaFlex is to plot the *Max Pipelay von Mises* strains, which is a combination of axial and hoop strain. whereas Abaqus has no equivalent Max Pipelay von Mises strain output. Equation 4.5 shows the equation used to calculate this in OrcaFlex, where ϵ_{zz} is the axial strain due to direct tensile strain and bending strain, ϵ_{cc} is the hoop strain, and $\epsilon_{zz} \cdot \epsilon_{cc}$ is the interaction factor that accounts for the coupling effect between them [21].

$$\epsilon_{vm} = \sqrt{\epsilon_{zz}^2 + \epsilon_{cc}^2 - \epsilon_{zz} \cdot \epsilon_{cc}} \tag{4.5}$$

The standard practice for the strain output during pipeline installation in Abaqus is only that of axial strains. For pipeline installation, especially those in the overbend section, axial strains are leading due to the high tensions and bending moments taking place due to the tension applied via the vessel and tension, and also due to the pipeline bending over the stinger. Additionally, the overbend section typically lies near or even above the waterline, thus strains due to pressure difference (hoop strains) are significantly less. Therefore, further strain comparisons are done by means of axial strains.

The axial strain output from OrcaFlex is that where, for every cross section, the tensile strain $\pm$ bending strain has the highest (thus worst) output value [21]. This can be either at the top-most or bottom-most outer diameter, and its direct output is referred to as the *Worst ZZ Strain*. Abaqus on the other hand does allow for output selection for the various cross-sections, choosing specifically to output strains from either the top or bottom of the pipe. In order to stay consistent with OrcaFlex, the strains at both pipe top and pipe bottom are extracted, and the highest (worst) values per cross-section are used. As the stress-strain values used for the plastic part of the material definition in Abaqus are *true*-stresses and *logarithmic strains* - also mentioned in Section 3.2.2, logarithmic strain outputs are used. This is common for such practices [23], and also standard within Allseas. For each cross section, the highest (worst) value is selected and classified as *Worst LE11* strain.

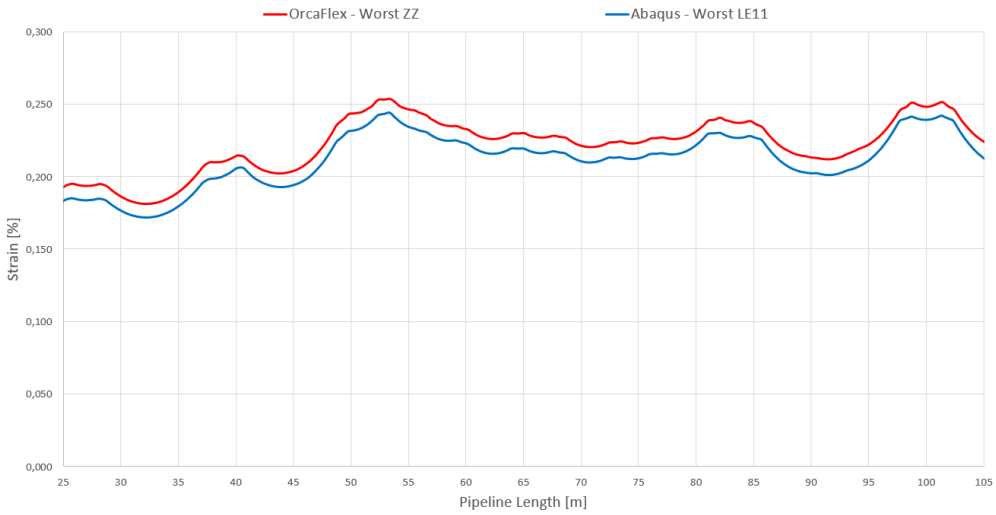


Figure 4.9: Axial strain components comparison for homogeneous line-element pipeline

The following components of the multi-body model and of the non-linear finite element models, which could lead to eventual strain differences, checks out: The materials for both homogeneous are the same, as discussed in Section 4.1.1; The rollerboxes configuration is replicated and coincides, as discussed above; The RB reaction forces seem to also not be the cause, as these are higher in Abaqus despite the strain from OrcaFlex resulting to be higher. Following these, it can be concluded that this constant difference between the strains curves is mainly caused by the differences in rollerbox reaction forces. As seen in Figure 4.7b, the resulting bending moments in OrcaFlex are slightly higher. Since the axial strain comprises of both tensile and bending strains, the overall axial strains are higher for OrcaFlex. For the comparison using solid elements, *LE Max. Principal (Abs)* is extracted. The strain results of the pipe section with solid elements is now included in the previous graphs and illustrated in Figure 4.10. This graph relates to the strains at the top of the pipe section, where the strain is the highest.

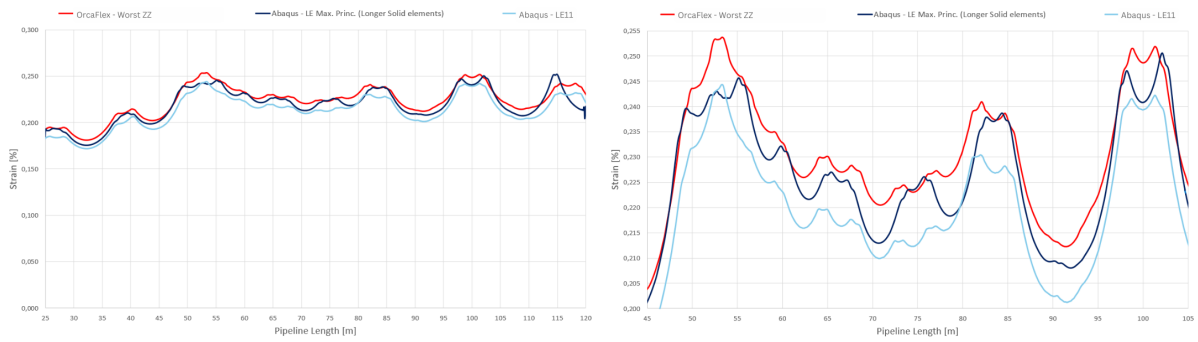


Figure 4.10: Strain comparison: a) OrcaFlex, Abaqus line and solid elements b) Zoomed In

Figure 4.10a now shows the resulting strains of the solid elements, which closely follows the curve of the line elements in Abaqus. Naturally, as the solid elements provide greater detail, its strain curve is slightly higher than that of the line elements in Abaqus especially where the pipeline comes into contact with the RBs - illustrated in greater detail in Figure 4.10b. Surprisingly, by OrcaFlex overestimating the axial strain compared to the line elements in Abaqus, its curve closely aligns with that of the solid elements.

Additionally, it should be noted that the axial strain curve of the solid elements in Figure 4.10a shows a sudden strain peak at the end of the graph. This strain peak is caused by a boundary condition effect which, as mentioned previously, is due to the transition of solid elements back to line elements at RB S10. Therefore, Figure 4.10b is graphed until after the second highest strain peak, which sits at RB S7.

Ovalization

By modelling the pipeline using the lumped masses, possible ovalization of cross sections throughout the pipeline is not taken into account. Therefore, the possible discrepancy of the change in centreline is also looked upon using 3D non-linear finite elements in Abaqus. Figure 4.11 illustrates the change in centreline when looking at the cross section of a pipeline.

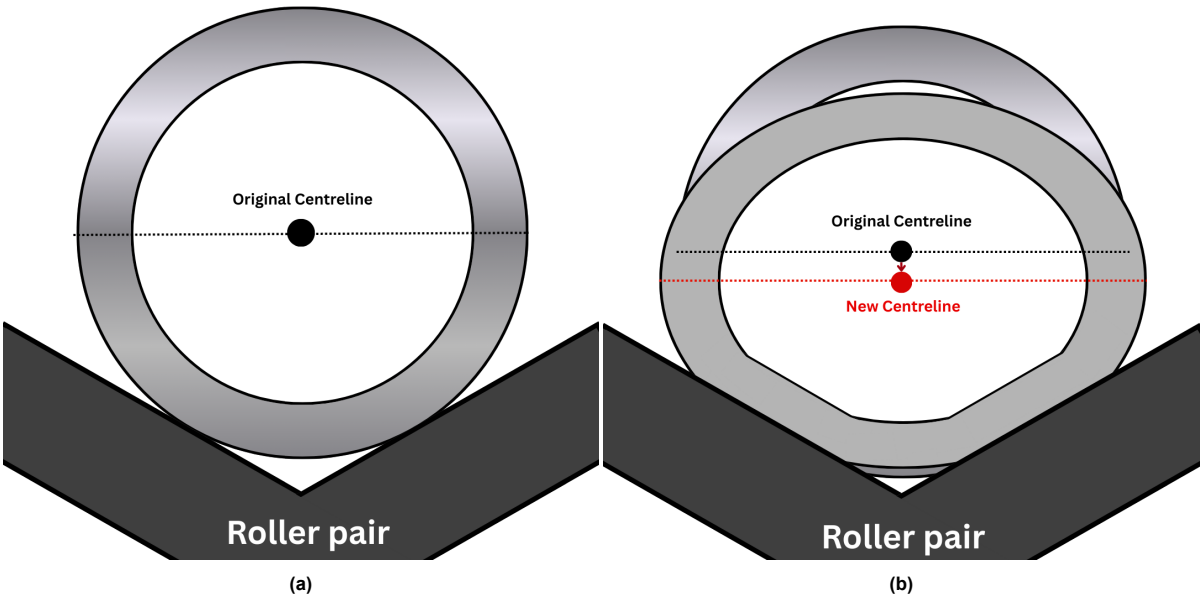


Figure 4.11: Change in the centreline by means of ovalization visualized: a) No ovalization b) Ovalization

Figure 4.12 illustrates the ovalization of the solid pipe section in the pipeline along its length by means of showing the height of the centreline. For this research project, the pipe outer diameter is of 508 $[mm]$. In the case of no ovalization, its centreline curve will lie at 254 $[mm]$, and its bottom and top wall at 0 and 508 $[mm]$ respectively. As mentioned, this does not take place when modelling the lumped-mass pipeline, thus the corresponding centreline stays the same throughout the overbend section in Figure 4.11a. Here, the centreline of the non-linear finite element pipeline also shows no fluctuation, implying no ovalization. Figure 4.12b now illustrates that, for the pipeline section modelled in Abaqus using solid elements, there is in fact a fluctuation of the centreline. The height of the centreline from the bottom wall decreases throughout the length of the solid-element pipe sections as ovalization takes place. Despite this fluctuation, the order of this fluctuations is of a fraction of a millimetre. Therefore, despite the presence of ovalization, it is deemed as insignificant and should not be accounted for when modelling the lumped-mass pipeline.

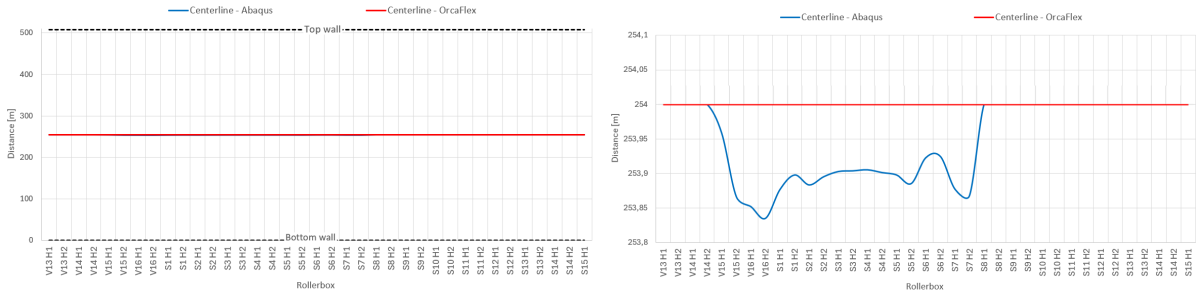


Figure 4.12: Change in the centreline comparison of Abaqus and OrcaFlex: a) Centreline between top and bottom wall b) A zoom in into the centreline change

4.2. Inline Structure Positions

As mentioned in Section 3.3, the governing RB for this stinger configuration is determined by the RB causing the highest strain peak. As discussed in Section 4.1 and illustrated in Figure 4.2a, RB S1 is governing. Subsequently, the two exact locations in which the ILS has to be placed for static analyses are determined using a Python script. As mentioned in Section 3.3.3, the ILS is initially placed right before coming into contact with RB S1, at 225 $[m]$ from the first node of the pipeline in OrcaFlex. It is then displaced in increments of $0,1$ $[m]$, over a length of 40 $[m]$. For every increment, a static analysis is done. The results are then displaced for all 400 increments, starting at the first displacement of 0 $[m]$, up to 40 $[m]$. A total of 401 OrcaFlex models were created and analysed after running the Python script. These consists of both the first model at 0 $[m]$ displacement and the last model at 40 $[m]$, both illustrated in Figure 4.13a and 4.13b respectively.

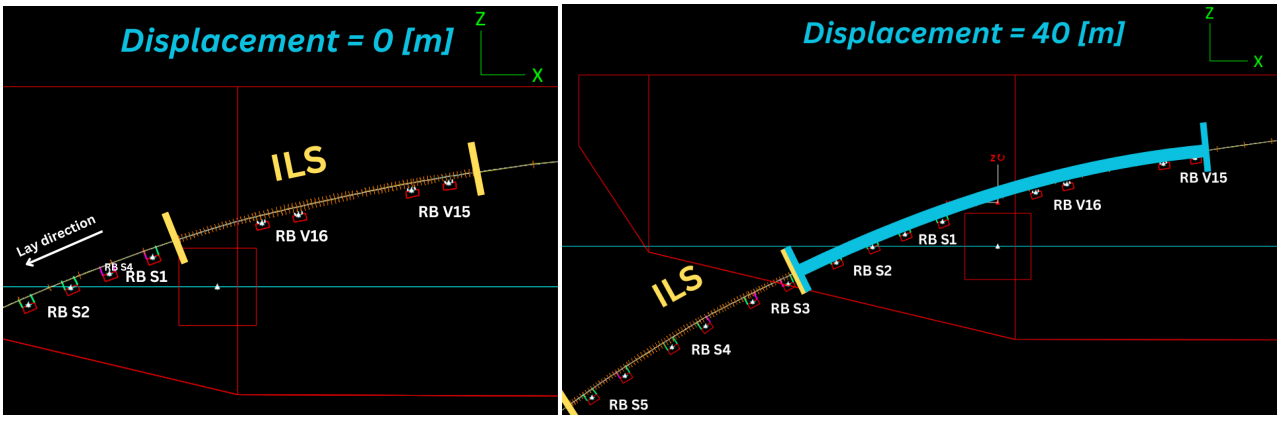


Figure 4.13: ILS displacement illustration: a) Start model b) End

4.2.1. Max. Rollerbox Reaction Force Position

Figure 4.14 illustrates the rollerbox reaction forces of RB V16, S1, and S2 for the complete 40 $[m]$ displacement. These are plotted for their corresponding displacement, with each data point being a completed static analysis. As expected, the curve of RB S1 has the highest peak, with a max rollerbox reaction force of 1607 $[kN]$ at a displacement of $15,2$ $[m]$ from the starting position - $240,2$ $[m]$ from the first node of the lumped-mass pipeline in OrcaFlex. This exact position of the ILS is illustrated in Figure 4.15.

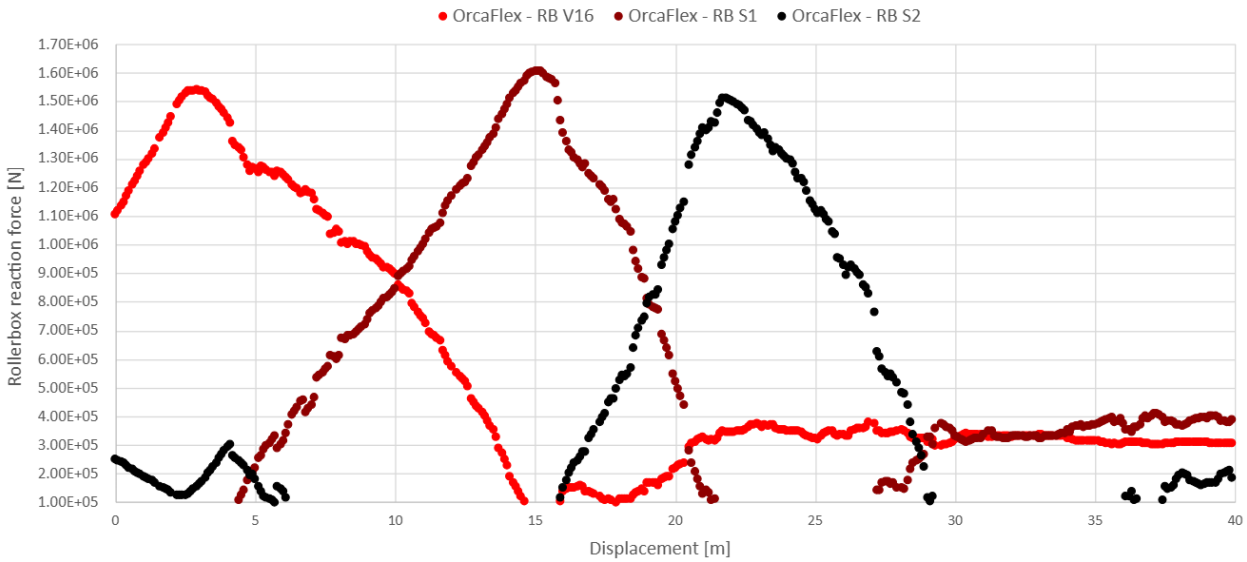


Figure 4.14: Rollerbox reaction forces for RB V16, RB S1, and RB S2 during all displacements

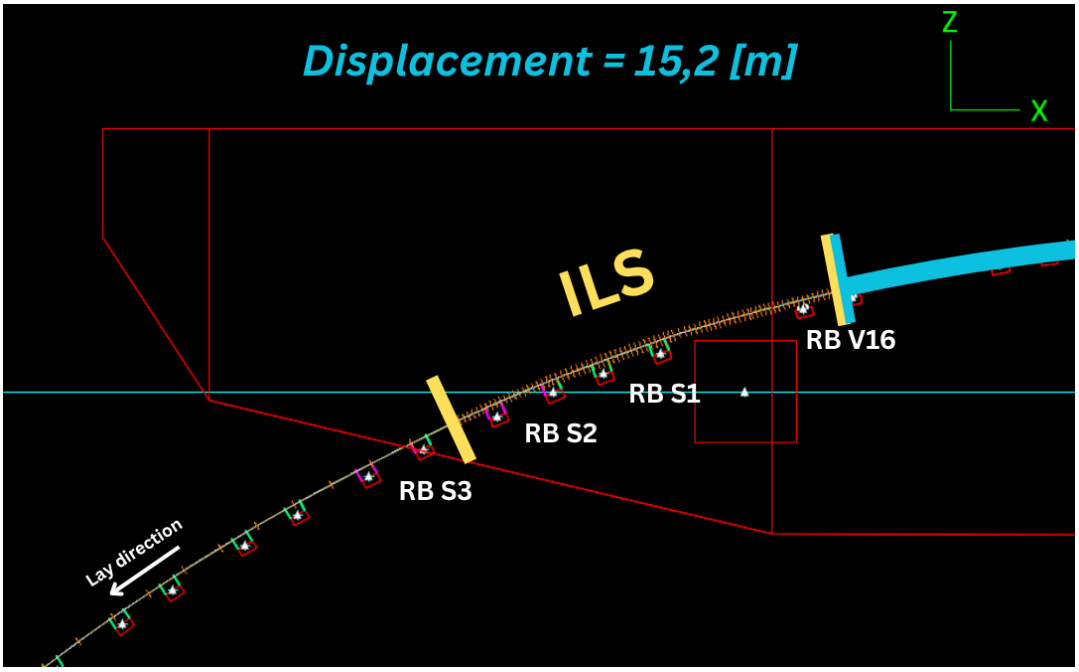


Figure 4.15: ILS position for the Max Rollerbox reaction force difference

4.2.2. Max. Bending Angle Difference Position

Figure 4.16 illustrates the bending angle differences of the ILS for every displacement. This curve shows a max bending angle difference of $7,38$ [°] at a displacement of $22,5$ [m] from the starting position - $247,5$ [m] from the first node of the lumped-mass pipeline in OrcaFlex. This exact position of the ILS is illustrated in Figure 4.17.

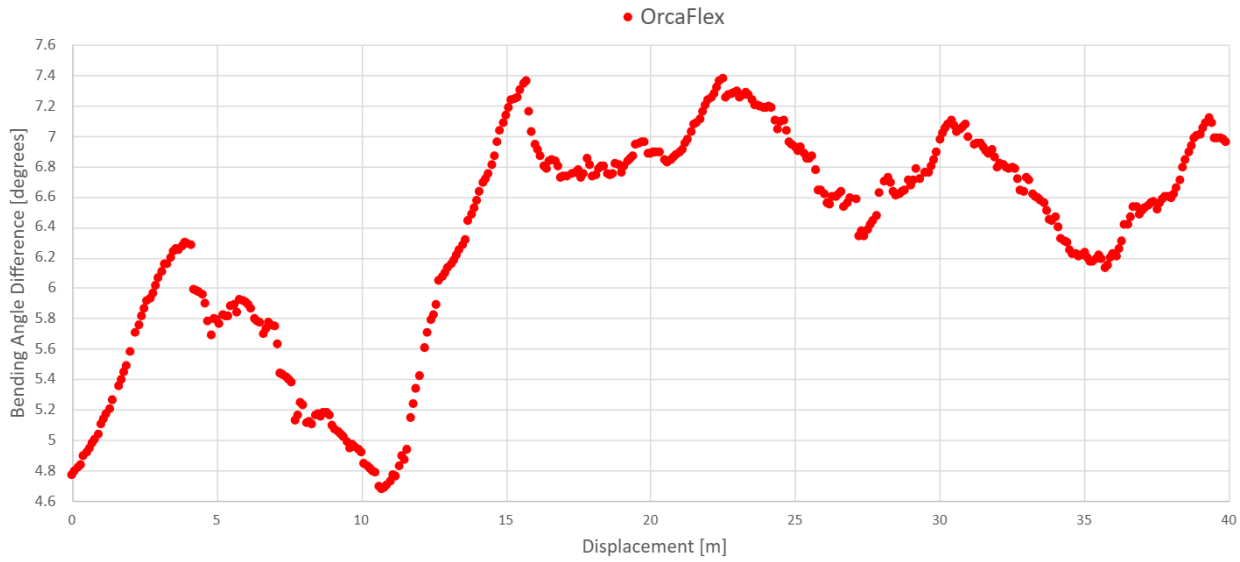


Figure 4.16: Bending angle difference of the inline structure ends during all displacements

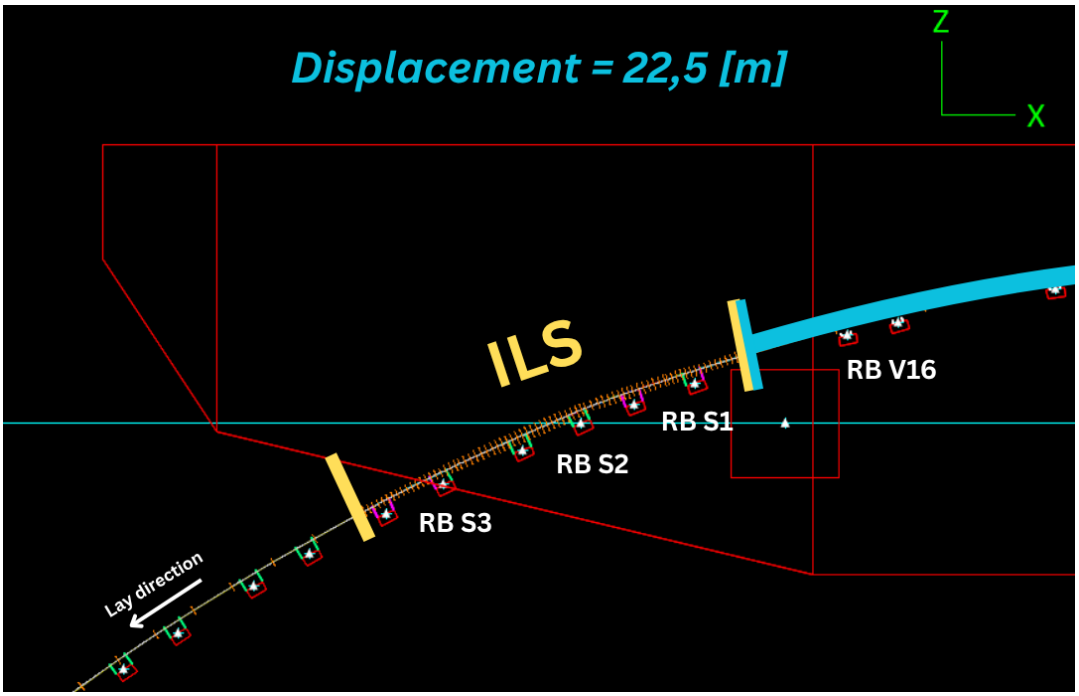


Figure 4.17: ILS position for the Max Bending angle difference

4.3. Inline Structure Difference

The procedure to replicate the rollerboxes configuration from OrcaFlex into Abaqus, explained in Section 4.1, is also applied to replicate the rollerboxes configurations for both ILS cases. The rollerbox dimensions naturally stay the same, thus Table B.12 is still used for respective L measurements. Yet, the resulting $\phi_{MainHinge}$ differs for both cases from the homogeneous pipeline case. The new declination angles for the Max RB Reaction Force and Max Bending Angle Difference case can be found in Tables B.15 & B.16, and B.17 & B.18, respectively. These declination angles are also illustrated in Figure B.28 and B.30. The results of both ILS cases are discussed separately, followed by an overall conclusion.

For both pipeline declination angle comparisons, there will be a gap in the curve of Abaqus. This gap relates to the ILS, and is left out as the declination angles can only be extracted from line elements. Additionally, strain curves will also have a gap. This gap also relates to the ILS, as the strains within the ILS are not relevant for this research project - only the strains of adjacent pipe sections are of interest. To ensure simplicity, tidiness, and consistency, the resulting graphs and curves corresponding to the lumped-mass pipeline will be coloured red, and labelled as “OrcaFlex”. Those corresponding to the non-linear finite elements pipeline will be coloured (a shade of) blue, and labelled as “Abaqus”.

4.3.1. Max RB Reaction Force case

As done for the homogeneous pipeline case, a static analysis on the corresponding lumped-mass pipeline in OrcaFlex is first performed, in order to replicate the rollerboxes configuration in Abaqus. The resulting rollerboxes configuration is illustrated in Figure 4.18, with respective declination angles illustrated in Figure B.28. The Python script used to recreate the rollerboxes configuration in Abaqus can be found in Appendix C.3. The Abaqus and OrcaFlex ILSs are illustrated side by side in Figure B.27.

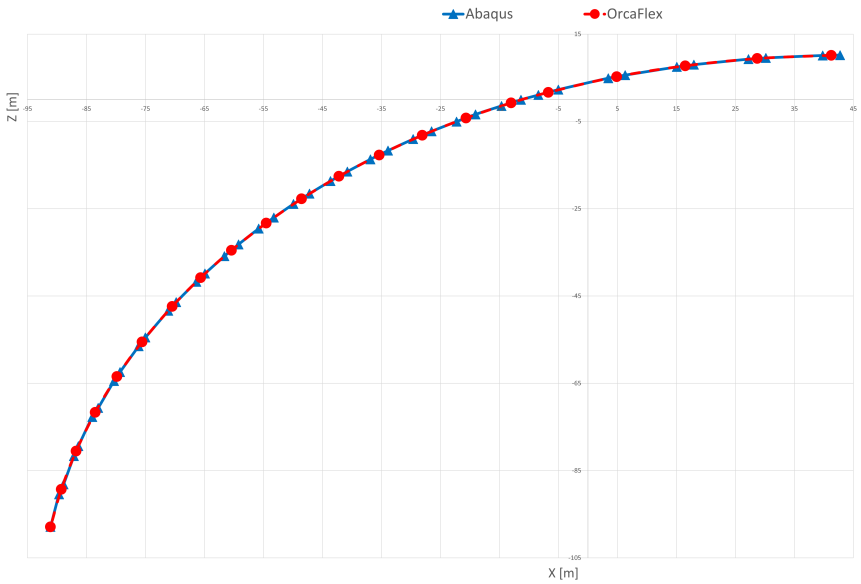


Figure 4.18: Max RB Reaction Force case: Rollerboxes configuration comparison

Declination Angle

The resulting declination angle of the pipeline is illustrated in Figure 4.19. The sudden break in the curve of Abaqus is where the ILS is placed. The declination angle curves are results from the line-element part of the pipeline, thus excluding the ILS.

The declination angle of both pipelines aligns perfectly for the overbend section, but differs at the end of the graph. The section where the pipeline reaches its declination angle peak (inflection point) is where the pipeline transitions from overbend section to sagbend section.

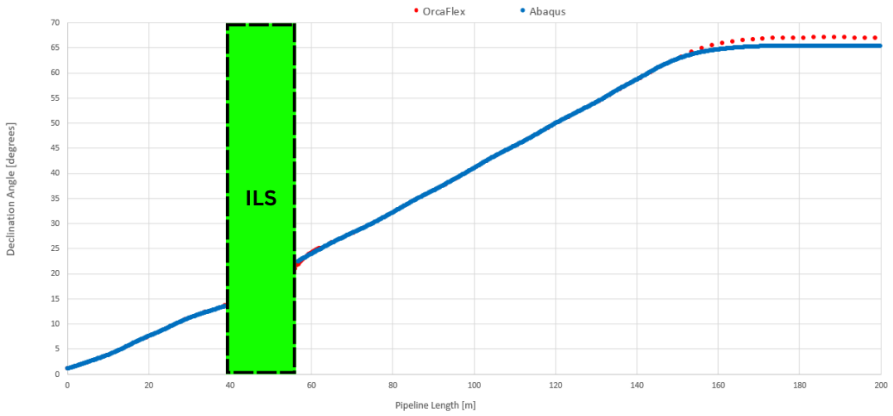


Figure 4.19: Max RB Reaction Force case: Overbend section Declination Angle comparison

Top & Bottom Tension

The resulting top and bottom tensions for the lumped-mass pipeline and the non-linear finite element pipeline are illustrated in Figure 4.20. The bottom tensions are identical, since this is the force that is replicated and applied. The top tension is a resulting force from the bottom tension pulling on the pipeline which is fixed at its first node. The top tensions therefore differs a bit, as OrcaFlex slightly overestimates it.

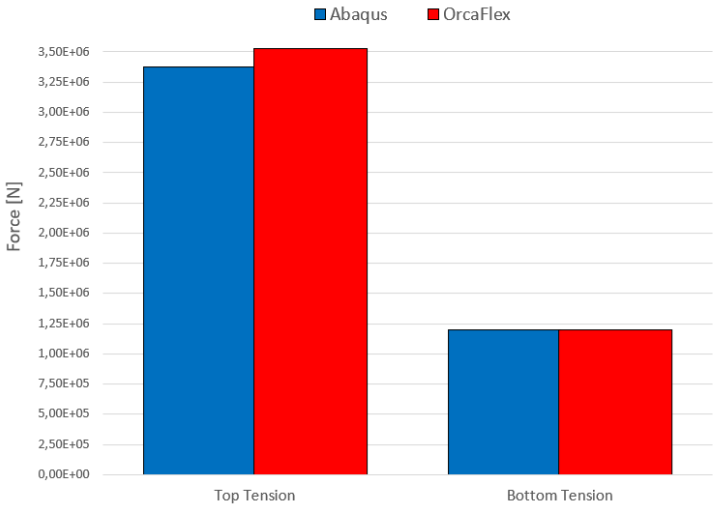


Figure 4.20: Max RB Reaction Force case: Top & Bottom Tension comparison

Loads at the ILS interface

The loads at the ILS-pipeline interface are illustrated in Figure 4.21, where both tensions and bending moments are mostly overestimated when modelling the ILS within the lumped-mass pipeline.

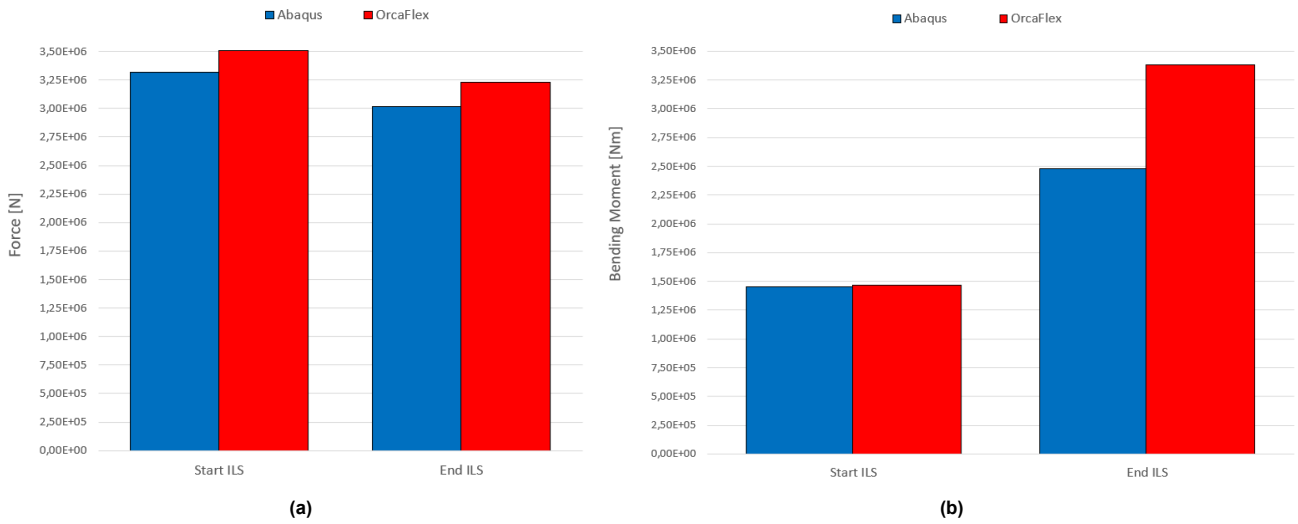


Figure 4.21: Max RB Reaction Force case: ILS interface loads comparison a) Tension b) Bending Moment

Rollerbox Reaction Forces

The RB reaction forces are illustrated in Figure 4.22. These RB reaction forces follow the same pattern, but as previously discussed, the load of the pipeline is not evenly distributed between RBs in Abaqus.

There is a Max RB reaction force at RB S1H1 in Abaqus. This RB reaction force is higher than that in OrcaFlex, as the reaction force of RB S1H2 is 0 [N]. Thus, the ILS does not make contact with this part of the rollerbox, and is being fully upheld by RB 12H1.

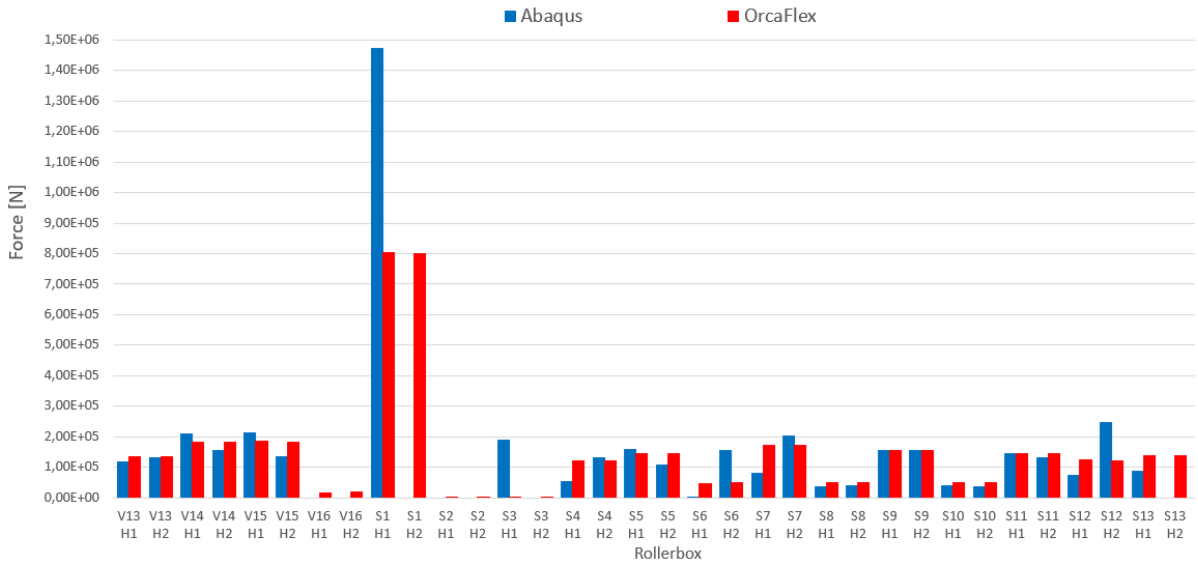


Figure 4.22: Max RB Reaction Force case: Rollerbox reaction force comparison

Axial Strain Comparison

As mentioned in the homogeneous pipeline comparison, the strain comparison will be a comparison of the axial strain - *Worst ZZ* strain in the lumped-mass pipeline in OrcaFlex, and *Worst LE11* strain in the non-linear finite element pipeline in Abaqus. Figure 4.23 illustrates this comparison, with a gap within the curves relating to the ILS.

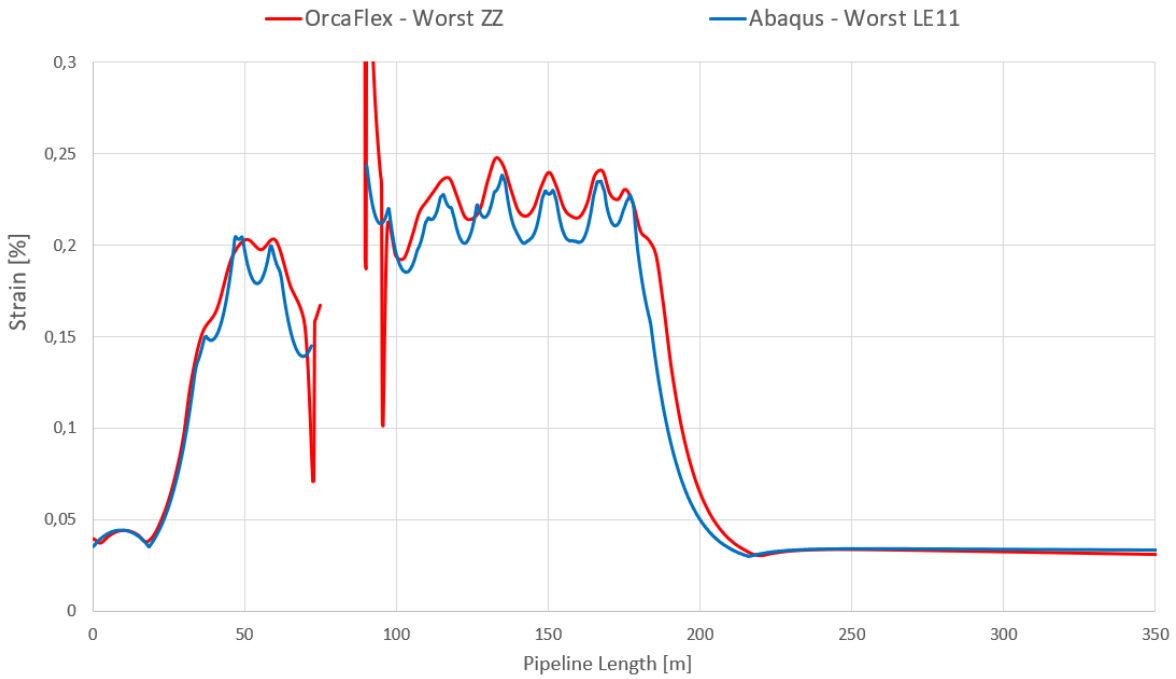


Figure 4.23: Max RB Reaction Force case: Axial strain comparison

Following the discussion of strain in the previous homogeneous pipeline case, the strain curve of the lumped-mass pipeline closely approximates the strain curve of the non-linear finite element pipeline. Strains further away from the ILS are closely approximated, yet still slightly overestimated in a conservative matter. The strains within ± 5 $[m]$ on each side of the ILS have a significant difference. This is mainly due to a consistency issue between the ILS models themselves. As seen in Figure 4.24, the lumped-mass ILS model contains extra components *A-A* and *T-T* that were not modelled using 3D non-linear finite elements. A clear difference of $9,24$ $[m]$ in total can be seen between both ILS models.

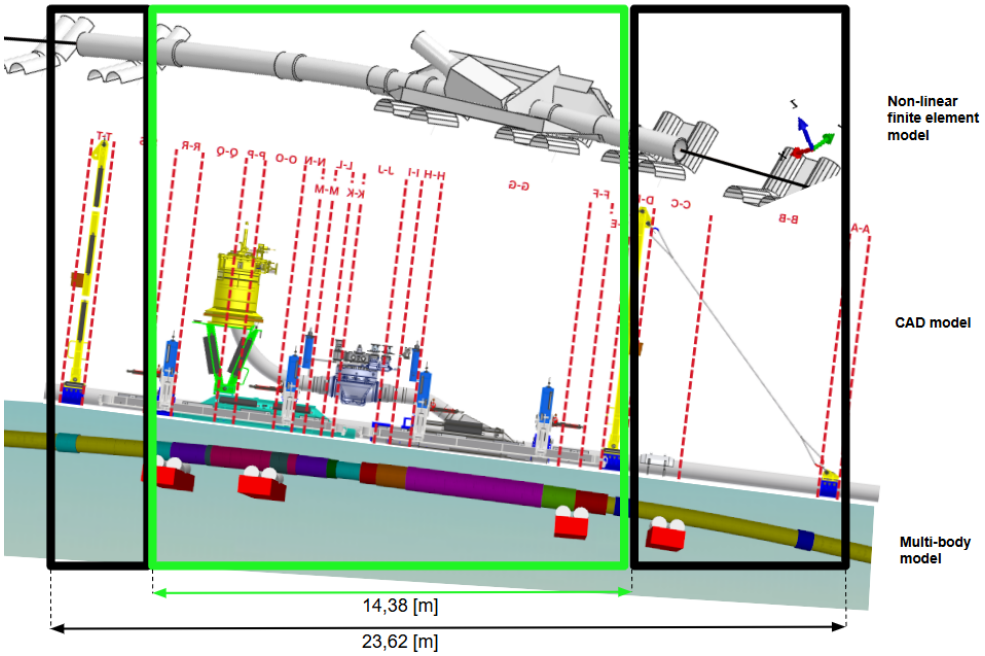


Figure 4.24: Inline Structure model comparison

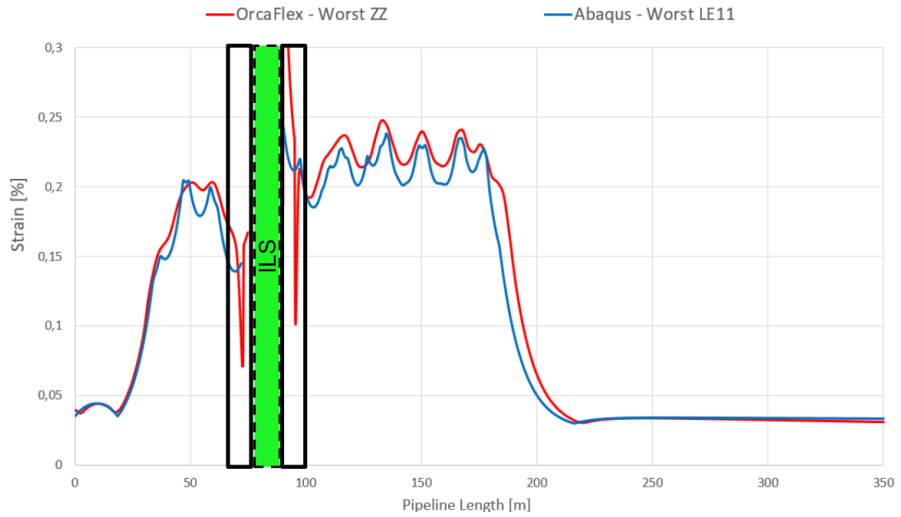


Figure 4.25: Max RB Reaction Force case: ILS modelling differences

Figure 4.25 illustrate the strain peak differences caused by the extra components in the lumped-mass model. By not taking these extra components into account in the 3D non-linear finite element ILS, they will indefinitely lead to (strain) result differences. Figure 4.26 now illustrates the resulting axial strains after the extra component axial strains are removed and both ILSs are of equal length. The area covered for the ILS is that of the length of the ILS in the lumped-mass pipeline of $23,62$ $[m]$. These strains are now related to only pipe sections. The visible resulting axial strains from the lumped-mass pipeline in OrcaFlex closely approximate those resulting from the non-linear finite elements in Abaqus, yet slightly overestimates these in a conservative matter. The overall difference is as previously discussed due to the inherent extra bending moment resulting in Abaqus. The extra bending moment induced will lead to higher bending strains thus higher axial strains. Additional strain peak differences, such as that for the lumped-mass pipeline at 180 $[m]$ is due to the fact that, as seen in Figure 4.22 there is still contact with RB S13H2 that is not present for the non-linear finite element pipeline. This strain curve flattens therefore a few meters before that of the lumped-mass pipeline. It is therefore crucial to ensure and maintain consistency between both ILS models.

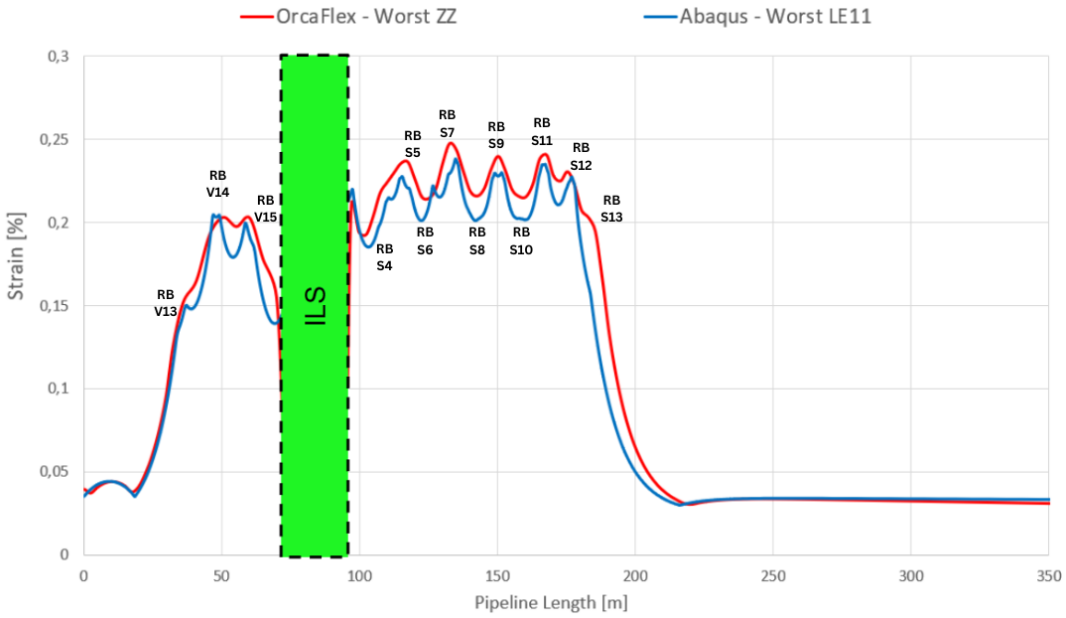


Figure 4.26: Max RB Reaction Force case: Axial strain comparison

4.3.2. Max Bending Angle Difference case

As done for the homogeneous pipeline case, a static analysis on the corresponding lumped-mass pipeline in OrcaFlex is first performed, in order to replicate the rollerboxes configuration in Abaqus. The resulting rollerboxes configuration is illustrated in Figure 4.27, with respective declination angles illustrated in Figure B.30. The Python script used to recreate the rollerboxes configuration in Abaqus can be found in Appendix C.4. The Abaqus and OrcaFlex ILSs are illustrated side by side in Figure B.29

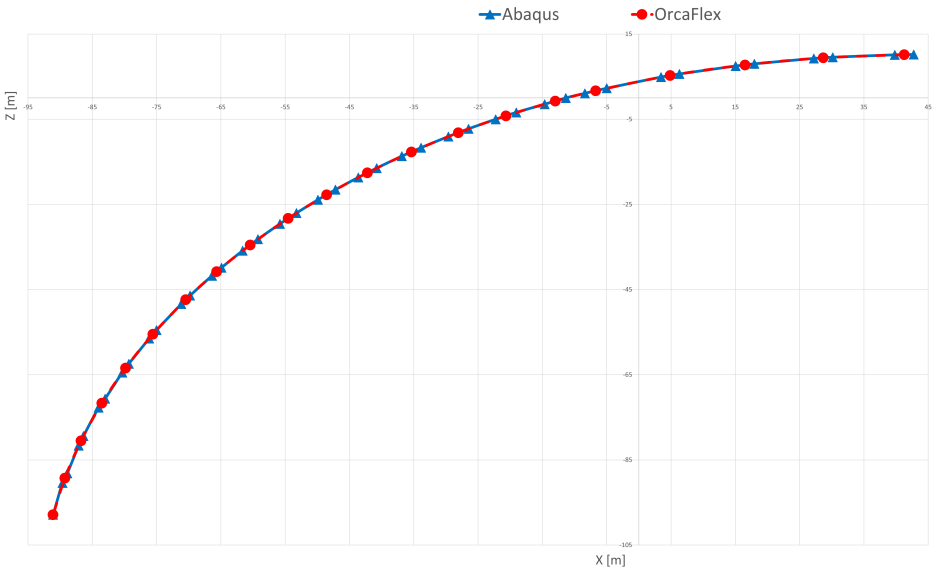


Figure 4.27: Max Bending angle difference case: Rollerboxes configuration comparison

Declination Angle

The resulting declination angle of the pipeline is illustrated in Figure 4.28. The sudden break in the curve of Abaqus is where the ILS is placed. The declination angle curves are results from the line-element part of the pipeline, thus excluding the ILS.

The declination angle of both pipelines aligns perfectly for the overbend section, but differs at the end of the graph. The section where the pipeline reaches its declination angle peak (inflection point) is where the pipeline transitions from overbend section to sagbend section.

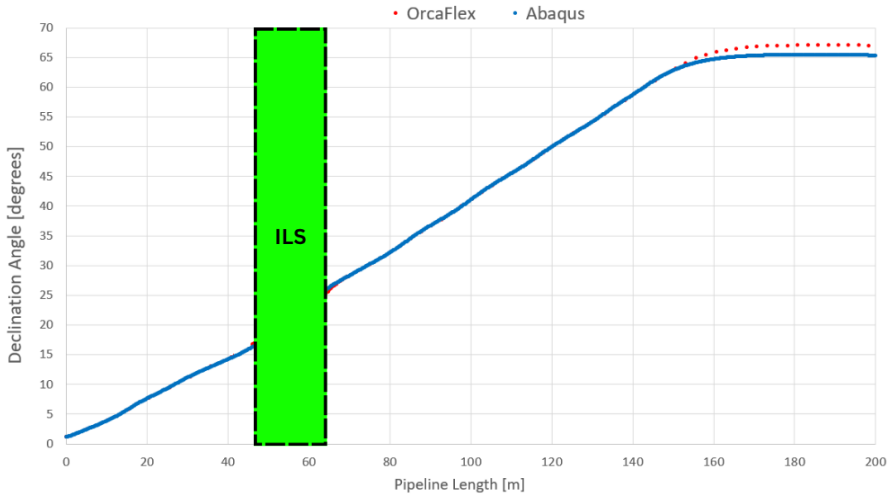


Figure 4.28: Max Bending angle difference case: Overbend section Declination Angle comparison

Top & Bottom Tension

The resulting top and bottom tensions for the lumped-mass pipeline and the non-linear finite element pipeline are illustrated in Figure 4.29. The bottom tensions are identical, since this is the force that is replicated and applied. The top tension is a resulting force from the bottom tension pulling on the pipeline which is fixed at its first node. The top tensions therefore differs a bit, as OrcaFlex slightly overestimates it.

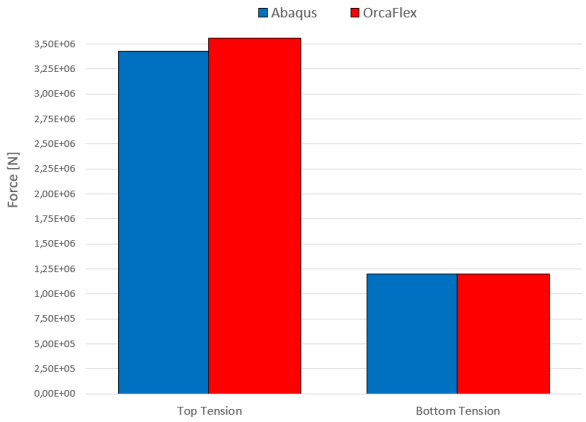


Figure 4.29: Max Bending angle difference case: Top & Bottom Tension comparison

Loads at the ILS interface

The loads at the ILS-pipeline interface are illustrated in Figure 4.30, where both tensions and bending moments are mostly overestimated when modelling the ILS within the lumped-mass pipeline.

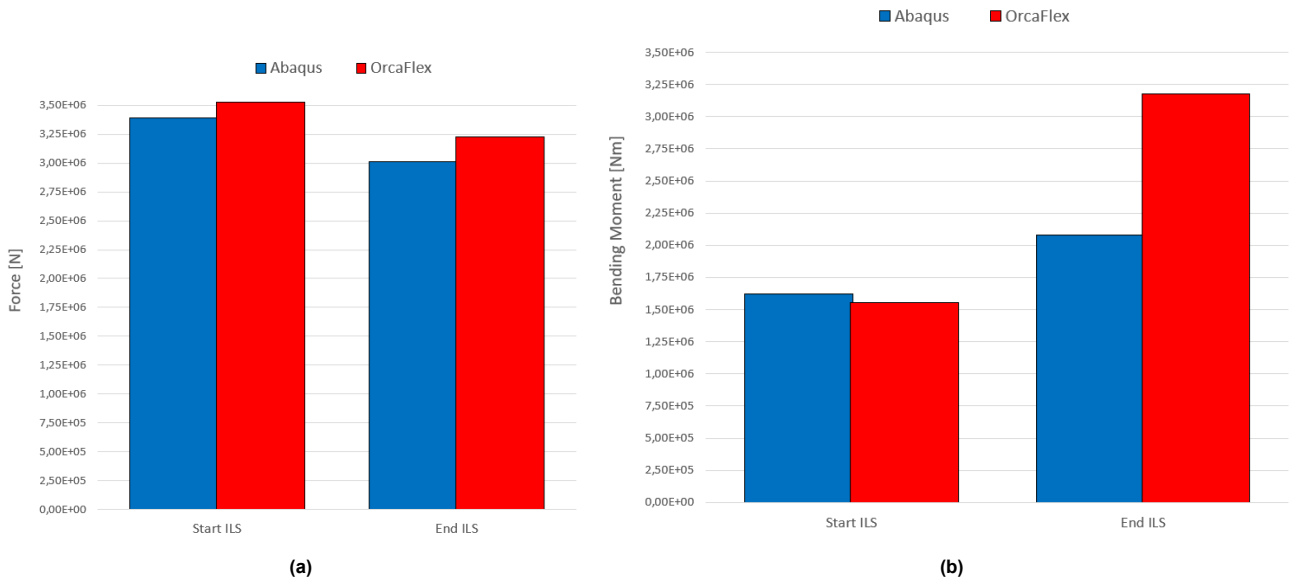


Figure 4.30: Max Bending angle difference case: ILS interface loads comparison a) Tension b) Bending Moment

Rollerbox Reaction Forces

The RB reaction forces are illustrated in Figure 4.22. These RB reaction forces follow the same pattern, but as previously discussed, the load of the pipeline is not evenly distributed between RBs in Abaqus. As the ILS is displaced, this pattern is naturally slightly different from the previous case.

There is a Max RB reaction force at RB S2H1 in Abaqus. This RB reaction force is higher than that of OrcaFlex, as the reaction force of RB S2H2 is 0 $[N]$. Thus, the ILS does not make contact with this part of the rollerbox, and is being fully upheld by RB S2H1.

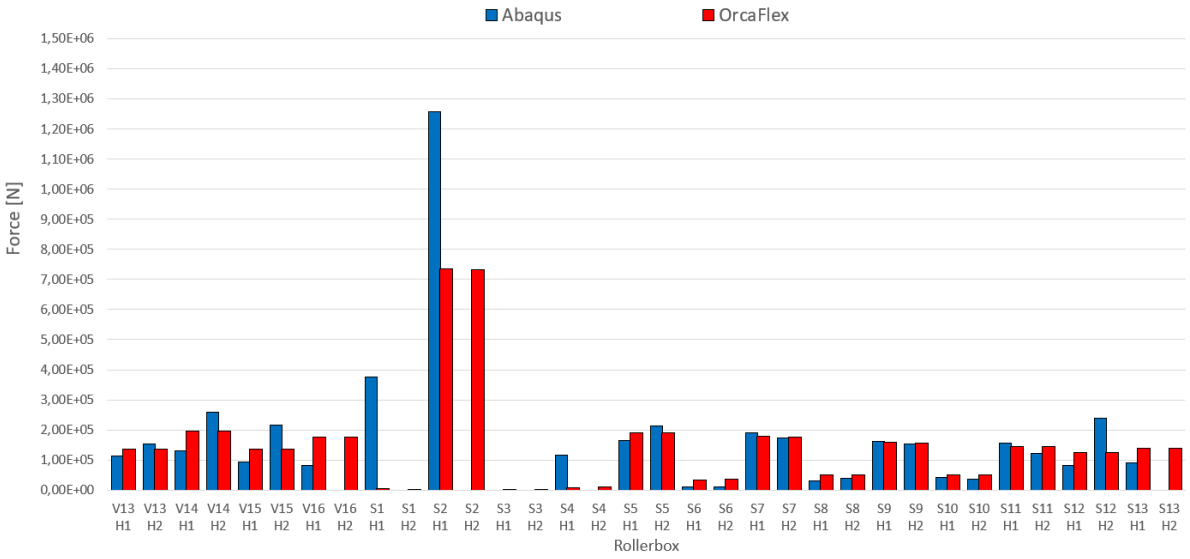


Figure 4.31: Max Bending angle difference case: Rollerbox reaction force comparison

Axial Strain Comparison

Following the aforementioned axial strain comparison of the previous case, the ILS strain gap is adapted accordingly as seen in Figure 4.32.

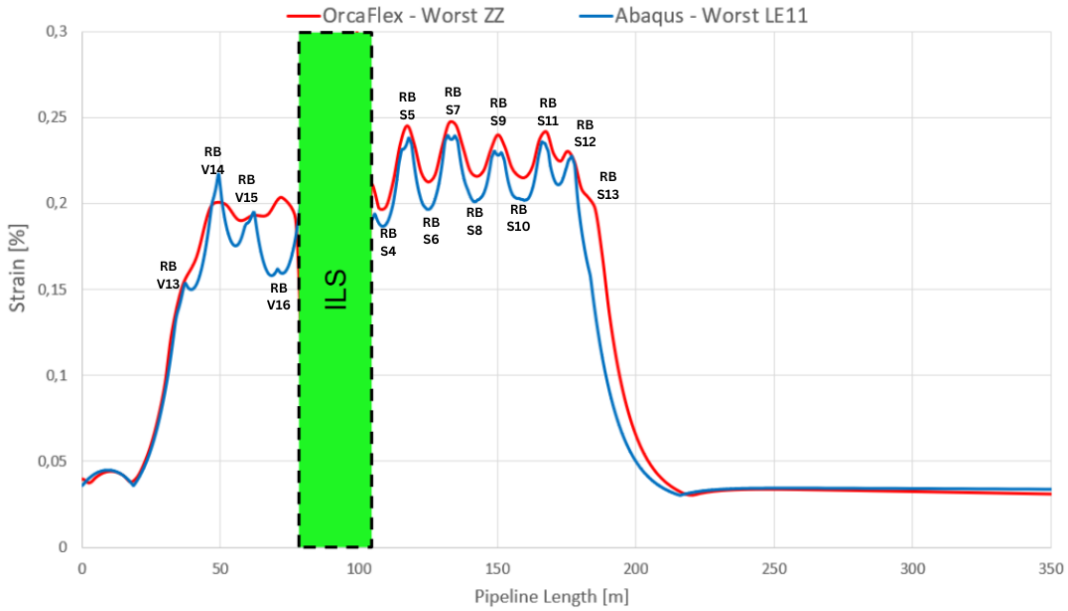
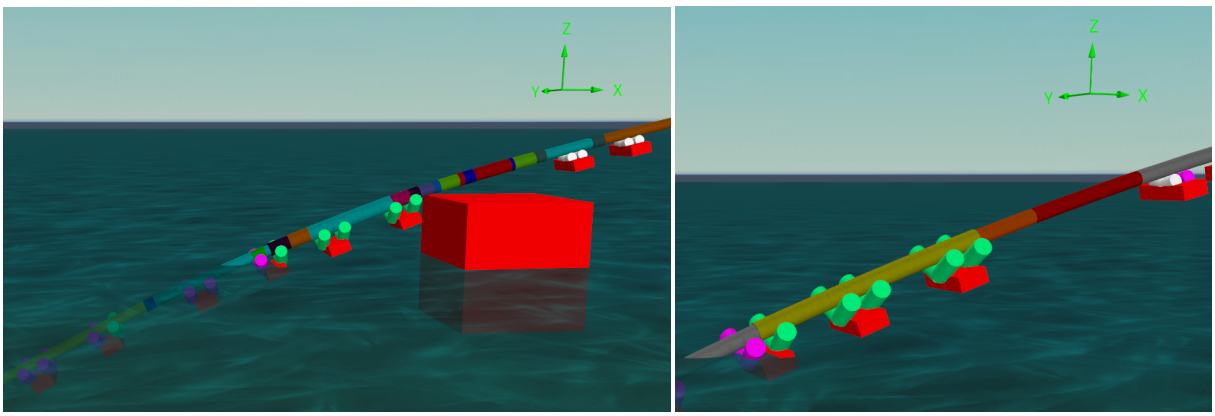


Figure 4.32: Max Bending angle difference case: Axial strain comparison

The resulting axial strains of the lumped-mass pipeline also closely approximate those of the non-linear finite element pipeline, yet slightly overestimate these in a conservative matter. Additional strain peak differences, such as that for the lumped-mass pipeline at 180 $[m]$ is due to the fact that, as seen in Figure 4.31, the pipeline in OrcaFlex still has contact with RB S13H2 but hat is not present for the non-linear finite element pipeline. This strain curve flattens therefore a few meters before that of the lumped-mass pipeline.

4.4. Time-efficient ILS Results

Following the aforementioned ILS comparisons, both methods discussed in Section 3.5 are used in order to recreate this ILS - resulting in two new (alternative) ILS, ILS #1 and ILS #2. Due to time constraints, these two alternative ILS are placed and compared only in one position in the overbend section, which is arbitrarily chosen as the position used in the Max Rollerbox Reaction Force case. Only the resulting RB reaction forces and strains are discussed, as the aforementioned discussions of declination angle, tensions and bending moments provide enough information. These are assumed to be similar, and not relevant for further discussions.



(a) (b)

Figure 4.33: Comparison of the inline structures: a) Original version b) Alternative version

As discussed in Section 3.5 , the sections used for the lumped-mass pipeline in OrcaFlex and their respective materials are derived following the aforementioned material-definition steps. The resulting ILS in OrcaFlex is illustrated in Figure 4.33b, next to the original ILS illustrated in Figure 4.33a. This figure illustrates that the alternative ILS now contains only 3 sections. The different colours are given arbitrarily and serve a visual purpose only. Both ILS #1 and ILS #2 will visually be the same as both are sectioned into 3 parts, yet the corresponding bending stiffnesses differ. The material curves of ILS #1 are obtained by taking the median bending moment-curvature of each section, and that of ILS #2 are obtained by taking the mean of the bending moment-curvature relation of each section. The resulting RB reaction forces are illustrated in Figure 4.34.

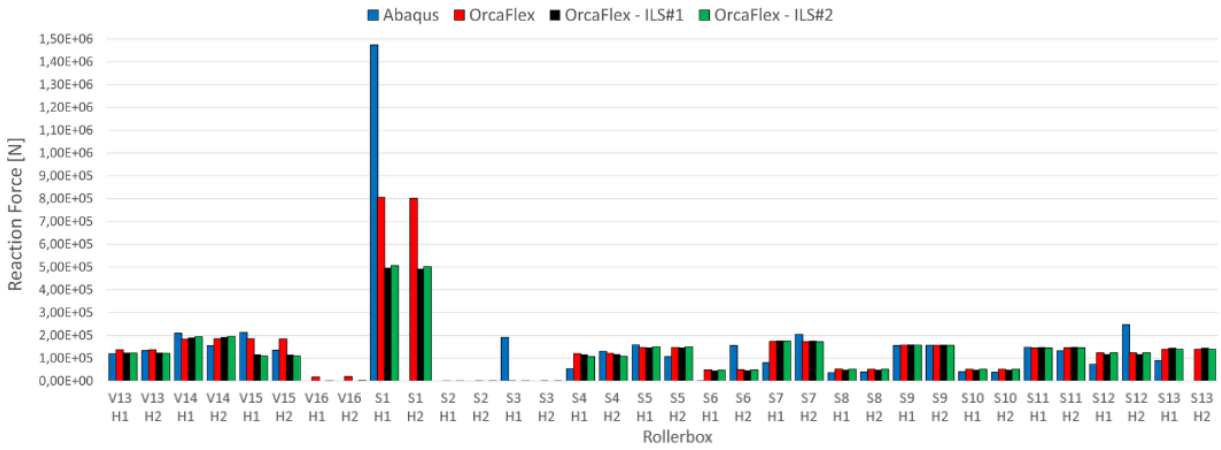


Figure 4.34: Rollerbox reaction forces for all four inline structures

As expected, the RB reaction forces are overall similar to that of the original ILS model in OrcaFlex. In addition, the load of the pipeline are evenly distributed between the rollerbox hinges. However, the

magnitude of the rollerboxes near and beneath both ILS #1 and #2 are significantly less than the original OrcaFlex model. This is due to the fact that the mass per unit length of each section of the original ILS was derived directly from the Inventor model. The mass per unit length of each section of ILS #1 and #2 were not taken into account, only their bending stiffness. However the resulting axial strains were closely approximated and are illustrated in Figure 4.35

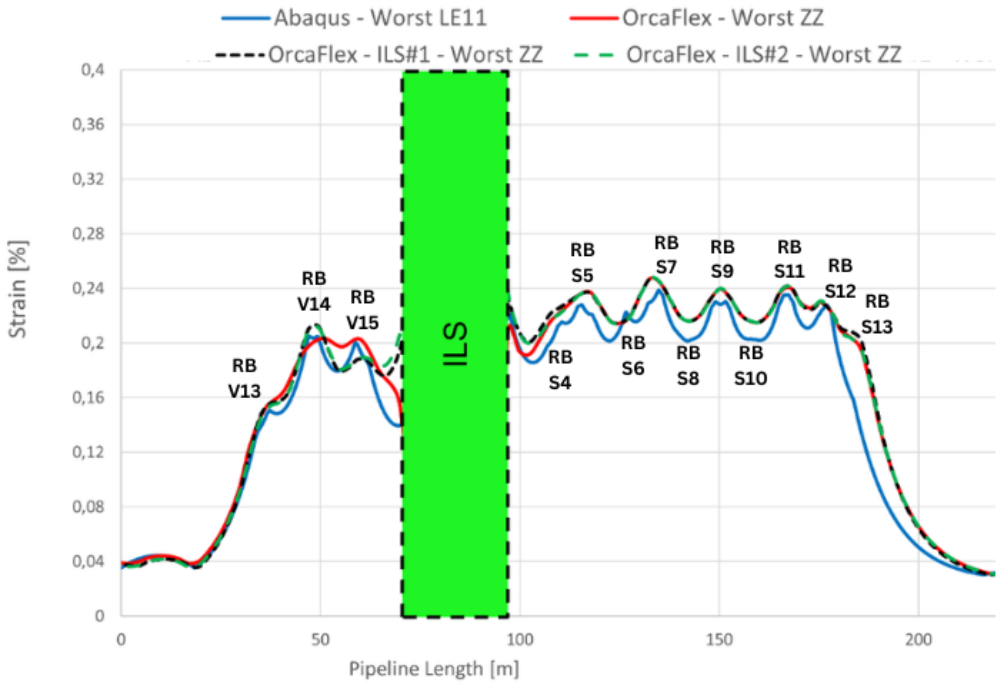


Figure 4.35: a) Comparison of the axial strains b) Zoomed In

Figure 4.35 shows both strains of the new ILS #1 and ILS #2 approximate that of the original ILS in OrcaFlex. Despite expecting closer results from the ILS #2 by taking the mean bending stiffness instead of the median bending stiffness, these axial strain results illustrate that there is barely any difference between using the mean and the median of the bending moment - curvature relation. The aforementioned RB reaction force differences have slight effects on the axial strains, which can be e.g at RB V14 and RB V15. The RB reaction forces of ILS #1 and #2 are higher at RB V14 and lower at RB V15 than the original OrcaFlex model. These result in a strain peak difference at corresponding location in Figure 4.35.

4.5. Implementation

The aforementioned steps are synthesized into the original framework illustrated in Figure 2.3. This new framework for the ILS creation method is illustrated in Figure 4.36.

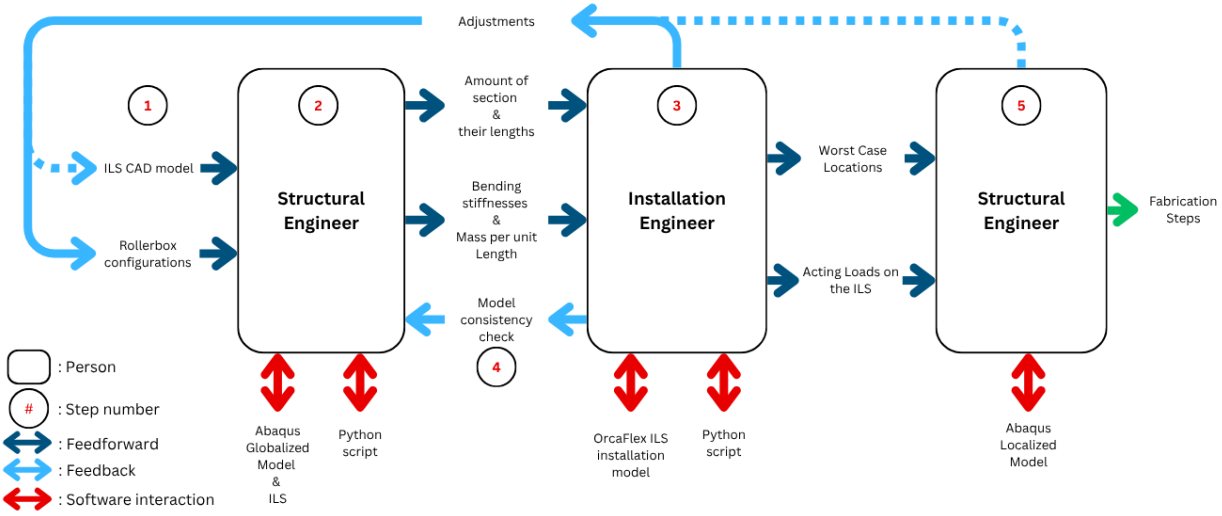


Figure 4.36: Detailed framework for the new ILS creation method

1. An in-house designer creates an ILS model based on the project specifications, and an installation engineer creates a multi-body model of homogeneous pipeline installation using lumped-mass pipeline in OrcaFlex. This model will be then used assess the stinger configuration e.g stinger length and stinger radius, to be used for this pipeline installation project. The installation engineer then runs a static analysis in order to provide RB data to the structural engineer.
2. The structural engineer receives two inputs:
   - The ILS CAD model from the in-house designer.
   - The RB Main Hinge coordinates and respective declination angles.

   To create the non-linear finite element model of the ILS installation in Abaqus, the structural engineer then used these inputs as follows:
   - The ILS CAD model is used to aid the creation of the 3D non-linear finite element ILS model.
   - The RB Main Hinge coordinates and respective declination angles are used to create the stinger in Abaqus using the Python script illustrated in Appendix C.2, as discussed in Section 4.1.3, using Equation 4.1, 4.2, 4.3, and 4.4.

   Before running static analyses, the ILS is placed within line elements in order to complete the pipeline. The modelling process of the ILS installation model is then completed by implementing simulation steps, boundary conditions, model interactions and loads in order to complete the model as discussed in Section 3.2.2.

   After completing static analyses, the structural engineer extracts the following:
   - The structural engineer divides the ILS into sections, as seen in Figure 3.20.
   - For each section, the median bending moment, curvature, and mass per unit length is extract following steps *a* through *f* mentioned in Section 3.5.1, using Equation 3.9 and 3.10, illustrated in Figure 3.22.
3. The installation engineer receives the following from the structural engineer:
   - Amount of sections used to discretize the ILS.

- For each section, the median bending moment, curvature, and mass per unit length. Additionally the respective section lengths and axial stiffness are also given.
- Properties of the regular pipe sections themselves, such as bending stiffness discussed in Section 4.1.1.

The installation engineer subsequently creates the ILS within a lumped-mass pipeline in a multi-body model in OrcaFlex. Once created, the Python script in Appendix C.1 is used in order to displace the ILS over the most governing rollerbox in increments of $0,1$ [m], and runs a static analysis at each position. The following is then extracted from both models. After running the Python script, the following is obtained:

- Completed static analyses models passing over the most governing RB, as illustrated in Figure 4.13a and 4.13b.
- Excel sheets, providing the locations of the most critical positions, i.e. the position with the highest Rollerbox reaction force, and the position where the ILS bends the most - discussed in Section 3.3 and 4.2.

4. After determining the two critical positions of the ILS in the overbend section, the lumped-mass and 3D non-linear finite element ILS are placed accordingly and static analyses are performed. Once complete, the following results are compared to ensure consistency between the models:
   - Declination angle of the pipeline is compared to ensure consistency between pipeline shape and RB configurations, as discussed in Section 3.2.3, Section 4.1, and illustrated in Figure 4.5, 4.19, and 4.28.
   - Axial strains are compared to ensure consistency between the ILS length, position, and overall pipeline behaviour.

   The resulting axial strains from the multi-body models in OrcaFlex are used to assess the behaviour of the pipeline during ILS installation, and whether its integrity is kept.
5. The structural engineer then receives the worst case locations and the acting loads at the ILS interface, in order to apply these directly on to the ILS model to run localized analyses, and assess if the integrity of the ILS and its components is kept following the loads applied.

! Following global analyses in OrcaFlex, it might occur that adjustments to the stinger configurations are needed in order to ensure installation feasibility. These adjustments are performed before running localized analyses, as seen in Figure 4.36 at Step 3.

! Following localized analyses in Abaqus, might occur that adjustments to the ILS itself are needed in order to ensure installation feasibility. These adjustments are performed before further fabrication steps are taken, as seen in Figure 4.36 at Step 5.

5

Discussion

This chapter will provide a set of interpretations of the results of the aforementioned chapter, in order to pave the way for the subsequent and final chapter, Chapter 6.

5.1. Modelling setup

The modelling setups have been compared, resulting differences and possible solutions have been proposed. Only two of these differences have been deemed relevant for this research, tackled and their solutions have been implemented.

In previous non-linear finite element models, the start of the pipeline used to be placed on top of the first rollerbox RB V13 as illustrated in Figure 3.2. The placing of the first node of the pipeline would have to be perfect. Having the first node too high (no contact) or too low (excessive contact) would lead to distorted contact behaviour at the first RB, also impacting contact behaviour of following RBs. To further avoid contact behaviour discrepancies, three additional support rollers have been added preceding RB V13, illustrated in Figure 3.4. The first node of the pipeline is placed further away from the RB V13. Now, the first ± 30 [m] of the pipeline are supported by these three support rollers, providing a leeway for the pipeline to settle horizontally before coming into contact with the rollerboxes. This mimics the setup found in both the actual pipelay vessels, Figure B.1a and B.1b, and that of the multi-body models in OrcaFlex, Figure B.8a.

In previous non-linear finite element models, the RBs used to be modelled as one rigid body, where only the declination angle was set at its reference point axis, Figure 3.3c. In other words, only main frame rotation illustrated in Figure 1.3 was taken into account. Only taking into account main frame rotation disregards the roller frame rotations, not allowing for even distribution of the load of the pipeline. To take into account roller frame rotation, the modelling of the RBs has been change to two rigid bodies as seen in Figure 3.5b. This is applied for all RBs besides RB S17, due to the fact that RB S17 does not have roller frame rotations to take into account, Figure B.15a. The individual bodies can now be rotated along their reference point axis, replicating the resulting RB configuration from OrcaFlex into Abaqus as discussed in Section 4.1.3, Figure 3.5a and 3.5b respectively. This resulted in identical RB configuration for all cases, illustrated in Figure 4.4, 4.18, and 4.27.

5.2. Homogeneous Pipeline

Homogeneous pipelines have been modelling in both multi-body models as non-linear finite elements models in order to compare the results of respective static analyses. As both models provide the ability to closely model and approximate realistic results of homogeneous pipeline, the aim of this comparison is to highlight inherent differences when modelling an identical pipeline using both models. A lumped-mass pipeline is compared to three non-linear finite element pipelines: one comprising of just line elements, one comprising mainly of line elements with an additional 80 [m] of a solid-element pipe section, and one comprising mainly of line elements with an additional 95 [m] of a solid-element pipe section. The line-element pipeline allows for a one-to-one comparison with the lumped-mass pipeline.

The 80 [m] solid-element pipe section provides high-fidelity results at a specific location, and the 95 [m] solid-element pipe section provides extra insights on the boundary condition effects where the pipeline switches from solid to line elements.

The results of the static analysis comparison for homogeneous pipelines show the following: After modifying the models of the RBs, an identical RB configuration is achieved by applying the resulting declination angles from OrcaFlex into Abaqus using Python script C.2, Figure 4.4. This subsequently leads to similar pipeline shapes as seen in Figure 4.5. The top tensions in Figure 4.6 result to be the same for all pipelines after applying the same bottom tension, regardless of it containing only line or solid elements. The interface tensions in Figure 4.7 is therefore the same at the solid pipe section interfaces, however there is a slight difference for the bending moments at the solid pipe section interfaces. The bending moments in the lumped-mass pipeline in OrcaFlex seem to be slightly higher. This difference in bending moment at the interfaces is caused by the difference in contact behaviour illustrated by the RB reaction forces in Figure 4.8. The RB reaction forces of Abaqus imply that the load of the non-linear finite element pipeline is not evenly distributed for all RBs. Most second hinges of the RBs seem to have a higher force magnitude, implying that the RBs should be slightly less horizontal to evenly distribute the load of the pipeline. By these RBs being more horizontal, the non-linear finite element pipeline is therefore bent slightly less than the lumped-mass pipeline, causing slightly higher bending moments in the lumped-mass pipeline as seen in 4.7b. This also becomes evident when plotting the axial strain in Figure 4.9, which comprises of both tensile and bending strain. The slight increase in bending moments throughout the lumped-mass pipeline induces higher bending strains, thus higher axial strains. The resulting higher-fidelity strains of the solid element pipe section seen in Figure 4.10 shows that using a lumped-mass pipeline model provides slightly-overestimated and conservative yet closely-approximated strains compared to those of non-linear finite elements. The additional 95 [m] pipeline illustrate that the solid-line element transition can cause for peak values for the RBs and for the strains, illustrated in Figure B.26c and 4.10a respectively.

5.3. Inline Structures

The exact location of the two critical positions of the ILS were first determined by means of displacing the ILS over the governing rollerbox RB S1, using the Python script C.1. The position with the highest rollerbox reaction force turned out to be whilst RB S1 was upholding the ILS in the middle, Figure 4.15. The position where the ILS bent the most turned out to be whilst the ILS was being upheld simultaneously by RB S1, RB S2, and RB S3, Figure 4.17.

As for the homogeneous pipeline case, the RB configurations in OrcaFlex were successfully replicated in Abaqus using Python scripts C.3 and C.4, illustrated in Figure 4.18 and 4.27 respectively. The resulting top tensions, Figure 4.20 and 4.29, and interface loads, Figure 4.21 and 4.30, were are over-estimated due to the lumped-mass ILS being an approximation of the 3D non-linear finite element ILS. The RB reaction forces seen in Figure 4.22 and 4.31 illustrate similar pipeline behaviour, yet uneven load distribution for the non-linear element pipeline. As seen for the homogeneous pipeline case, the RB reaction forces resulting from the multi-body model provide more realistic RB reaction force values due to the inherent ability to take into account the RB rotational behaviour. For the axial strains, Figure 4.23 illustrate that the aforementioned overestimated tensions and bending moments induce higher axial strains. Figure 4.23 also illustrate two strains peaks, which after further examination, are caused by components *A-A* and *T-T* which were not taken into account when creating the 3D non-linear finite element ILS as illustrated in Figure 4.24. The corresponding strain peaks are highlighted respectively in Figure 4.25. To correctly compare the strains, a new axial strain graph is provided in Figure 4.26 and 4.32.

5.4. Time-efficient Methods

Two new viable time-efficient methods were proposed to model an ILS within a lumped-mass pipeline, where the difference between these two lies in the material definition. The material data is extracted from the 3D non-linear finite element ILS in Abaqus itself. This ILS is first divided into sections as illustrated in Figure 3.21, where the first method proposed to use the median bending moment - curvature relation of each section, and the second method proposed to use the mean bending moment - curvature relation of each section. The exact steps for both methods are discussed in Section 3.5.1. The 3D non-

linear finite element ILS was divided into a total of three sections to recreated within the lumped-mass pipeline in OrcaFlex, as seen in Figure 4.33b, where each section only varied in its bending stiffness - bending moment - curvature relation.

Two ILS were created differencing in just the method the bending stiffness was extracted from the 3D non-linear finite element ILS in Abaqus, where ILS #1 uses the median bending stiffness and ILS #2 uses the mean bending stiffness of every section. The results of these two ILS were compared to the previously-discussed ILSs. Both time-efficient methods showed similar RB reaction forces behaviour illustrated in Figure 4.34. However, the force peak values differ significantly from those resulting from the original ILS in the lumped-mass pipeline in OrcaFlex. This is mainly due to a discrepancy in the mass definitions of the sections in the new ILSs. As discussed, only the median and mean bending stiffness per section was taken. The difference in mass per unit length of each section affected the RB reaction forces, which subsequently affected the axial strains illustrated in Figure 4.35. As seen for the example of RB V15, there is a big difference in RB reaction force which eventually leads to an axial strain peak difference at this location. Also seen in Figure 4.35, not taking into account the mass per unit length for these time-efficient ILS can influence the axial strain results of up to 25 [m] of pipeline from the ILS.

6

Conclusion

This chapter will synthesise preceding chapters in concise statements in Section 6.1, additionally providing definitive answers to the research question and sub-questions in Section 6.2.

6.1. Key conclusions

- The addition of three support rollers to the full-overbend-model setup in Abaqus allowed for corresponding contact behaviour, avoiding boundary condition effects at the start of the pipeline.
- The modification of the rollerboxes from one rigid body to two rigid bodies allows for identical recreation of the rollerboxes configurations of OrcaFlex in Abaqus.
- The Python scripts written to recreate the rollerboxes configurations of OrcaFlex in Abaqus provide the ability to automate this process whilst still providing identical results.
- For the stinger configuration for this research project, rollerbox S1 is governing.
- Despite identical replication of the rollerboxes configurations in a non-linear finite element model by means of using the resulting declination angles from a multi-body model, the behaviour results discussed are not identical between both models.
- The declination angle and tensions resulting from the lumped-mass pipeline closely approximate those resulting from the non-linear finite element pipeline, for the comparison case of a homogeneous pipeline.
- The bending moments in the lumped-mass pipeline are slightly higher, induced by the pipeline slightly bending more.
- The rollerbox reaction forces resulting from the lumped-mass pipeline in OrcaFlex and the non-linear finite element pipeline in Abaqus, for the homogeneous pipeline comparison case, follow the same behaviour pattern. Yet there are slight value discrepancies due to the lack of even distribution for the non-linear finite element pipelines in Abaqus. It hereby becomes evident that despite the approximation of RB behaviour, replicating the RB Configuration from OrcaFlex does not deliver a correct load of pipeline distribution in Abaqus.
- From the RB behaviour pattern resulting from the RB reaction loads, it becomes evident that for most RB the second Hinge has more contact than the first Hinge. This implies that the RBs upholding the non-linear finite element pipeline in Abaqus should have a greater declination angle than those upholding the lumped-mass pipeline in OrcaFlex.
- The rollerbox reaction forces from the non-linear finite element models in Abaqus imply that these pipeline bend slightly less, clarifying the slightly higher bending moments in the lumped-mass pipeline.
- The rollerbox reaction forces clearly show that for realistic force magnitude, RB reaction forces should be extracted from OrcaFlex due to even distribution of the pipeline. Until now, the non-linear finite element models in Abaqus do not take into account rollerbox rotation during analyses

due to a trade off in convergence time. A method to take these rotation is later recommended in Chapter 7.

- Axial strains are typically used during pipeline installation as these account for both tensile and bending strains. Axial strains are the leading type of strains during pipeline installation analyses, due to the high tensions and bending moments taking place whilst installing a pipeline. These are thus extracted in similar way and used for further strain comparisons.
- The axial strains resulting from the lumped-mass pipeline are slightly overestimated when compared to that of a line-elements pipeline in Abaqus. Resulting from the difference in RB behaviour pattern, a higher bending moment is induced in the lumped-mass pipeline resulting in a higher axial strain, comprising of both tensile and bending strain. However, this overestimation ensures close representation of more detailed axial strains resulting from a solid-elements pipeline in Abaqus.
- The declination angle, tensions and bending moments resulting in the lumped-mass pipeline are assumed to be a close approximation of those resulting in the non-linear finite element pipeline for the both ILS comparison cases. The slight overestimations are deemed as conservative measures during ILS behaviour analyses and ILS design.
- The rollerbox reaction forces, as for the homogeneous pipeline case, show the same behaviour pattern and also a magnitude difference between the non-linear finite element and lumped-mass pipeline.
- The resulting axial strains from the lumped-mass pipeline ILS approximates those resulting from the 3D non-liner finite element pipeline in the overbend section. However, the axial strains of pipe sections adjacent to the ILS are not closely approximated by OrcaFlex. This is due to a modelling difference for the ILS, where the lumped-mass ILS contained two extra ILS components not modelled in the 3D non-linear finite element ILS.
- After ensuring consistent length between both ILS in the lumped-mass pipeline and that in the non-linear finite element pipeline, sudden strain peak overestimations disappeared. The resulting axial strain graph from the lumped-mass pipeline closely approximates that of the non-linear finite element pipeline. ILS consistency between both models is thus highly important.
- The RB reaction force differences have significant effects on the pipeline, which can be seen in the graphs of the resulting tensions and bending moments at the ILS interfaces.
- Overall, the lumped-mass pipeline results in OrcaFlex slightly overestimate those of the non-linear finite element pipeline in Abaqus, leading to a conservative ILS behaviour analysis, thus a conservative ILS design.
- For the new time-efficient ILSs, the corresponding rollerbox reaction forces in OrcaFlex follow the same contact behaviour pattern as previous ILSs. These forces are mostly approximated, however due to only the bending stiffness being taken into account and not the mass per unit length, the peak reaction force magnitude differs. It is therefore crucial to also extract the mass per unit length when extracting the bending stiffness of each section.
- For the new ILSs, the resulting axial strains closely approximate the axial strains previous ILSs. As a result of the mass per unit length not being taken into account, the difference in RB reaction forces also affect the resulting axial strains, for up to 25 $[m]$ of pipeline adjacent to the ILS.
- Aforementioned results conclude that there is barely any difference between using the mean or median bending stiffness for the material definition in OrcaFlex
- Following the aforementioned results it can be concluded that, despite not taking the mass per unit length into account, the fidelity resulting from both time-efficient ILS creation methods provide close approximations of the original lumped-mass pipeline ILS, negating the expected trade off between modelling simplicity and result approximation. Implementing the mass per unit length during the creation method will increase the fidelity of the resulting, approximating the previous ILS results even more.

6.2. Research questions

6.2.1. Sub-questions and answers:

How are pipeline installations currently analysed?

Pipeline installations are currently analysed by using both a multi-body model and a non-linear finite element model of the installation setup. The multi-body model allows for quick analyses by assessing the overall pipeline behaviour, whilst the non-linear finite element allows for more localized analyses and detailed results of its behaviour.

At first glance, the multi-body models and the non-linear finite element models seem to have quite some differences. Both models have the pipeline as main component. Subsequently, identical RB configurations can be achieved. The relevant differences found between the multi-body models and the non-linear finite elements models were: A pipeline-rollerbox contact differences due to boundary condition effects following positioning of the first node in Abaqus; and a discrepancy in the shape of the pipeline throughout the overbend section due to the previous rollerbox models. The boundary condition effects were mitigated by adding support rollers, allowing for horizontal settlement of pipe sections at the start of the pipeline. The shape of the pipeline, and thus its declination angles were replicated by means of modifying the rollerboxes in Abaqus to match those in OrcaFlex. The modification of the start of the pipeline now provides realistic contact behaviour for the first few rollerboxes, avoiding irregular contact behaviour previously seen. The modification of the rollerboxes now allow for roller frame rotation to be accounted for instead of just the main frame rotation, allowing for a better distribution of the load of the pipeline

What are the differences between using lumped-mass and 3D non-linear finite-elements representations for the modelling of homogeneous pipelines?

Outputs such as pipeline declination angles and loads are closely approximated between both representations. The main difference lies in the resulting axial strains, where the lumped-mass pipeline slightly overestimates these due to slightly higher bending moments throughout the pipeline, induced by a bigger bend of the pipeline.

What are the critical positions of an inline structure in the overbend section?

The critical positions of the inline structure in the overbend section is where the rollerbox reaction force of RB S1 is the highest, and where the inline structure bends the most. The highest rollerbox reaction force at RB S1 occurs when the middle of the inline structure lies on this rollerbox. The inline structure bends the most once it is further down the overbend section, whilst simultaneously being upheld by RB S1, RB S2, and RB S3. These two positions were successfully replicated in both software to answer the following sub-question.

What are the differences between using lumped-mass and 3D non-linear finite-elements representations for the modelling of inline structures?

The rollerbox reaction forces for the inline structure comparison between the lumped-mass pipeline and the non-linear finite element pipeline follow the same behaviour pattern, however the peak values differ due to the inherent pipeline-load-distribution discrepancy. It is recommended to only assess the rollerbox reaction forces from the multi-body models in OrcaFlex, as these properly account for the rollerbox behaviour. The main difference lies in the resulting axial strains. The axial strains in most pipe sections in the overbend section were closely approximated once ILS modelling consistency was ensured.

What is a viable time-efficient method for the modelling of inline structures?

Additional to the original previous project ILS, this research project proposed two time-efficient ways to analyse the structural integrity of an ILS. These method are proposed by simplifying the inline structure to just three pipe sections, and extracting pipe section properties from the 3D non-linear finite element

ILS model. Two ILS variations were created, differing in their bending stiffness. The material of the first inline structure is defined by the median data point of the bending moment-curvature relation resulting from Abaqus. The material of the second inline structure is defined by the mean of the data points of the bending moment-curvature relation resulting from the 3D non-linear finite element ILS in Abaqus. Apart from the material definition, both follow the same modelling procedure. Despite having a significant lower fidelity than the original ILS, these alternative time-efficient models still approximated its behaviour.

As for the original inline structure, the resulting reaction forces for the two new inline structure approximated each other. The lack of proper definition for the mass per unit length resulted in lower peak reaction forces, and is therefore recommended to also include mass per unit length definition during the creation of an ILS. When comparing the axial strains of the two new inline structures, these strains are closely approximated. The modelling of an inline structure using just three sections, defined by the median data point of the bending moment - curvature relation, offers adequate approximation for the level of simplicity it provides. The axial strain results show despite the second material definition being much more intricate, no real advantage was gained. It is recommended to also include a definition of the mass per unit length when defining the bending stiffness for each section to have a viable modelling method.

6.2.2. Main question and answer:

How can the structural integrity of an inline structure be analysed in the overbend section?

The standard practice to analyse the structural integrity of an inline structure in the overbend section is to create a high-fidelity inline structure within the lumped-mass pipeline and analyse this pipeline by running a static analysis on this multi-body model. This high-fidelity ILS requires additional pipe section properties in order to approximate its behaviour. This research project has shown that two alternative time-efficient methods for the modelling of an ILS - an ILS with a reduced fidelity - also provides similar results to those previously done. The ILS within the lumped-mass pipeline can be recreated by means of using material curves extracted from the 3D non-linear finite element ILS model. After doing so, the critical positions for respective stinger configuration has to be determined. Once the ILS is placed in these positions, it can be benchmarked with its non-linear finite-element equivalent model. By means of ensuring this consistency, the correctness of the multi-body model can be ensured. Once ensured, further global and local analyses can be independently analyses using the multi-body model and non-linear finite element model respectively, in order to analyse structural integrity of an ILS in the overbend section.

7

Recommendations

1. The positions used to place the solid elements in the pipeline in Section 3.2 and Section 4.1, and also that of the ILS used in Section 3.4 and Section 4.3, were all determined through multi-body models in OrcaFlex for this research project. This is done due to the fact that the runtime using a multi-body model in OrcaFlex is significantly smaller. Yet, by modelling an ILS by means of approximating its behaviour using a lumped-mass pipeline, the positions deemed critical might not exactly represent the critical positions that would result with an actual/3D non-linear finite element model. The lumped-mass pipeline ILS in OrcaFlex is an approximation, thus these positions are also approximation of the critical positions for this ILS. For this research project, one rollerbox is deemed governing and used for locating the critical positions in the vicinity. This simplification is acknowledged and accepted, mainly due to time constraints. In the case of actual installation analyses, it is recommended to not only assess one governing rollerbox and critical positions in its vicinity, but assess the entire overbend section. Other rollerboxes might by close-second governing, inducing high strain peaks, and should also be assessed during the location of critical positions.
2. The “Spring/Damper” feature in Abaqus is a feature that has been used in previous full overbend models. This feature is specifically applied to a rollerbox - instead of fully constraining it - in order to allow one specific rollerbox to better adapt to the pipeline it is supporting. This specific rollerbox is the rollerbox where the highest peak strain occurs, thus it is added after the first completion of the static analysis. Allowing this rollerbox to rotate and adapt during analysis - by applying a spring in its axis of rotation, with an arbitrary stiffness - will lower this peak strain. Adding this degree of freedom ensures that the non-linear finite element model is one step closer to actuality due to the fact that the actual rollerboxes do rotate. The addition of a spring is barely done due to the fact that it significantly increases the convergence time of the non-linear finite element model, and is therefore only done for one rollerbox. For this research project, no spring has been (initially) added in order to first assess whether directly replicating the positions and declination angles of the rollerboxes from OrcaFlex into Abaqus would suffice. From the results it became clear that the rollerboxes do not distribute the pipeline weight evenly, thus it is recommended to further analyse the full overbend models with the additional spring feature, and assess its convergence time trade-off.
3. The ILS used for this research project has a clear modelling inconsistency between that in the multi-body model and that in the non-linear finite element model. This discrepancy is not only due to the method of modelling, lumped-mass vs. non-linear finite elements, but also due to a difference in the length of the ILS. It is suggested that a consistency check is done between both models, prior to running further analyses.
4. To put the two time-efficient ILS creation methods for the lumped-mass pipeline to the test, the 3D non-linear finite element ILS in Abaqus has been divided into three sections in 3.5, which has been deemed as the minimal amount of sections into which the ILS could be divided. The axial strain results approximated those of the original lumped-mass pipeline ILS, thus the ILS fidelity resulting

from the two new methods was deemed as sufficient in the case of time constraints. However, it is recommended to slightly increase the amount of sections created - divide the 3D non-linear finite element ILS into more sections - in order to increase its fidelity in the lumped-mass pipeline. As these new methods only require the definition of the bending stiffness, enhancing the fidelity of the ILS and corresponding results will not significantly increase the modelling complexity nor the resulting workload.

5. The two time-efficient ILS creation methods for the lumped-mass pipeline only took into account the median and mean bending stiffness. This resulted in inaccurate RB reaction forces due to the mass per unit length not being accounted for. It is recommended to also take the median and/or mean of the Mass per Unit Length from corresponding parts and sections, respectively.

References

[1] Allseas, *Audacia - allseas*, en-US, Apr. 2024. [Online]. Available: https://allseas.com/equipment/audacia/?_gl=1*qu7z7t*_up*MQ..*_ga*MTc3MTQyNTAwNy4xNzQxMjY1NTQ2*_ga_MZ8Y7ZBM0N*MTc0MTI2NTU0NS4xLjEuMTc0MTI2NTU0NS4wLjAuMA..

[2] S. Gong, P. Xu, S. Bao, W. Zhong, N. He, and H. Yan, "Numerical modelling on dynamic behaviour of deepwater s-lay pipeline," *Ocean Engineering*, vol. 88, pp. 393–408, 2014, ISSN: 0029-8018. DOI: https://doi.org/10.1016/j.oceaneng.2014.07.016. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0029801814002790.

[3] S. Gong and P. Xu, "The influence of sea state on dynamic behaviour of offshore pipelines for deepwater s-lay," *Ocean Engineering*, vol. 111, pp. 398–413, 2016, ISSN: 0029-8018. DOI: https://doi.org/10.1016/j.oceaneng.2015.11.013. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0029801815006265.

[4] W. Xu, Z. Song, C. Meng, and H. Yi, "Experimental study on fully coupled s-lay pipeline installation in shallow water," *Ocean Engineering*, vol. 324, p. 120 724, 2025. DOI: 10.1016/j.oceaneng.2025.120724. [Online]. Available: https://www.sciencedirect.com/science/article/abs/pii/S0029801825004391.

[5] Allseas, *Allseas website*. [Online]. Available: https://www.allseas.com/en.

[6] O. States, *Subsea pipeline products - oil states industries*. [Online]. Available: https://oilstates.com/offshore/subsea-pipeline-products/.

[7] *AutoDesk Inventor Software*. [Online]. Available: https://www.autodesk.com/eu/products/inventor/overview.

[8] O. Ltd, *OrcaFlex - dynamic analysis software for offshore marine systems*, Jun. 2019. [Online]. Available: https://www.orcina.com/orcaflex/.

[9] Orcina, *Technical Specification Version 11.5*. [Online]. Available: https://www.orcina.com/wp-content/uploads/OrcaFlex-v11.5-Tech-Spec.pdf.

[10] *Abaqus - Dassault Systèmes*, 2024. [Online]. Available: https://www.3ds.com/products/simulia/abaqus.

[11] P. Xu, Y. Hu, X. Wang, J. Zheng, and B. Han, "Refined vfife modeling on dynamic behaviours of deepwater s-lay pipeline in overbend section," *Ocean Engineering*, vol. 302, p. 117 698, 2024, ISSN: 0029-8018. DOI: https://doi.org/10.1016/j.oceaneng.2024.117698. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0029801824010357.

[12] T. Renes, "Radial stinger flexibility – the influence on roller box loads and strains in the pipeline during deep water pipelay -," 2010.

[13] T. de Haan, *Rollerbox stiffness offpipe's input parameters*, Confidential report, Allseas in collaboration with The Hague University of Applied Sciences, 2012.

[14] J. Haarman, "Roller box loads on the solitaire stinger," p. 124, 2011.

[15] H. Kramer, "Rolling resistance in rubber coated rollers," 2009. [Online]. Available: https://repository.tudelft.nl/record/uuid:fd597dac-0776-42b5-8da7-b96d492221ee.

[16] Q. Bai and Y. Bai, "8 - finite element analysis of in situ behavior," in *Subsea Pipeline Design, Analysis, and Installation*, Q. Bai and Y. Bai, Eds., Boston: Gulf Professional Publishing, 2014, pp. 171–185, ISBN: 978-0-12-386888-6. DOI: https://doi.org/10.1016/B978-0-12-386888-6.00008-0. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780123868886000080.

[17] G. LUN, Y. LIU, P. YI, C. LI, X. MA, and Y. QU, “Impacts of ocean current and seabed friction on the picking-up and laying-down processes of oil and gas pipelines,” *Petroleum Exploration and Development*, vol. 40, no. 1, pp. 119–125, 2013, ISSN: 1876-3804. DOI: https://doi.org/10.1016/S1876-3804(13)60013-0. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1876380413600130.

[18] D. S. [Simulia], *Abaqus Analysis User’s Guide, vol3*, en, 2016th ed. 2016, vol. 3. [Online]. Available: http://130.149.89.49:2080/v2016/pdf_books/ANALYSIS_3.pdf.

[19] R. Ramasamy and T. Tuan Ya, “Nonlinear finite element analysis of collapse and post-collapse behaviour in dented submarine pipelines,” *Applied Ocean Research*, vol. 46, pp. 116–123, 2014, ISSN: 0141-1187. DOI: https://doi.org/10.1016/j.apor.2014.02.007. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0141118714000212.

[20] en. [Online]. Available: https://help.3ds.com/2026/English/DSSIMULIA_Established/SIMACAETHERefMap/simathe-c-beams.htm?contextscope=all&id=28f487afe70f43989fd66f10e1e95753.

[21] [Online]. Available: https://orcina.com/webhelp/OrcaFlex/.

[22] [Online]. Available: http://130.149.89.49:2080/v2016/books/usb/default.htm?startat=pt06ch27s01abo25.html#usb-elm-egeneral.

[23] [Online]. Available: https://classes.engineering.wustl.edu/2009/spring/mase5513/abaqus/docs/v6.6/books/stm/default.htm?startat=ch01s04ath05.html.

A

Research Paper

Inline Structure fidelity in OrcaFlex

Ivan Lucas Gonzalo[a]

[a]*Allseas & Delft University of Technology,*

Abstract

This research investigates the pipeline installation analyses process using both multi-body models and non-linear finite element models, specifically for the installation of a pipeline containing an inline structure (ILS). The current method involves a tedious discretization workflow, where the ILS is divided into different sections, and each section with respective properties are recreated in order to approximate its actual behaviour in a multi-body model. The installation of previous homogeneous pipelines and that containing an ILS is analysed and understood. Additionally, an alternative method for the creation of an ILS is investigated, namely its trade-off between simplicity and result fidelity is assessed by benchmarking its results with those of previous models. Results indicate that simplifying the ILS in just three sections, and taking the median of the bending moment - curvature relation provides slightly overestimated yet adequate results when compared to previous models.

Keywords: Inline Structure, OrcaFlex, Pipeline installation analysis, Offshore

1. Introduction

Offshore pipelines are considered the most efficient and economically-viable means to transport these hydrocarbons, whether crude oil or natural gas, from subsea wellheads to offshore or coastal processing facilities Gong et al. (2014)Gong and Xu (2016). In order to create such infrastructures, various techniques have been invented and developed. In general, the so-called S-lay and J-lay pipe installation techniques are the most recurring, especially in deep-water pipeline installations Gong and Xu (2016). The names refer to the shape to which the pipeline bends once it is lowered into the sea, as seen in Figure 1 for S-lay.

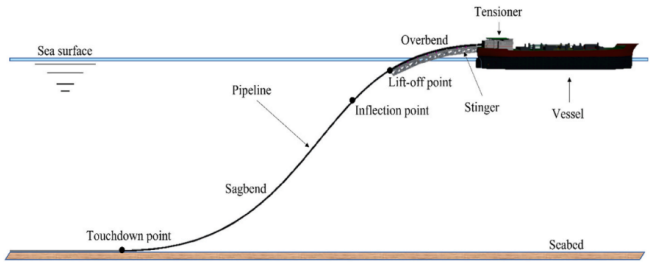


Figure 1: Schematic diagram of S-Lay installation process Xu et al. (2025)

Besides connecting regular pipe sections to each other in order to create pipelines, some structures are also connected within these pipe sections in the pipeline, in order to further develop these infrastructures. As these structures lie within the pipeline - therefore called inline structures - they also go through the same installation process as pipe sections.

By continuously connecting pipe sections to each other, the pipeline is elongated until the required length is reached. As the pipeline gets longer, the pipe sections get shifted from the vessel down towards the stinger, Figure 1. Once on the stinger, the pipeline is upheld by its rollerboxes (RBs) - a set of usually 4 V-shaped roller pairs, confined in a metallic frame. These stinger and its RBs ensure that the pipeline does not abruptly drop into the sea, following its S-shape and thus maintaining its integrity. Once the pipeline leaves the stinger, it will sag down until it reaches and lies on the seabed.

Before these pipeline installations can be executed, rigorous analyses are performed in order to ensure feasibility. These analyses vary between global and local analyses using multi-body and non-linear finite element models, respectively. In the case of ILS analyses, the ILS has to be modelled accordingly. Non-linear finite element models can entail extensive 3D modelling, thus the ILS is modelled with high fidelity such that its results are considered realistic. However, the ILS in multi-body models can only be modelled by means of adjusting certain line-element properties at specific locations. This 1D representation of the ILS thus tries to approximate the behaviour of the actual ILS, assumed to be the ILS in Abaqus for this research project. This method is still standard practice due to its ability to capture global behaviour within seconds. This modelling method entails discretizing the ILS by means of dividing it into different section. Each section requires tedious data extraction to adjust its respective line-element properties. Therefore, this research project not only investigates overall pipeline installation analyses, but also investigates the trade-off between simplicity and result fidelity when modelling an ILS by proposing and benchmarking an alternative modelling method.

2. Method

Firstly, a comprehensive characterization of differences and similarities between multi-body models in OrcaFlex and non-linear finite elements models in Abaqus has to be obtained.

These are obtained following the subsequent case comparisons:

1. A modelling-setup comparison between (previously used) Abaqus and OrcaFlex models, Section 2.1.
2. A homogeneous-pipeline case comparison between the multi-body model and the non-linear finite element model, Section 2.1.
3. An inline-structure case comparison between the multi-body model and the non-linear finite element model, Section 2.1.

The first case comparison will deliver insights into the differences and similarities between the setups in both models. This allows for better understanding and assessment on how the different components are modelled, what differences in results could arise, and whether these differences are even relevant to this research.

The second case comparison subsequently builds on these results. A homogeneous pipeline in modelled and analysed due to the fact that both software provide enough modelling details in order to accurately analyse a homogeneous pipeline. By then comparing the results of an identical setup, resulting differences could be boiled down to inherent systematic differences.

Having these known difference, the last case comparison can be performed. To further narrow the scope, only one ILS is used for further comparisons. This ILS is of a past project, and the then-modelled ILS are used. The resulting comparisons will give sufficient insights in order to then propose an alternative modelling method for an ILS. Subsequently, this modelling method is put to the test by benchmarking its results against that of the previously-used models.

2.1. Modelling Setup case comparison

For the case comparison of the modelling setups, both the multi-body model in OrcaFlex and the non-linear finite element model in Abaqus are compared. A general comparison between modelling setup is done, followed by an in-depth comparison of the pipelines and of an ILS. The comparisons are boiled down and only the most relevant are discussed. The most relevant differences are the following two:

- Support Rollers
- Rollerboxes

Their differences and possible solution are condensed in the following sections.

2.1.1. Support Rollers

The setup in OrcaFlex contains support rollers, which are extra rollers placed before the RBs, which supporting the pipeline horizontally throughout the vessel. The setup in Abaqus does not model these, and the first node of the pipeline is placed right above the first RB. By not modelling these, boundary condition effects might take place in the case that the coordinates of the first node are not exact. By not correctly placing this node, the rest of the nodes might be too close to these RBs or not even come into contact with them, subsequently delivering inaccurate contact results. To solve this problem, support rollers are added to the Abaqus model, preceding the first RB. These will uphold the pipeline horizontally before coming into contact with the first RB. In addition, this will allow for slight discrepancies in the coordinates of the first node, as the pipeline will correct its contact behaviour on support roller, followed by accurate contact behaviour on the rollerboxes. The amount of support rollers added in Abaqus is arbitrarily chosen as three, Figure 2, and its initial test delivered improved contact results. By already improving the contact results, no additional support rollers are added as each roller support would increase the convergence time in Abaqus.

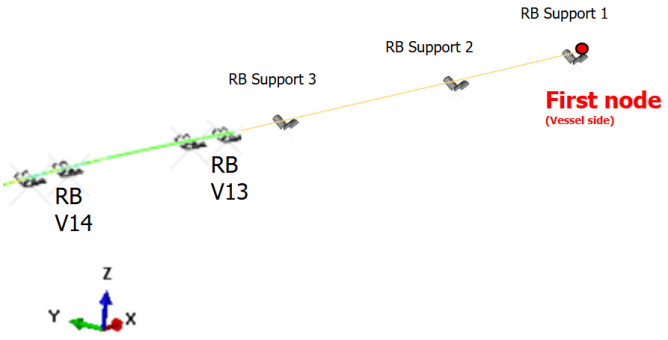


Figure 2: First part of the Abaqus full overbend model setup after adding support rollers

2.1.2. Rollerboxes

The RBs in OrcaFlex closely approximate their actual behaviour due to the fact that OrcaFlex allows for corresponding rotations during analyses. Each individual hinge has its own degree of freedom, where additional displacements are consequent of rotation at the main hinge. Figure 2 further illustrates this close approximation between the actual RB and its OrcaFlex representation. However, Abaqus does not allow for rotation during analyses and the RB behaviour is therefore simplified. Until now, the RBs have each been modelled as one rigid body, fixed at a certain angle at a certain location. This rigid body does not allow for the individual rotations at the hinges and respective displacements, and only accounts for rotation at the main hinge. In order to still account for the individual rotations of the hinges with their respective displacements, the RBs are modified and now modelled using two rigid bodies. Each rigid body contains its own reference point (RP), where corresponding angles and coordinates are set. This now allows closer approximation of pipeline behaviour between both models.

2.2. Homogeneous Pipeline case comparison

After the addition of support rollers and the modification of the RBs, the case comparison can take place. As mentioned before, the non-linear finite element model and its materials are deemed as reality and are thus used to modify and benchmark the multi-body model. In order to ensure consistency, the pipeline properties are compared and matches in order to assure both pipelines are identical. Pipe section properties such

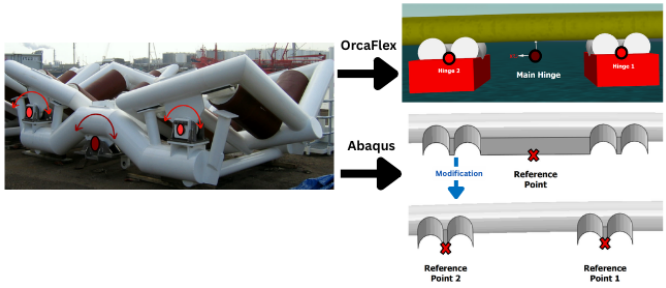


Figure 3: Illustration comparison of an actual rollerbox with respective OrcaFlex and Abaqus (original and modified) models

as outer diameter (OD) are directly filled in both software. Yet, the pipeline of the multi-body model is represent by means of a lumped-mass model: the pipeline comprises of masses (nodes) interconnected by springs (segments). Most properties such as diameters are lumped onto the nodes, whereas axial and torsional stiffness are lumped onto the segments. By doing so, the convergence times of these models can be kept significantly low. The pipeline specification thus differs. The main driver property here is the bending stiffness, which requires data points for the bending stiffness of each pipe section, defined by the relation between curvature and resulting bending moments of the material. To ensure identical definition of the material, both bending moment $NFORCESO5$ and curvature $SK1$ data is extracted from a completed static analysis of the pipeline in Abaqus, and used as data point input in OrcaFlex. Once material consistency is obtained, a static analysis can be performed of the homogeneous pipeline models.

The following results are extracted and compared:

1. Rollerboxes configurations
2. Pipeline declination angles.
3. Loads at pipeline interfaces
4. Rollerbox reaction forces
5. Strains

Rollerboxes configurations comparison illustrates consistency between both model setups. *Pipeline declination angles* comparison illustrates resulting pipeline behaviour, assuring the aforementioned setups are identical. *Top and bottom tension* comparison illustrate whether the applied force (bottom tension) is identical and applied correctly, and how the resulting top tensions between models relate. *Tensions and bending moment at the element interfaces* comparison refers to the loads where the pipeline transitions from line to solid elements and vice versa, and is checked for boundary condition effects. *Rollerbox reaction forces* comparison illustrate the rollerbox-pipeline contact behaviour differences. *Strains* comparison illustrates the elongation differences between both models. This comparison is the most important, as the strains are looked upon for integrity assessments. It is thus relevant to know whether these are approximated between the models. In addition, as there are a various amount of strains measurement outputs, it is key to find the correct output in order to compare these measurements. The non-linear finite elements are modelled in three different ways: Only using $PIPE31H$ line elements; Combining $PIPE31H$ line elements with a section of $C3D8R$ solid elements; Similar combination of line and solid elements, where solid-element pipe section is longer. The line element pipeline will allow a 1:1 comparison with the line elements of the lumped-mass pipeline. Including solid elements now allows for more-detailed results. Further elongating this allows for the comparison in results, where the effects of the transitions of element type are studied.

The subsequent case comparisons will also maintain the aforementioned list of results to be extracted, yet not all results might be deemed as relevant to show. As mentioned before, the most relevant output to be compared are the strains.

2.3. Inline Structure case comparison

After getting insights on possible systematic differences, the ILSs themselves are compared. As mentioned before, the ILS models used during the respective projects are used and implemented within the previously-used models for the homogeneous pipeline case comparison. For the ILSs, the strains within the ILS themselves are not extracted, as the strains from the ILS in the multi-body model will naturally lead to different strains. For detailed ILS strains, structural engineers perform localised analyses, which is namely outside of the scope for this research project. The strains that are relevant for this research project are those of adjacent pipe sections, as this directly relates to the integrity of the pipeline which is crucial to be maintained.

The ILs will be placed and compared in two positions: 1) Where the RB reaction force is the highest 2) Where the ILS bends the most. Where the RB reaction force is the highest relates to a critical position as this has to be studied according to Allseas' analyses procedure, in order to assure rollerbox structure integrity limits are maintained. Where the ILS bends the most relates to a critical as this location is where the ILS bends the most, causing the highest resulting ILS strains. The exact location along the overbend section of the pipeline in which the ILS will be, is determined by displacing the ILS in OrcaFlex over the governing rollerbox in increments of $0,1$ $[m]$. At every position, a static analysis will be performed and results are extracted into one excel file. This process is all automated by means of a Python script.

2.4. Alternative Modelling method

Lastly, an alternative modelling method is proposed in order to create ILSs in OrcaFlex. This new method will make use of the non-linear finite element ILS, as its behaviour and analyses results are deemed as reality. The intention of this is to use realistic results in order to closely approximate the behaviour of the ILS within the lumped-mass pipeline. Using the resulting new method, the ILS is recreated within the lumped-mass pipeline in OrcaFlex and its static analysis results are compared to the previous ILS models. Following these results, the original ILS-creation work flow is modified by implementing this new method.

3. Results

3.1. Homogeneous Pipeline case results

3.1.1. Material Definition

The resulting bending moment and curvatures of the pipeline in Abaqus result in a noisy curve which can not directly be used in OrcaFlex. This curve is refined from negative and recurring data points in order to achieve a filtered version of the curve. Figure 4 illustrates the old material curve in OrcaFlex, the resulting Abaqus curve and its filtered version which is subsequently used as input for the lumped-mass pipeline in OrcaFlex. Following completed static analyses for homogeneous

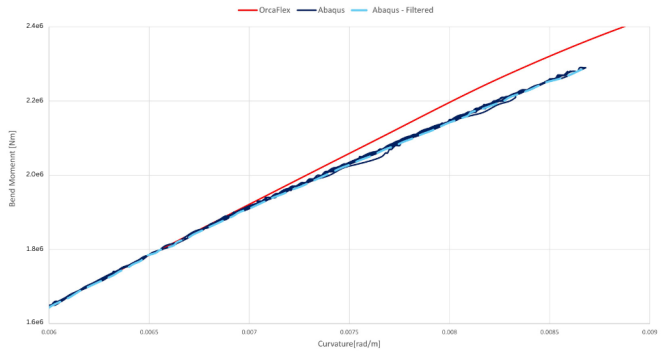


Figure 4: Bending moment-curvature relation curves

lumped-mass and non-linear finite element pipeline, the following conclusions are taken: Both rollerboxes configurations and declination angles of the pipeline are correctly replicated, implying identical setups. The resulting bottom tensions are correctly applied, where also top tensions are identical. The loads at the element-transition interfaces are similar, where those in the lumped-mass pipeline are slightly overestimated. Due to the lack of degrees of freedom in Abaqus, the load of the pipeline is not evenly distributed. The resulting RB declination angles illustrate that those of OrcaFlex are not identical to those in Abaqus. Those in Abaqus should be slightly increased in order to obtain an even distributed of the load of the pipeline. For the strains, the axial strains are compared. The axial strain measures both the direct tensile strains and the bending strain. During pipeline installation, these are the leading strains in the overbend section due to high tension applied and high bending moments due to the pipeline bending over the stinger. Additionally, other strains such as hoop strain are significantly low due to the overbend section being above or relatively close to the waterline.

The direct output for axial strain in OrcaFlex is *Worst ZZ strain*. The *worst* refers to the highest value of tensile strain±bending strain between the outer-most and bottom-most outer diameter for each cross section. This is not a direct output in Abaqus, yet the corresponding curve is still obtained by extracting both outer-most and bottom-most axial strains $LE11$, and taking only the highest/worst value for each cross section. These strains are illustrated in Figure 5.

The lumped-mass pipeline strains are consistently overestimated when compared to those in the non-linear finite element

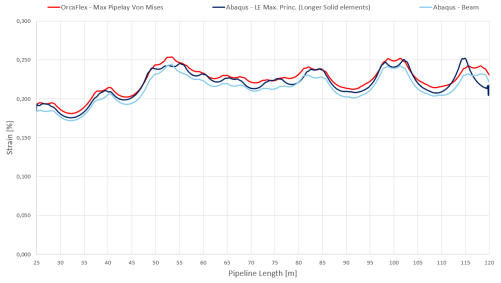


Figure 5: Axial strain comparison

pipeline. However, by overestimating these strains, its curve closely approximates the strains curve resulting from the more-detailed solid-elements pipeline.

3.2. Inline Structure case results

After displacing the ILS over the most governing rollerbox - deemed as RB S1 for the stinger configuration of this research project - the exact locations of both critical positions are obtained, and the result comparisons are discussed in the subsequent sections. ILS model differences and inconsistencies, illustrated in Figure 6, were accounted for by removing mismatching results.

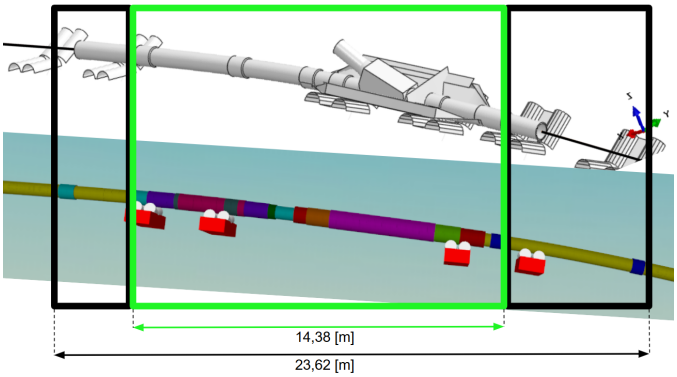


Figure 6: Inline Structure modelling differences

3.2.1. Max RB reaction force case comparison

For the Max RB reaction force case comparison, the following results are obtained.

The rollerboxes configuration are identically replicated between OrcaFlex and Abaqus. However, the declination angles of the pipeline are closely approximated, up until reaching the inflection point. As most of the overbend is closely approximated, the differences are deemed as insignificant. Further tensions and bending moments are approximated yet mostly overestimated in the lumped-mass pipeline. The RB reaction loads follow the same pattern, yet as mentioned for the previous case comparison, due to the lack of degrees of freedom the RBs in Abaqus do not properly distribute the load of the pipeline. The axial strains comparison is illustrated in the left graph in Figure 7. These are closely approximated, yet slightly overestimated in a conservative matter.

3.2.2. Max Bending angle difference case comparison

For the Max Bending angle difference case comparison, the following results are obtained.

The rollerboxes configuration are identically replicated between OrcaFlex and Abaqus. However, the declination angles of the pipeline are closely approximated, up until reaching the inflection point. As most of the overbend is closely approximated, the differences are deemed as insignificant. Further tensions and bending moments are approximated yet mostly overestimated in the lumped-mass pipeline. The RB reaction loads follow the same pattern, yet as mentioned for the previous case comparison, due to the lack of degrees of freedom the RBs in Abaqus do not properly distribute the load of the pipeline. The strains comparison is illustrated in the right graph in Figure 7. These are closely approximated, yet slightly overestimated in a conservative matter.

3.3. Simplified Modelling method results

Following the results of setup, homogeneous pipeline, and ILS comparison, the following modelling methods are proposed. Technically, the steps in order to recreate the ILS in OrcaFlex are identical, however the way that the material is defined differs. Both methods divide the ILS into sections. For the purpose of being as simplistic as possible, the ILS is divided into three sections. The first method takes the median data point of the bending moment-curvature relation of each section, whereas the second method uses the mean of the bending moment-curvature relation of each section. Once the materials are defined, a section in OrcaFlex is created and respective material data points are used as input. After doing so for both methods, two ILSs are obtained. After completing a static analysis on both ILS, the results are benchmarked against those of the previous models. The respective RB reaction forces and axial strains are illustrated in Figure 8 and 9 respectively.

4. Discussion

This chapter will provide a set of interpretations of the results of the aforementioned chapter, in order to pave the way for the following conclusions.

4.1. Modelling setup

The modelling setups have been compared, resulting differences and possible solutions have been proposed. Only two of these differences have been deemed relevant for this research, tackled and their solutions have been implemented.

In previous non-linear finite element models, the start of the pipeline used to be placed on top of the first rollerbox RB V13. The placing of the first node of the pipeline would have to be perfect. Having the first node too high (no contact) or too low (excessive contact) would lead to distorted contact behaviour at the first RB, also impacting contact behaviour of following RBs. To further avoid contact behaviour discrepancies, three additional support rollers have been added preceding RB V13, illustrated in Figure 2. The first node of the pipeline is placed further away from the RB V13. Now, the first ± 30 $[m]$ of the pipeline are supported by these three support rollers, providing a leeway for the pipeline to settle horizontally before coming into contact with the rollerboxes. This mimics the setup found in both the actual pipelay vessels and that of the multi-body models in OrcaFlex.

In previous non-linear finite element models, the RBs used to be modelled as one rigid body, where only the declination angle was set at its reference point axis, Figure 3. In other words, only main frame rotation was taken into account. Only taking into account main frame rotation disregards the roller frame rotations, not allowing for even distribution of the load of the pipeline. To take into account roller frame rotation, the modelling of the RBs has been change to two rigid bodies as seen in Figure 3. This is applied for all RBs besides RB S17, due to the fact that RB S17 does not have roller frame rotations to take into account. The individual bodies can now be rotated along their reference point axis, replicating the resulting RB configuration from OrcaFlex into Abaqus. This resulted in identical RB configuration for all cases.

4.2. Homogeneous Pipeline

Homogeneous pipelines have been modelling in both multi-body models as non-linear finite elements models in order to compare the results of respective static analyses. As both models provide the ability to closely model and approximate realistic results of homogeneous pipeline, the aim of this comparison is to highlight inherent differences when modelling an identical pipeline using both models. A lumped-mass pipeline is compared to three non-linear finite element pipelines: one comprising of just line elements, one comprising mainly of line elements with an additional 80 $[m]$ of a solid-element pipe section, and one comprising mainly of line elements with an additional 95 $[m]$ of a solid-element pipe section. The line-element pipeline allows for a one-to-one comparison with the lumped-mass pipeline. The 80 $[m]$ solid-element pipe section provides high-fidelity results at a specific location, and the 95 $[m]$ solid-element pipe section provides extra insights on the boundary condition effects where the pipeline switches from solid to line elements.

The results of the static analysis comparison for homogeneous pipelines show the following: After modifying the models of the RBs, an identical RB configuration is achieved by applying the resulting declination angles from OrcaFlex into Abaqus using Python script. This subsequently leads to similar pipeline shapes. The top tensions result to be the same for all pipelines after applying the same bottom tension, regardless of it containing only line or solid elements. The interface tensions are therefore the same at the solid pipe section interfaces, however there is a slight difference for the bending moments at the solid pipe section interfaces. The bending moments in the lumped-mass pipeline in OrcaFlex seem to be slightly higher. This difference in bending moment at the interfaces is caused by the difference in contact behaviour. The RB reaction forces of Abaqus imply that the load of the non-linear finite element pipeline is not evenly distributed for all RBs. Most second hinges of the RBs seem to have a higher force magnitude, implying that the RBs should be slightly less horizontal to

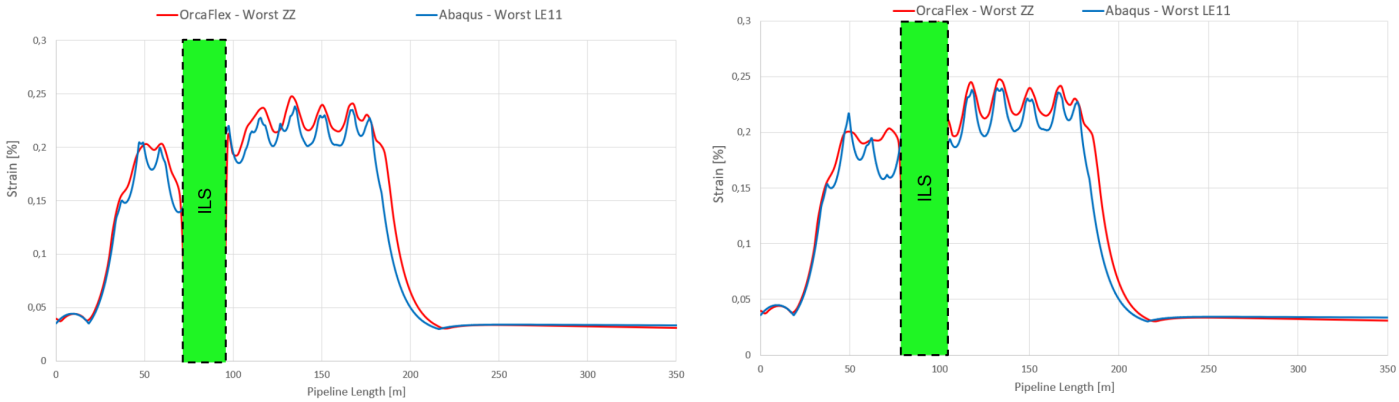


Figure 7: Axial strains comparison for the Max RB Reaction Force case (left) and Max Bending Angle difference case (right)

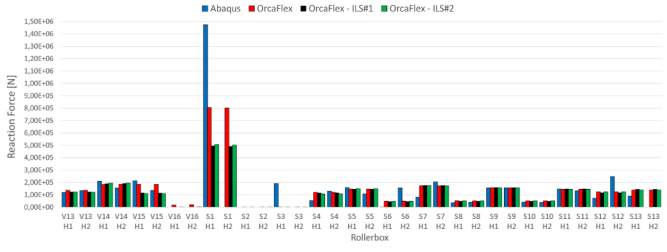


Figure 8: Rollerbox reaction forces comparisons between all four inline structures

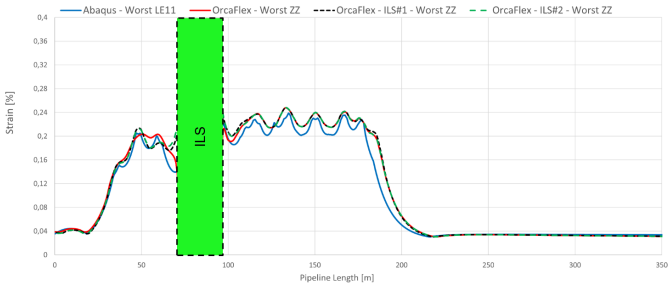


Figure 9: Axial strain comparisons between all four inline structures

evenly distribute the load of the pipeline. By these RBs being more horizontal, the non-linear finite element pipeline is therefore bent slightly less than the lumped-mass pipeline, causing slightly higher bending moments in the lumped-mass pipeline. This also becomes evident when plotting the axial strain in Figure 5, which comprises of both tensile and bending strain. The slight increase in bending moments throughout the lumped-mass pipeline induces higher bending strains, thus higher axial strains. The resulting higher-fidelity strains of the solid element pipe section seen in Figure 5 shows that using a lumped-mass pipeline model provides slightly-overestimated and conservative yet closely-approximated strains compared to those of non-linear finite elements. The additional 95 $[m]$ pipeline illustrate that the solid-line element transition can cause for peak values for the RBs and for the strains.

4.3. Inline Structures

The exact location of the two critical positions of the ILS were first determined by means of displacing the ILS over the governing rollerbox RB S1, using the respective Python scripts. The position with the highest rollerbox reaction force turned out to be whilst RB S1 was upholding the ILS in the middle. The position where the ILS bent the most turned out to be whilst the ILS was being upheld simultaneously by RB S1, RB S2, and RB S3.

As for the homogeneous pipeline case, the RB configurations in OrcaFlex were successfully replicated in Abaqus using the respective Python scripts. The resulting top tensions and interface loads were are overestimated due to the lumped-mass ILS being an approximation of the 3D non-linear finite element ILS. The RB reaction forces imply similar pipeline behaviour, yet uneven load distribution for the non-linear element pipeline. As seen for the homogeneous pipeline case, the RB reaction forces resulting from the multi-body model provide more realistic RB reaction force values due to the inherent ability to take into account the RB rotational behaviour. Initial strain results show sudden strain peaks, which after further examination, are caused by components *A-A* and *T-T* which were not taken into account when creating the 3D non-linear finite element ILS as illustrated in Figure 6. To correctly compare the strains, adapted axial strain graphs are provided in Figure 7 for both cases.

4.4. Time-efficient Methods

Two new viable time-efficient methods were proposed to model an ILS within a lumped-mass pipeline, where the difference between these two lies in the material definition. The material data is extracted from the 3D non-linear finite element ILS in Abaqus itself. This ILS is first divided into three sections where the first method proposed to use the median bending moment - curvature relation of each section, and the second method proposed to use the mean bending moment - curvature relation of each section. The 3D non-linear finite element ILS was divided into a total of three sections to recreated within the lumped-mass pipeline in OrcaFlex, where each section only varied in its bending stiffness - bending moment - curvature relation.

Two ILS were created differencing in just the method the bending stiffness was extracted from the 3D non-linear finite element ILS in Abaqus, where ILS #1 uses the median bending stiffness and ILS #2 uses the mean bending stiffness of every section. The results of these two ILS were compared to the previously-discussed ILSs. Both time-efficient methods showed similar RB reaction forces behaviour illustrated in Figure 8. However, the force peak values differ significantly from those resulting from the original ILS in the lumped-mass pipeline in OrcaFlex. This is mainly due to a discrepancy in the mass definitions of the sections in the new ILSs. As discussed, only the median and mean bending stiffness per section was taken. The difference in mass per unit length of each section affected the RB reaction forces, which subsequently affected the axial strains illustrated in Figure 9. As seen for the example of RB V15, there is a big difference in RB reaction force which eventually leads to an axial strain peak difference at this location. Also seen in Figure 9, not taking into account the mass per unit length for these time-efficient ILS can influence the axial strain results of up to 25 [m] of pipeline from the ILS.

5. Conclusion

This chapter will synthesise preceding discussions and itemize respective conclusions:

- The addition of three support rollers to the full-overbend-model setup in Abaqus allowed for corresponding contact behaviour, avoiding boundary condition effects at the start of the pipeline.
- The modification of the rollerboxes from one rigid body to two rigid bodies allows for identical recreation of the rollerboxes configurations of OrcaFlex in Abaqus.
- The Python scripts written to recreate the rollerboxes configurations of OrcaFlex in Abaqus provide the ability to automate this process whilst still providing identical results.
- For the stinger configuration for this research project, rollerbox S1 is governing.
- Despite identical replication of the rollerboxes configurations in a non-linear finite element model by means of using the resulting declination angles from a multi-body model, the behaviour results discussed are not identical between both models.
- The declination angle and tensions resulting from the lumped-mass pipeline closely approximate those resulting from the non-linear finite element pipeline, for the comparison case of a homogeneous pipeline.
- The bending moments in the lumped-mass pipeline are slightly higher, induced by the pipeline slightly bending more.
- The rollerbox reaction forces resulting from the lumped-mass pipeline in OrcaFlex and the non-linear finite element pipeline in Abaqus, for the homogeneous pipeline comparison case, follow the same behaviour pattern. Yet there are slight value discrepancies due to the lack of even distribution for the non-linear finite element pipelines in Abaqus. It hereby becomes evident that despite the approximation of RB behaviour, replicating the RB Configuration from OrcaFlex does not deliver a correct load of pipeline distribution in Abaqus.
- From the RB behaviour pattern resulting from the RB reaction loads, it becomes evident that for most RB the second Hinge has more contact than the first Hinge. This implies that the RBs upholding the non-linear finite element pipeline in Abaqus should have a greater declination angle than those upholding the lumped-mass pipeline in OrcaFlex.
- The rollerbox reaction forces from the non-linear finite element models in Abaqus imply that these pipeline bend slightly less, clarifying the slightly higher bending moments in the lumped-mass pipeline.
- The rollerbox reaction forces clearly show that for realistic force magnitude, RB reaction forces should be extracted from OrcaFlex due to even distribution of the pipeline. Until now, the non-linear finite element models in Abaqus do not take into account rollerbox rotation during analyses due to a trade off in convergence time. A method to take these rotation is later recommended in Chapter **??**.
- Axial strains are typically used during pipeline installation as these account for both tensile and bending strains. Axial strains are the leading type of strains during pipeline installation analyses, due to the high tensions and bending moments taking place whilst installing a pipeline. These are thus extracted in similar way and used for further strain comparisons.
- The axial strains resulting from the lumped-mass pipeline are slightly overestimated when compared to that of a line-elements pipeline in Abaqus. Resulting from the difference in RB behaviour pattern, a higher bending moment is induced in the lumped-mass pipeline resulting in a higher axial strain, comprising of both tensile and bending strain. However, this overestimation ensures close representation of more detailed axial strains resulting from a solid-elements pipeline in Abaqus.
- The declination angle, tensions and bending moments resulting in the lumped-mass pipeline are assumed to be a close approximation of those resulting in the non-linear finite element pipeline for the both ILS comparison cases. The slight overestimations are deemed as conservative measures during ILS behaviour analyses and ILS design.
- The rollerbox reaction forces, as for the homogeneous pipeline case, show the same behaviour pattern and also

a magnitude difference between the non-linear finite element and lumped-mass pipeline.

- The resulting axial strains from the lumped-mass pipeline ILS approximates those resulting from the 3D non-liner finite element pipeline in the overbend section. However, the axial strains of pipe sections adjacent to the ILS are not closely approximated by OrcaFlex. This is due to a modelling difference for the ILS, where the lumped-mass ILS contained two extra ILS components not modelled in the 3D non-linear finite element ILS.

- After ensuring consistent length between both ILS in the lumped-mass pipeline and that in the non-linear finite element pipeline, sudden strain peak overestimations disappeared. The resulting axial strain graph from the lumped-mass pipeline closely approximates that of the non-linear finite element pipeline. ILS consistency between both models is thus highly important.

- The RB reaction force differences have significant effects on the pipeline, which can be seen in the graphs of the resulting tensions and bending moments at the ILS interfaces.

- Overall, the lumped-mass pipeline results in OrcaFlex slightly overestimate those of the non-linear finite element pipeline in Abaqus, leading to a conservative ILS behaviour analysis, thus a conservative ILS design.

- For the new time-efficient ILSs, the corresponding roller-box reaction forces in OrcaFlex follow the same contact behaviour pattern as previous ILSs. These forces are mostly approximated, however due to only the bending stiffness being taken into account and not the mass per unit length, the peak reaction force magnitude differs. It is therefore crucial to also extract the mass per unit length when extracting the bending stiffness of each section.

- For the new ILSs, the resulting axial strains closely approximate the axial strains previous ILSs. As a result of the mass per unit length not being taken into account, the difference in RB reaction forces also affect the resulting axial strains, for up to 25 $[m]$ of pipeline adjacent to the ILS.

- Aforementioned results conclude that there is barely any difference between using the mean or median bending stiffness for the material definition in OrcaFlex

- Following the aforementioned results it can be concluded that, despite not taking the mass per unit length into account, the fidelity resulting from both time-efficient ILS creation methods provide close approximations of the original lumped-mass pipeline ILS, negating the expected trade off between modelling simplicity and result approximation. Implementing the mass per unit length during the creation method will increase the fidelity of the resulting, approximating the previous ILS results even more.

References

Gong, S., Xu, P., 2016. The influence of sea state on dynamic behaviour of offshore pipelines for deepwater s-lay. Ocean Engineering 111, 398–413. URL: https://www.sciencedirect.com/science/article/pii/S002 doi:https://doi.org/10.1016/j.oceaneng.2015.11.013.

Gong, S., Xu, P., Bao, S., Zhong, W., He, N., Yan, H., 2014. Numerical modelling on dynamic behaviour of deepwater s-lay pipeline. Ocean Engineering 88, 393–408. URL: https://www.sciencedirect.com/science/article/pii/S002 doi:https://doi.org/10.1016/j.oceaneng.2014.07.016.

Xu, W., Song, Z., Meng, C., Yi, H., 2025. Experimental study on fully coupled s-lay pipeline installation in shallow water. Ocean Engineering 324, 120724. URL: https://www.sciencedirect.com/science/article/abs/pii/ doi:10.1016/j.oceaneng.2025.120724.

B

Figures & Tables

B.1. Figures

B.1.1. Chapter 1: Introduction

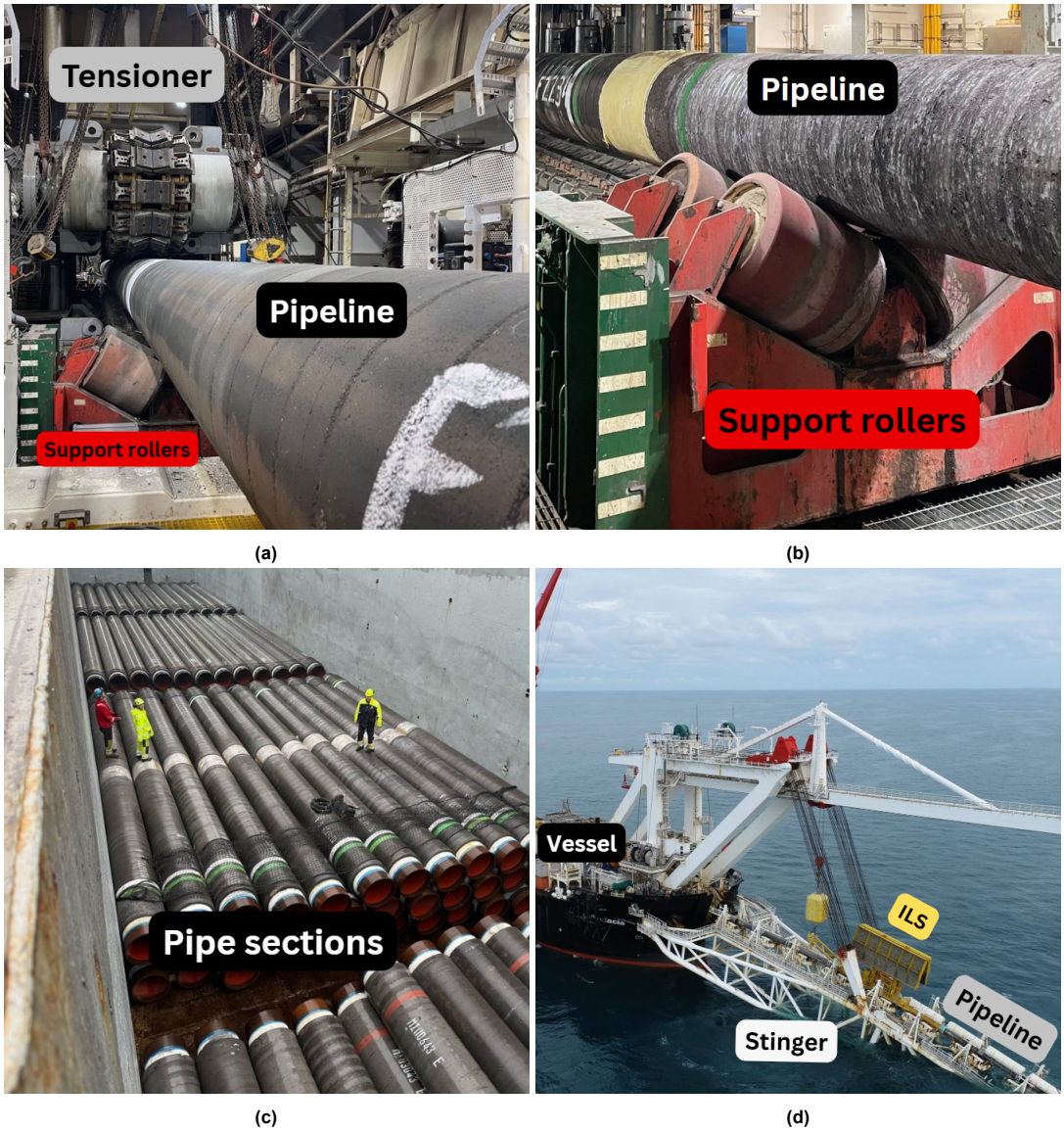
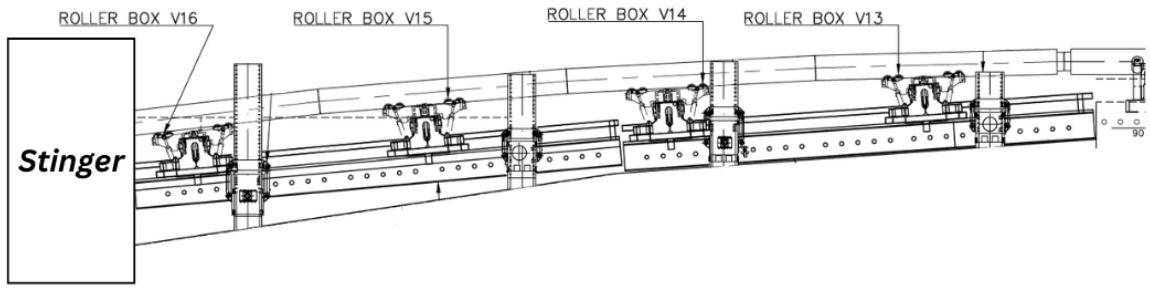
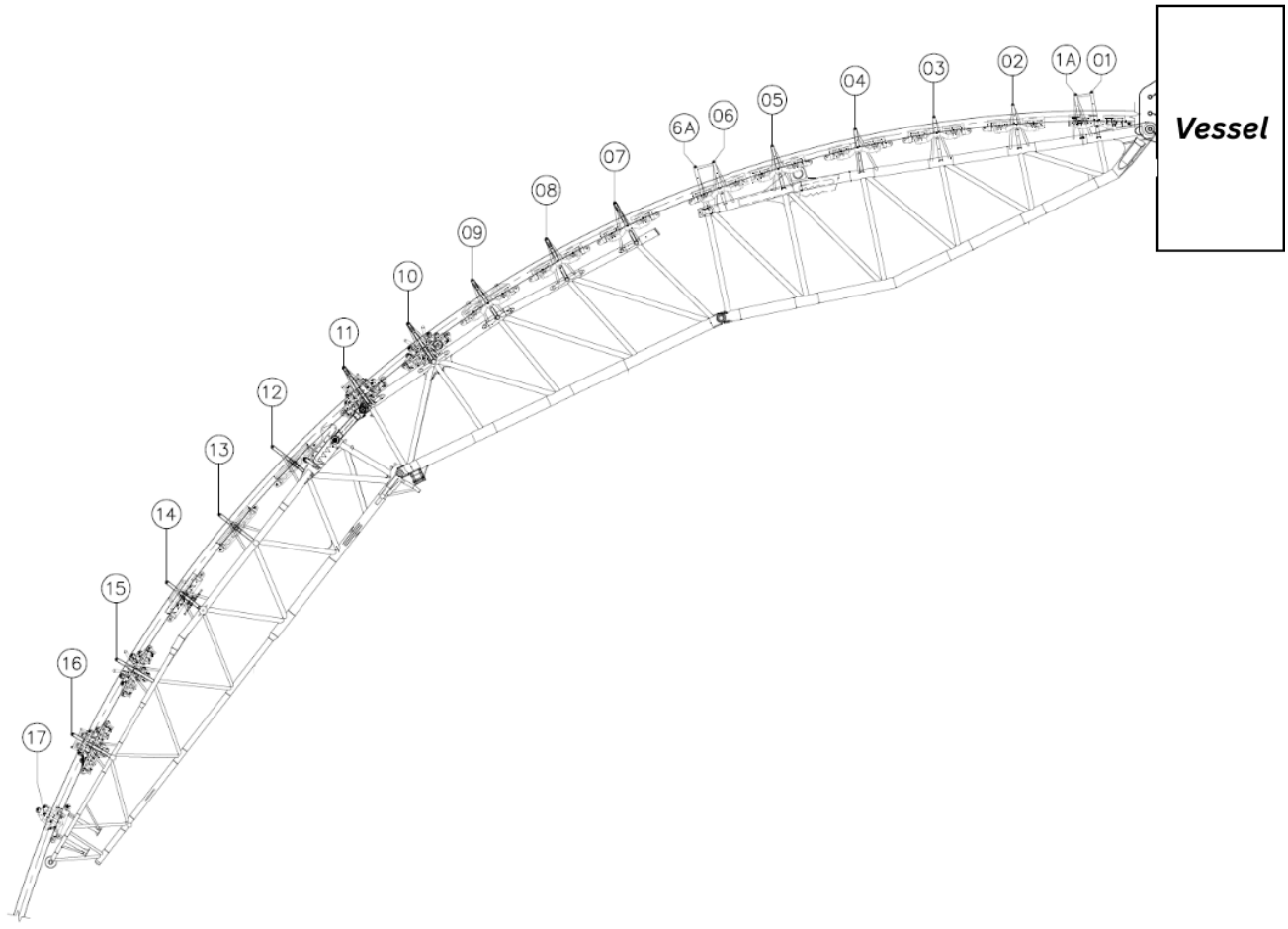


Figure B.1: Images during pipeline installation of: a) Pipeline through a tensioner b) Vessel Support Rollers c) Pipe sections d) ILS on the stinger



(a)



(b)

Figure B.2: Rollerbox setup drawing for Solitaire, with respective tags: a) Vessel rollerboxes b) Stinger rollerboxes

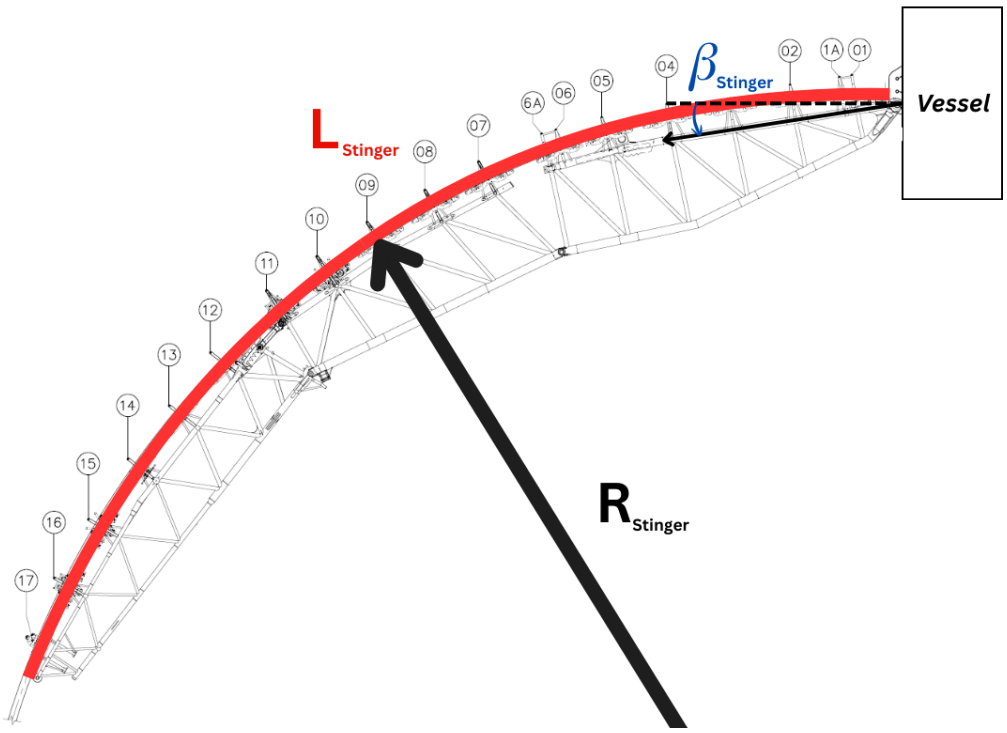


Figure B.3: Stinger configuration: Stinger Length ($L_{Stinger}$), Stinger Radius ($R_{Stinger}$), Stinger Angle ($\beta_{Stinger}$)

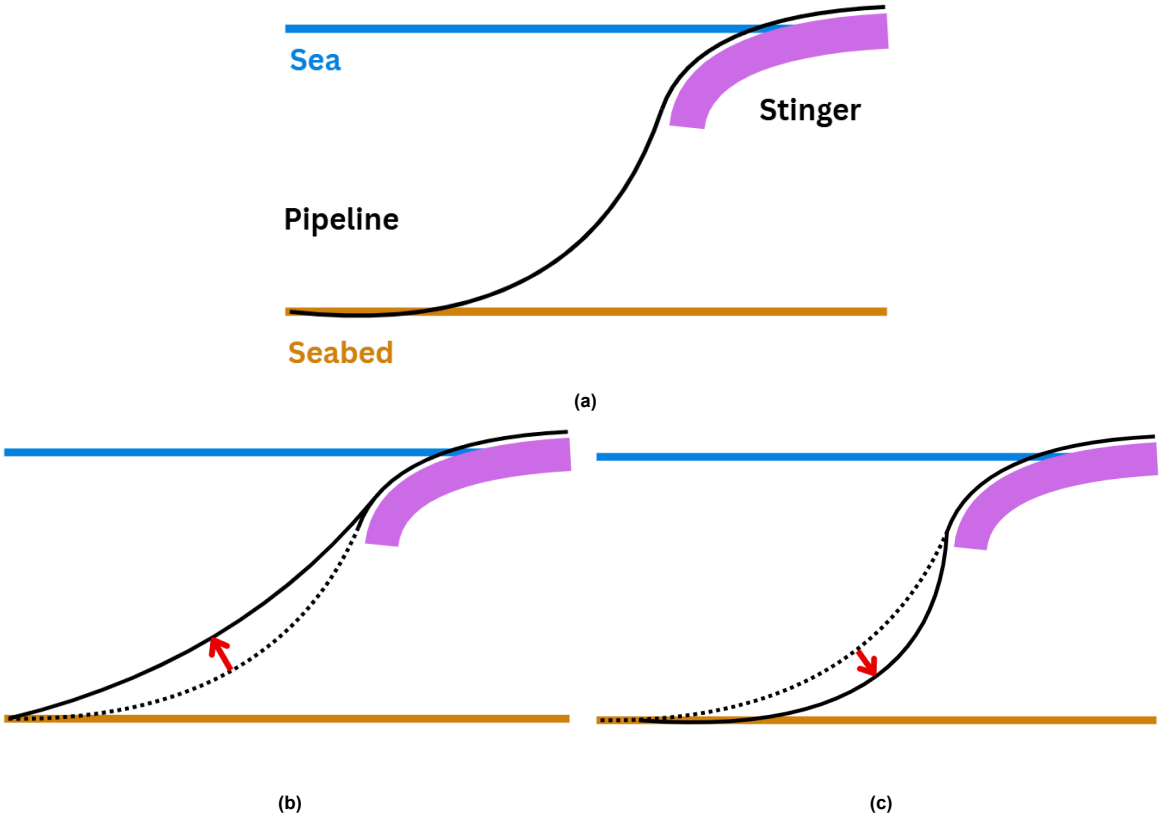


Figure B.4: a) Overview b) Pipeline after increasing tension c) Pipeline after reducing tension

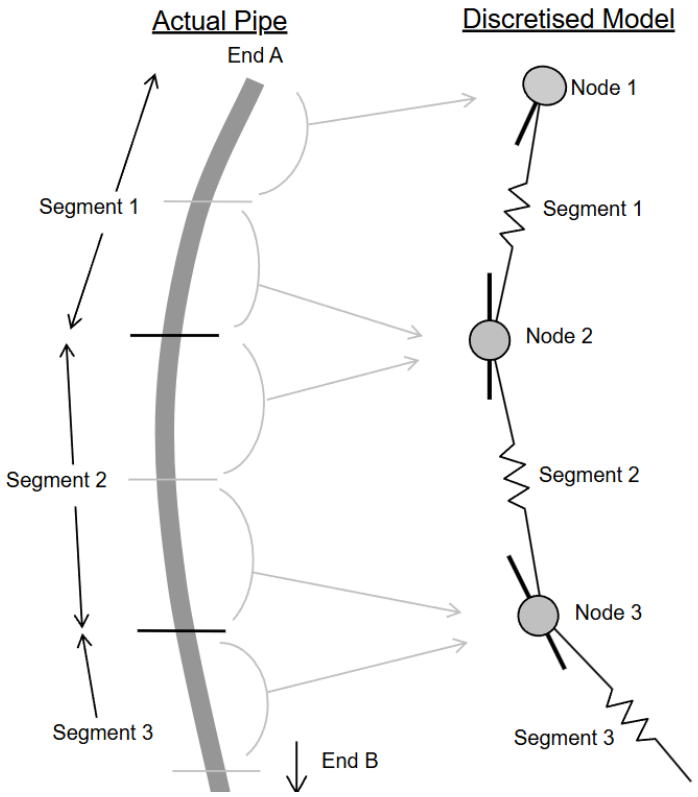


Figure B.5: Actual pipeline and its discretized model in OrcaFlex, illustratin respective nodes and springs [21]

B.1.2. Chapter 2: Problem Formulation

	Clamp	Pipe	Transition	Yoke Clamp	Rollerguide start	Taper RG	Wye	Taper RG	Taper Valve	Center Valve
Section	A-A	B-B	C-C	D-D	E-E	F-F	G-G	H-H	I-I	J-J
Distance of start point from ref. [mm]	0	490	5562	6192	6427	7427	8445	12680	13600	14120
Length [mm]	490	5072	630	235	1000	1018	4235	920	520	745
OD [mm]	571.5	508	571.5	508	571.5	609.6	647.7	609.6	571.5	508
Thickness [mm2]	62.66	30.91	62.66	30.91	62.66	81.50	91.50	81.50	62.66	30.91
ID [mm]	446.18	446.18	446.18	446.18	446.18	446.18	446.18	446.18	446.18	446.18
Moment of inertia [mm4]	3.29E+09	1.32E+09	3.29E+09	1.32E+09	3.29E+09	4.82E+09	6.35E+09	4.82E+09	3.29E+09	1.32E+09
Area [mm2]	100166.3	46328.6	100166.3	46328.6	100166.3	1.91E+05	2.81E+05	1.91E+05	100166.3	46328.6
Area [m2]	0.1002	0.0463	0.1002	0.0463	0.1002	0.1907	0.2812	0.1907	0.1002	0.0463
Offset [mm]	31.75	0	31.75	0	65.59	86.295	107	86.295	65.59	0
Contact diameter [mm]	571.5		571.5		639.2	680.5	722.0	680.5	639.2	
Volume [m^3]	0.13	1.03	0.16	0.05	0.26	0.30	1.40	0.27	0.13	0.15
Displacement [t]	0.13	1.05	0.17	0.05	0.26	0.30	1.43	0.28	0.14	0.15
Weight of steel [t]	0.39	1.84	0.50	0.09	0.79	1.52	9.35	1.38	0.41	0.27
Weight total [t]	1.24	0.79	3.50	0.04	0.52	6.62	24.31	1.56	0.27	5.89
Weight [t/m]	2.52	0.16	5.56	0.16	0.52	6.50	5.74	1.69	0.52	7.91
Ramberg Osgood [Coeffecient A]	0.0329	0.0762	0.0329	0.0762	0.0329	0.0276	0.0265	0.0276	0.0329	0.0762
Ramberg Osgood [Exponent B]	12.1475	11.4646	12.1475	11.4646	12.1475	11.9910	11.8997	11.9910	12.1475	11.4646

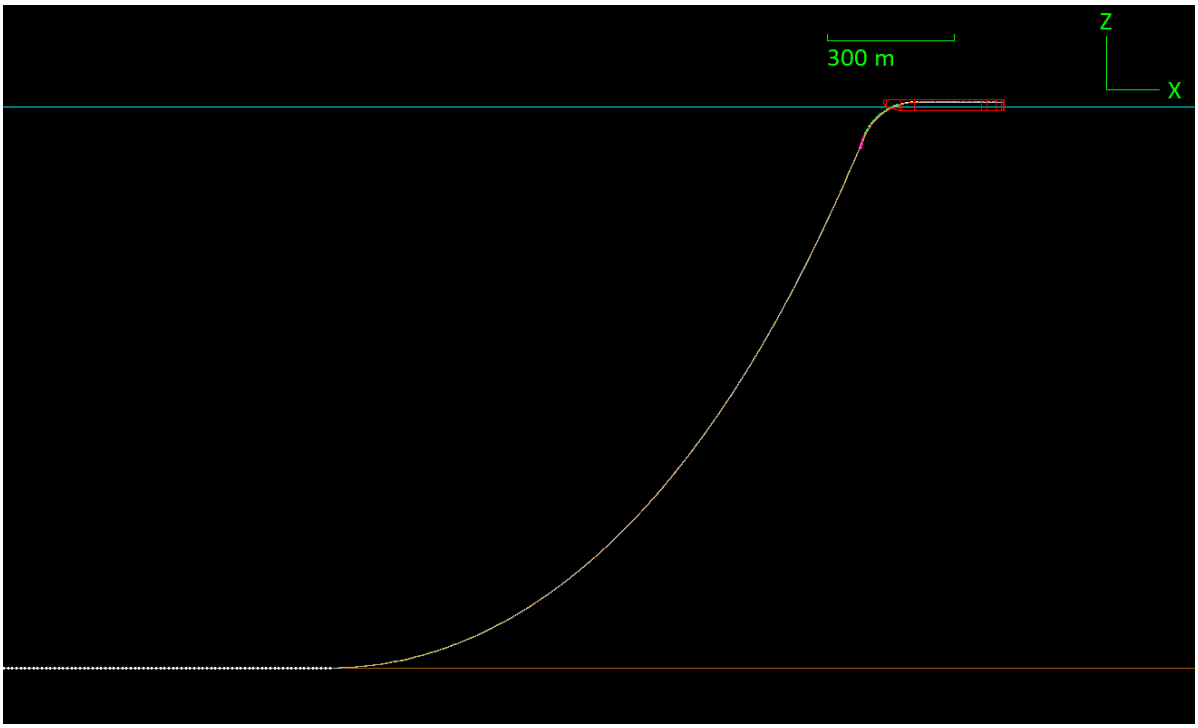
(a)

	Taper Valve	Taper RG2		Mudmat tower	Before T tower	Tower	Tower-Mudmat tower	Mudmat tower	Pipe	Yoke clamp
Section	K-K	L-L	M-M	N-N	O-O	P-P	Q-Q	R-R	S-S	T-T
Distance of start point from ref. [mm]	14865	15163	16115	16377	16920	18773.41	18943.41	20034.41	20577.41	22987.41
Length [mm]	298	952	262	543	1853.41	170	1091	543	2410	630
OD [mm]	508	508	508	571.5	508	571.5	508	571.5	508	571.5
Thickness [mm2]	30.91	30.91	30.91	62.66	30.91	62.66	30.91	62.66	30.91	62.66
ID [mm]	446.18	446.18	446.18	446.18	446.18	446.18	446.18	446.18	446.18	446.18
Moment of inertia [mm4]	1.32E+09	1.32E+09	1.32E+09	3.29E+09	1.32E+09	3.29E+09	1.32E+09	3.29E+09	1.32E+09	3.29E+09
Area [mm2]	46328.6	46328.6	46328.6	100166.3	46328.6	100166.3	46328.6	100166.3	46328.6	100166.3
Area [m2]	0.0463	0.0463	0.0463	0.1002	0.0463	0.1002	0.0463	0.1002	0.0463	0.1002
Offset [mm]	31.75	47.625	63.5	63.5	63.5	63.5	47.625	31.75	0	31.75
Contact diameter [mm]	571.5	603.3	635.0	635.0	635.0	635.0	603.3	571.5		571.5
Volume [m^3]	0.06	0.19	0.05	0.14	0.38	0.04	0.22	0.14	0.49	0.16
Displacement [t]	0.06	0.20	0.05	0.14	0.39	0.04	0.23	0.14	0.50	0.17
Weight of steel [t]	0.11	0.35	0.10	0.43	0.67	0.13	0.40	0.43	0.88	0.50
Weight total [t]	2.36	7.53	2.07	4.29	14.65	1.34	8.63	0.28	0.38	0.33
Weight [t/m]	7.91	7.91	7.91	7.91	7.91	7.91	7.91	0.52	0.16	0.52
Ramberg Osgood [Coeffecient A]	0.0762	0.0762	0.0762	0.0329	0.0762	0.0329	0.0762	0.0329	0.0762	0.0329
Ramberg Osgood [Exponent B]	11.4646	11.4646	11.4646	12.1475	11.4646	12.1475	11.4646	12.1475	11.4646	12.1475

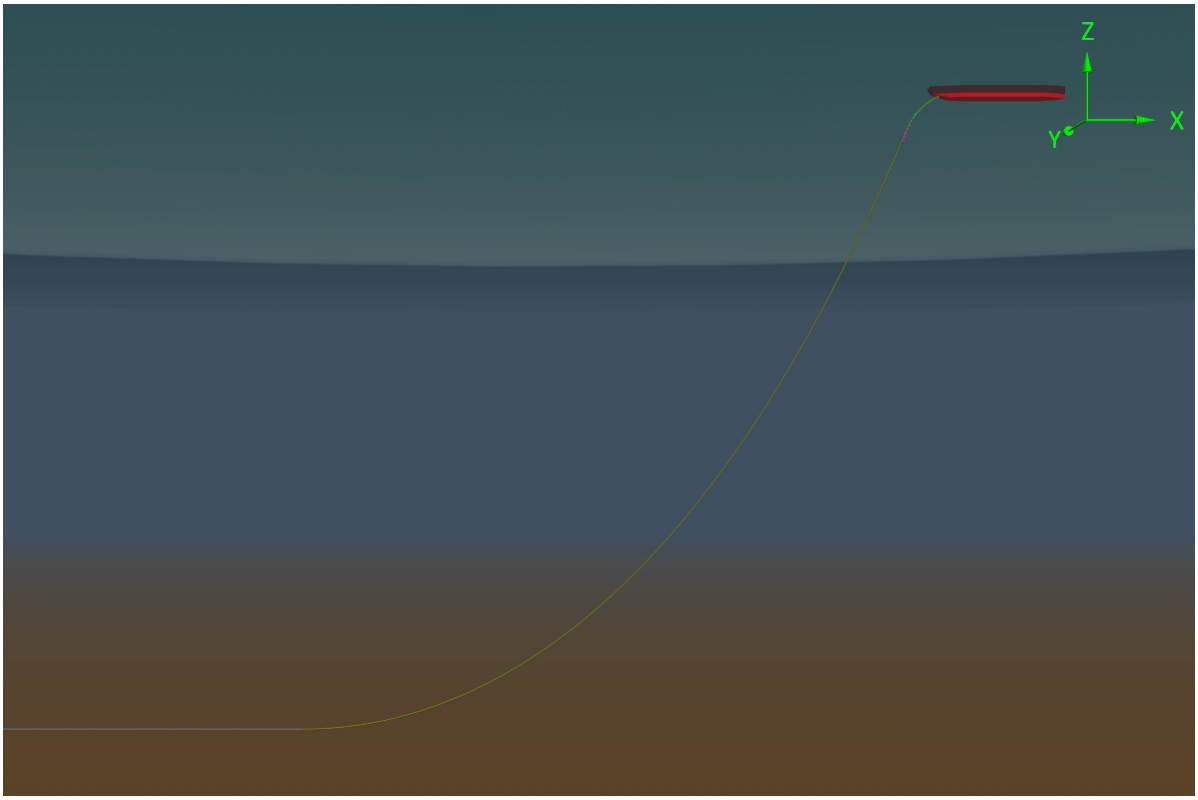
(b)

Figure B.6: OrcaFlex inline structure section property sheet

B.1.3. Chapter 3: Method

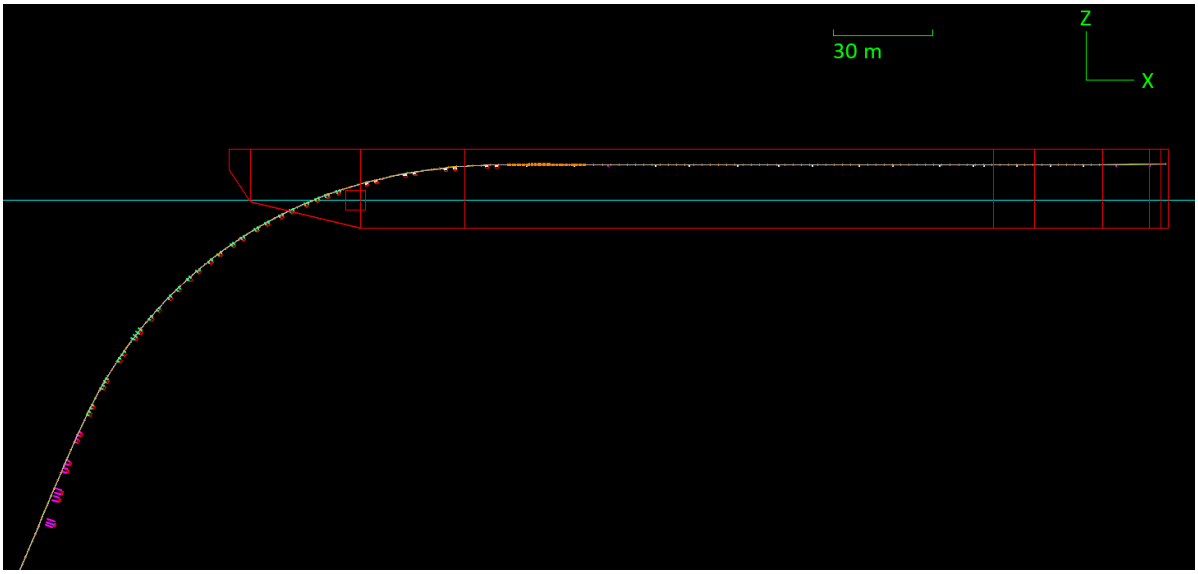


(a) 2D view

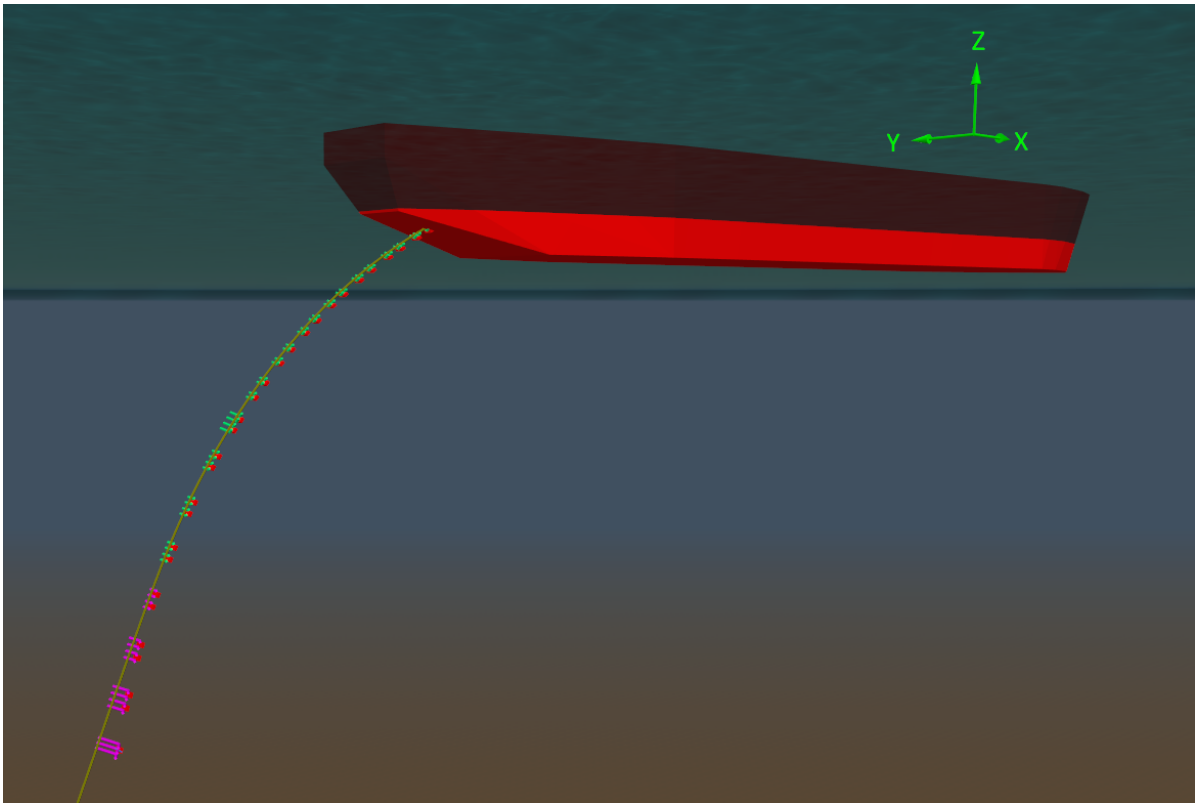


(b) 3D view

Figure B.7: Full OrcaFlex model: a) 2D view b) 3D view

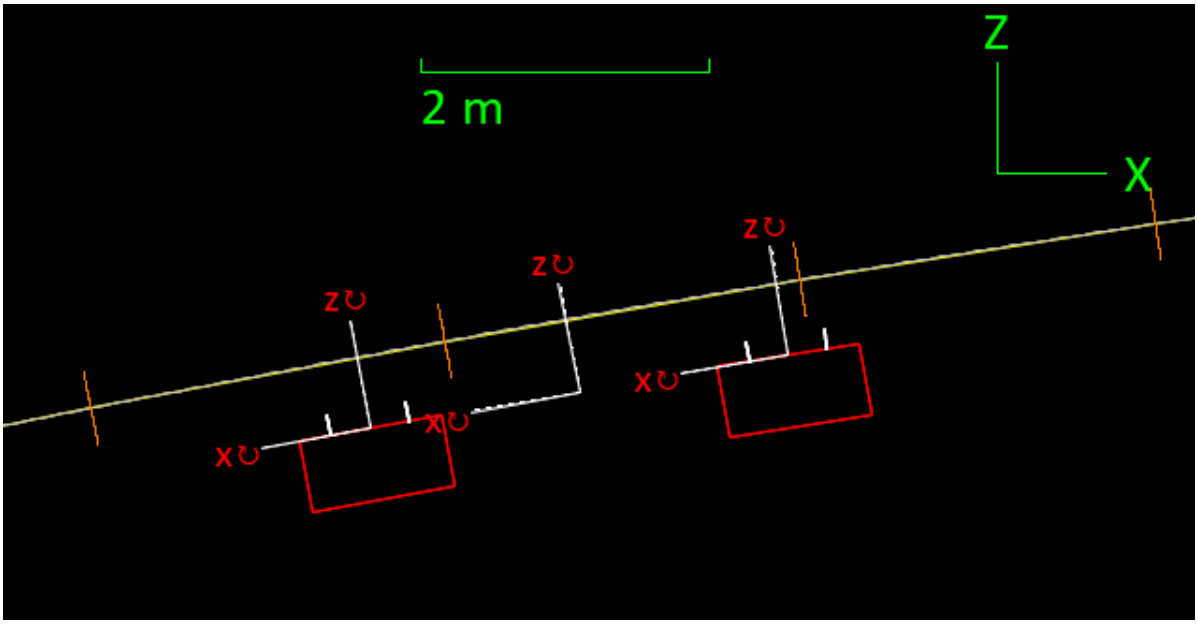


(a) 2D view

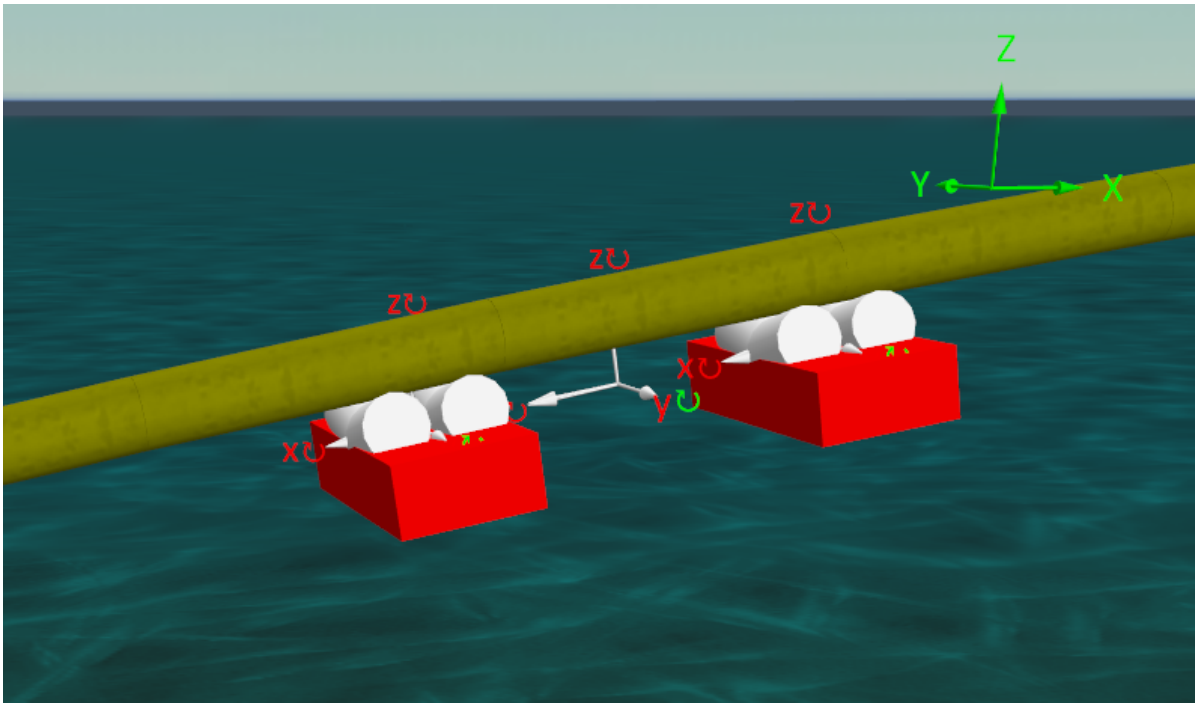


(b) 3D view

Figure B.8: Zoom into the vessel and stinger in the OrcaFlex model: a) 2D view b) 3D view

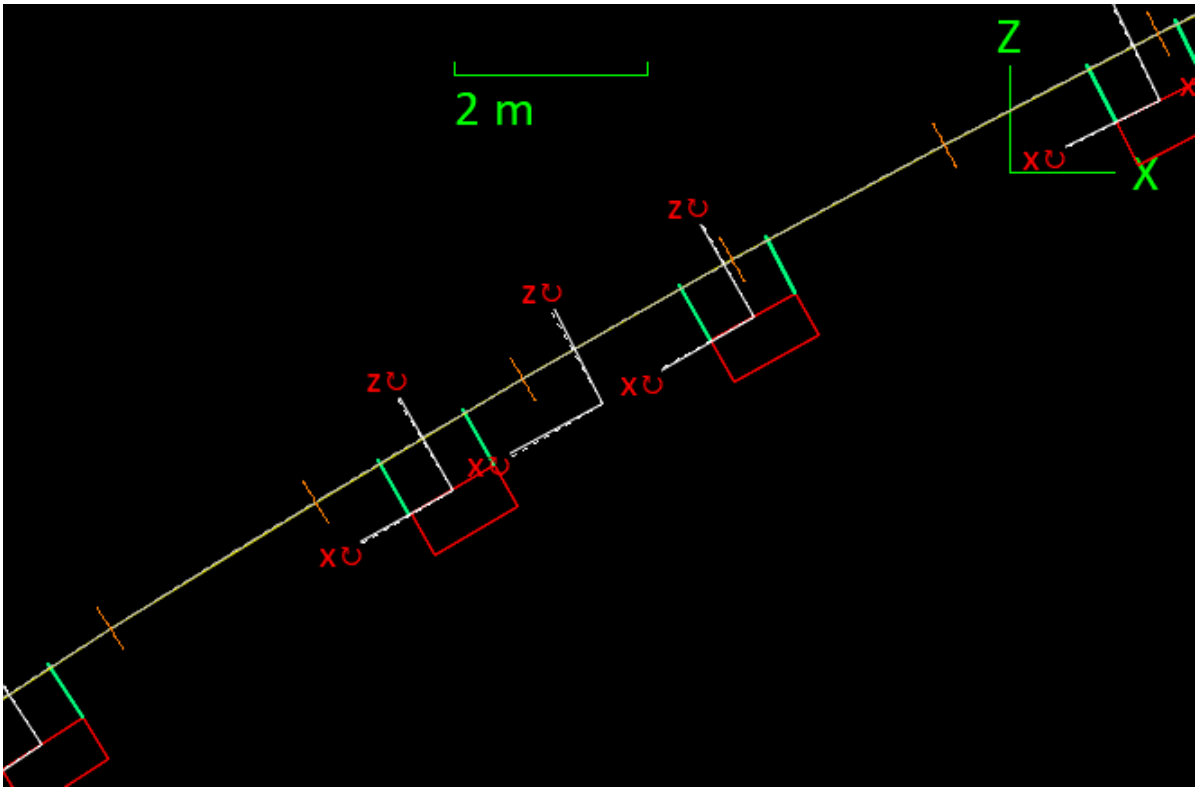


(a) 2D view

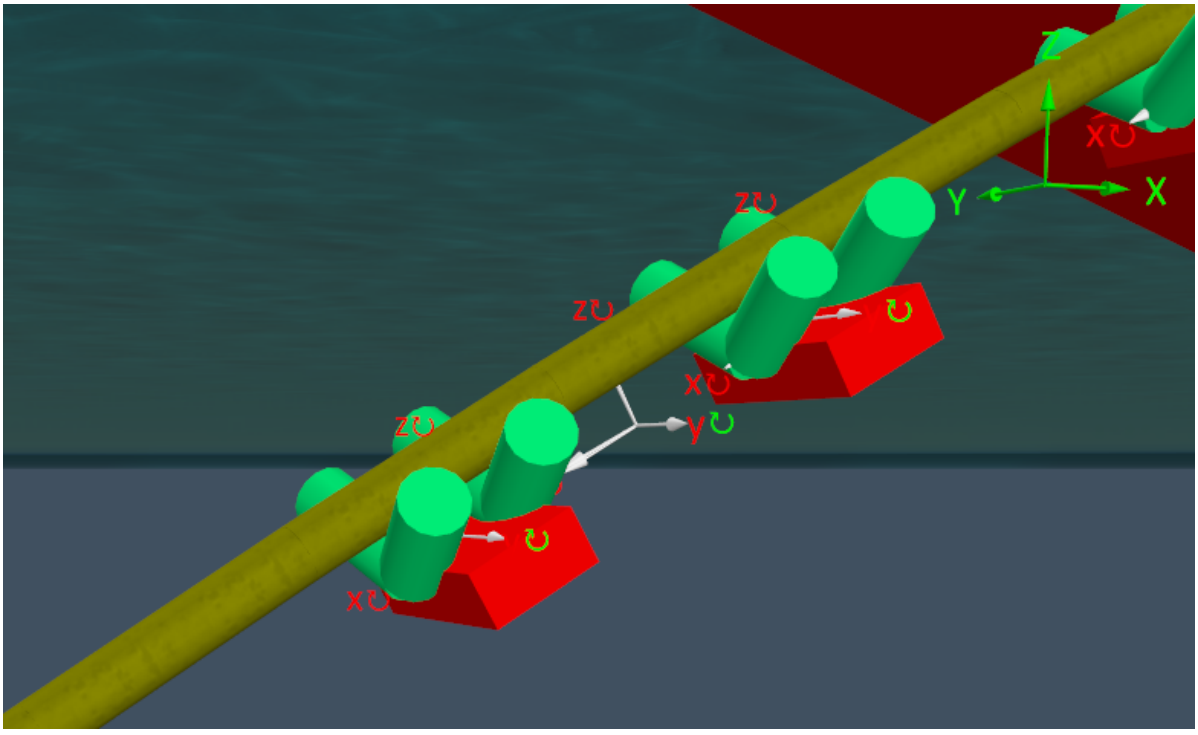


(b) 3D view

Figure B.9: Zoom into a vessel rollerbox in the OrcaFlex model: a) 2D view b) 3D view

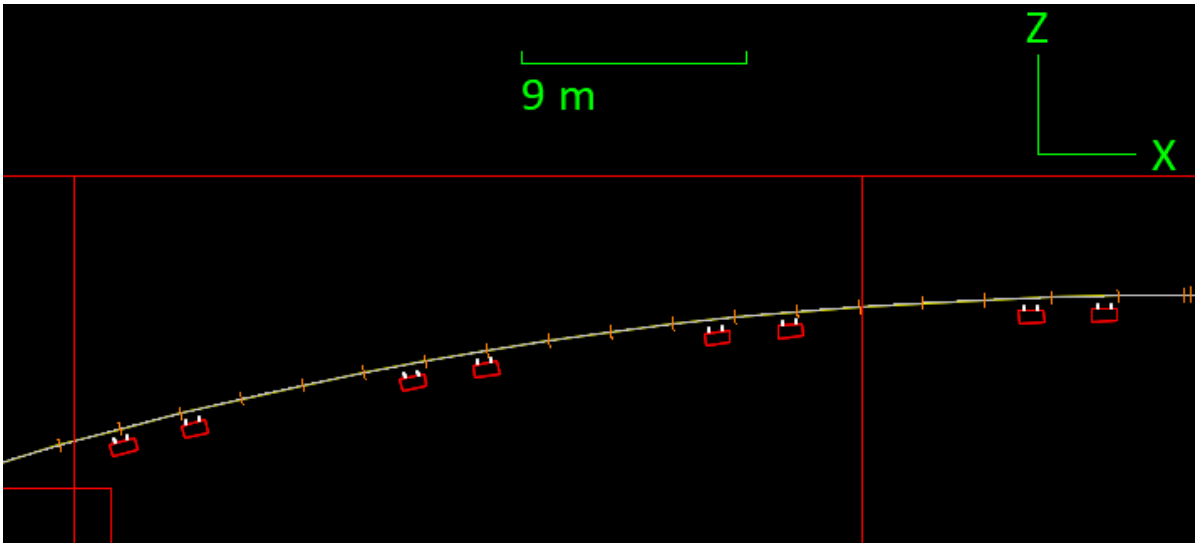


(a) 2D view

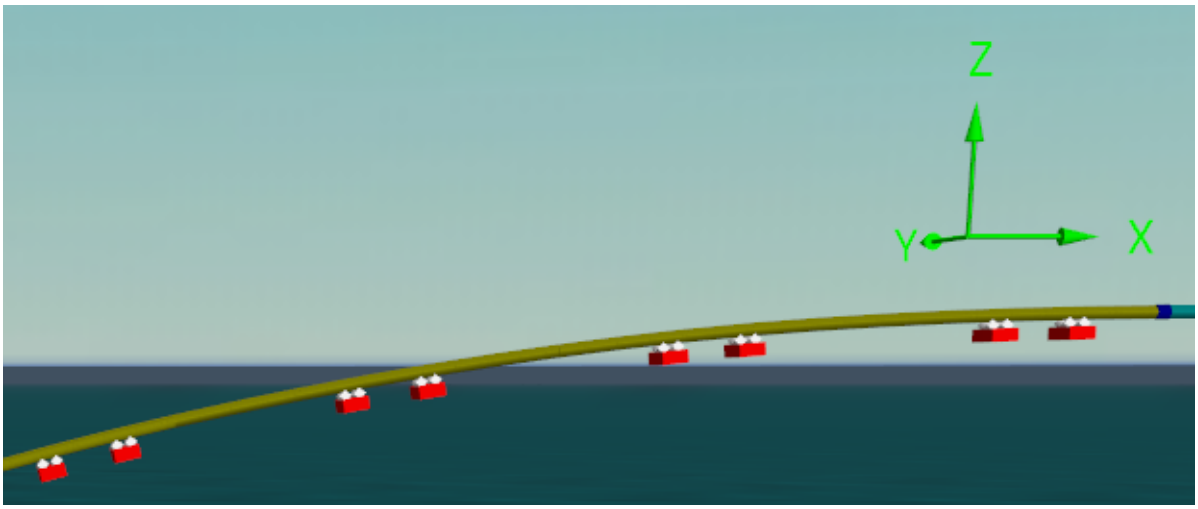


(b) 3D view

Figure B.10: Zoom into a stinger rollerbox in the OrcaFlex model: a) 2D view b) 3D view

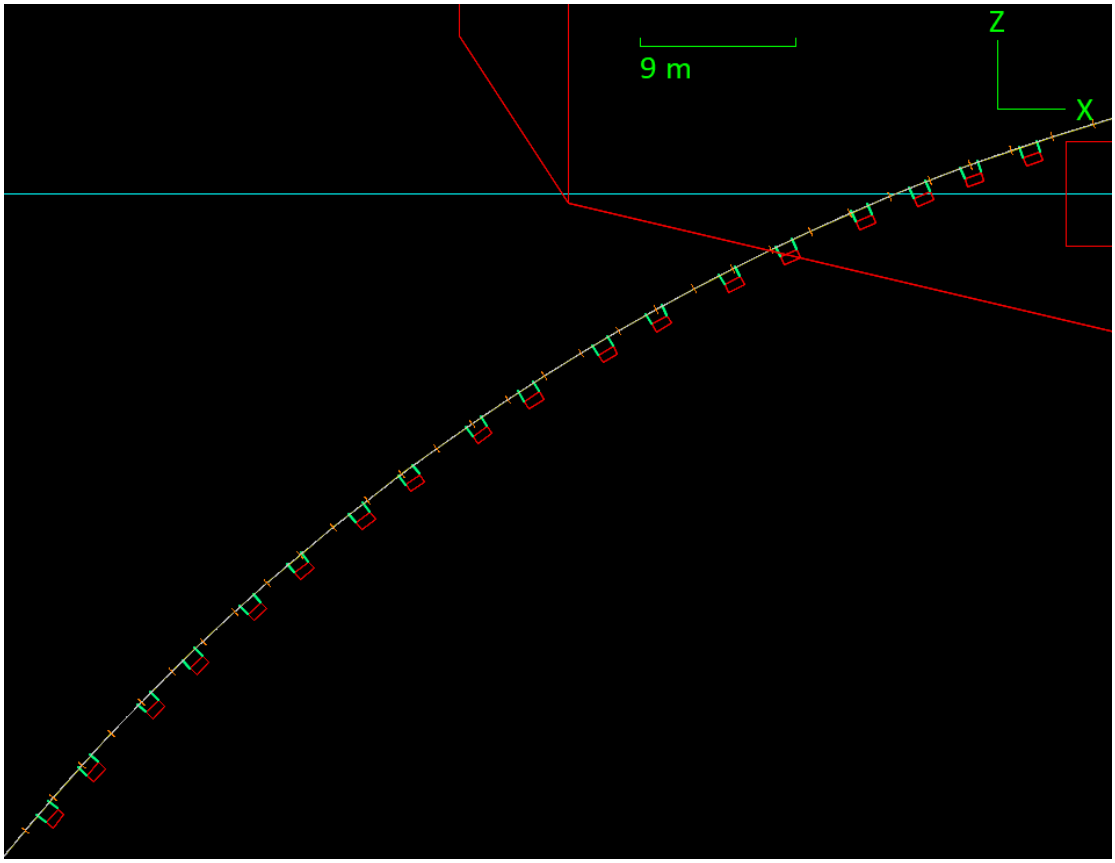


(a) 2D view

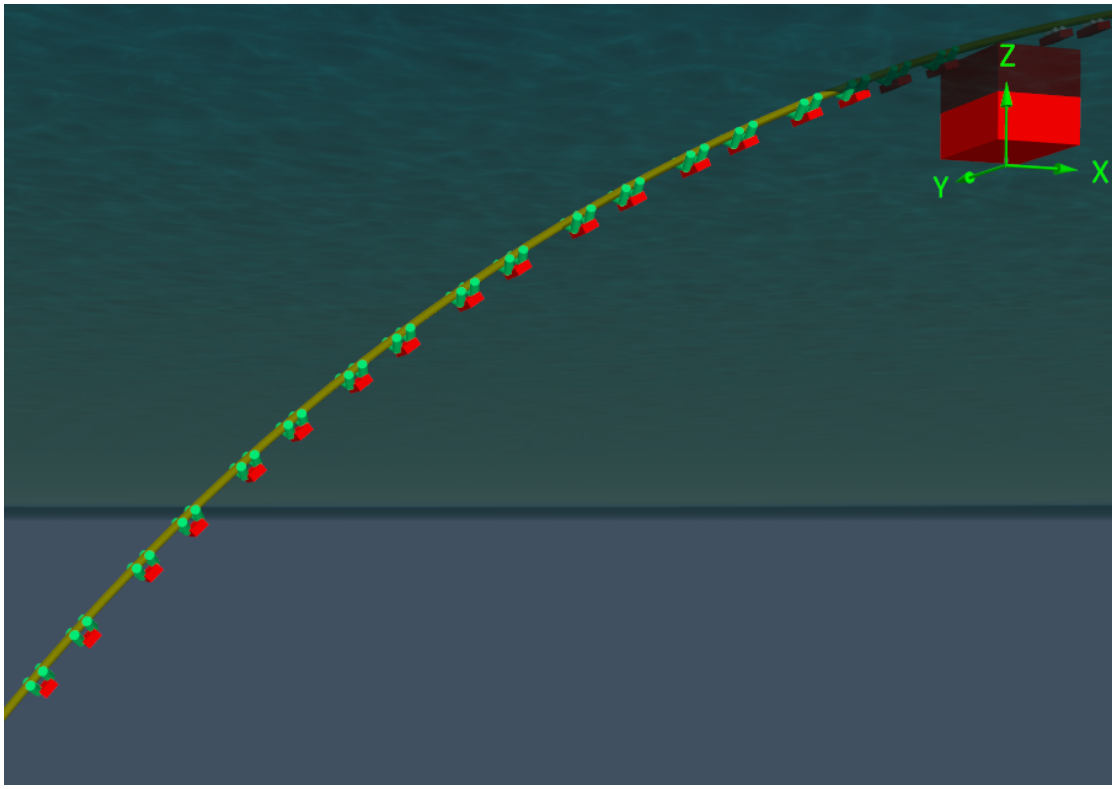


(b) 3D view

Figure B.11: Zoom into the vessel rollerboxes V13-V16 in the OrcaFlex model: a) 2D view b) 3D view

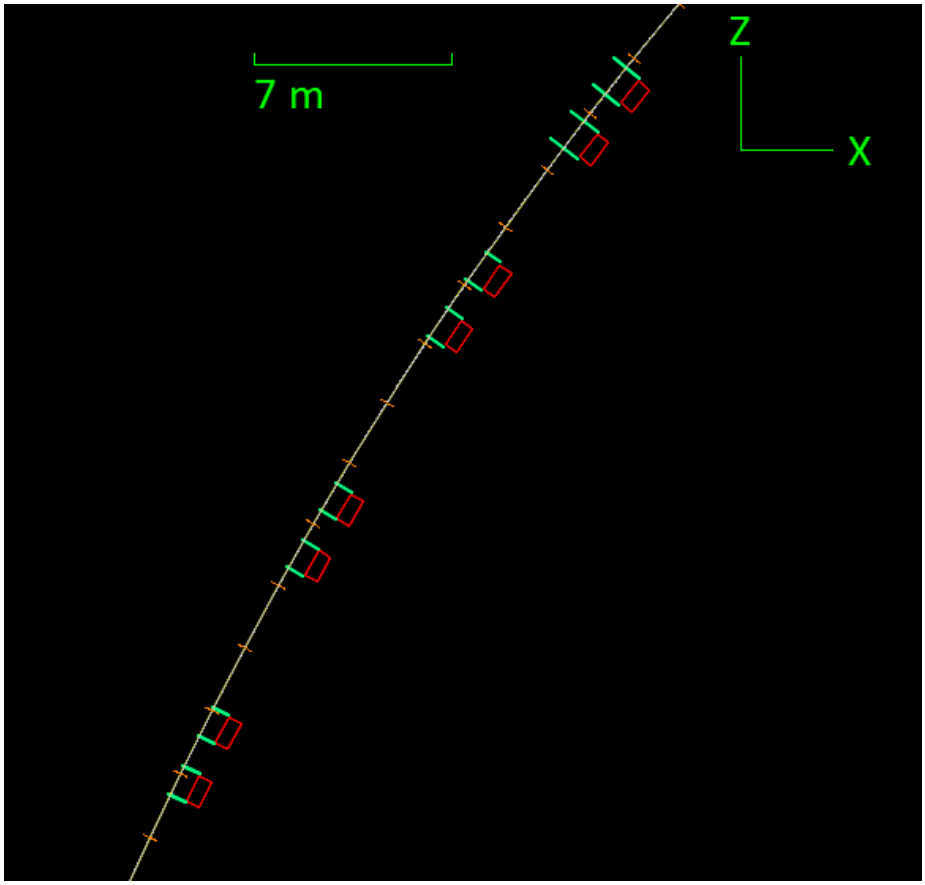


(a) 2D view

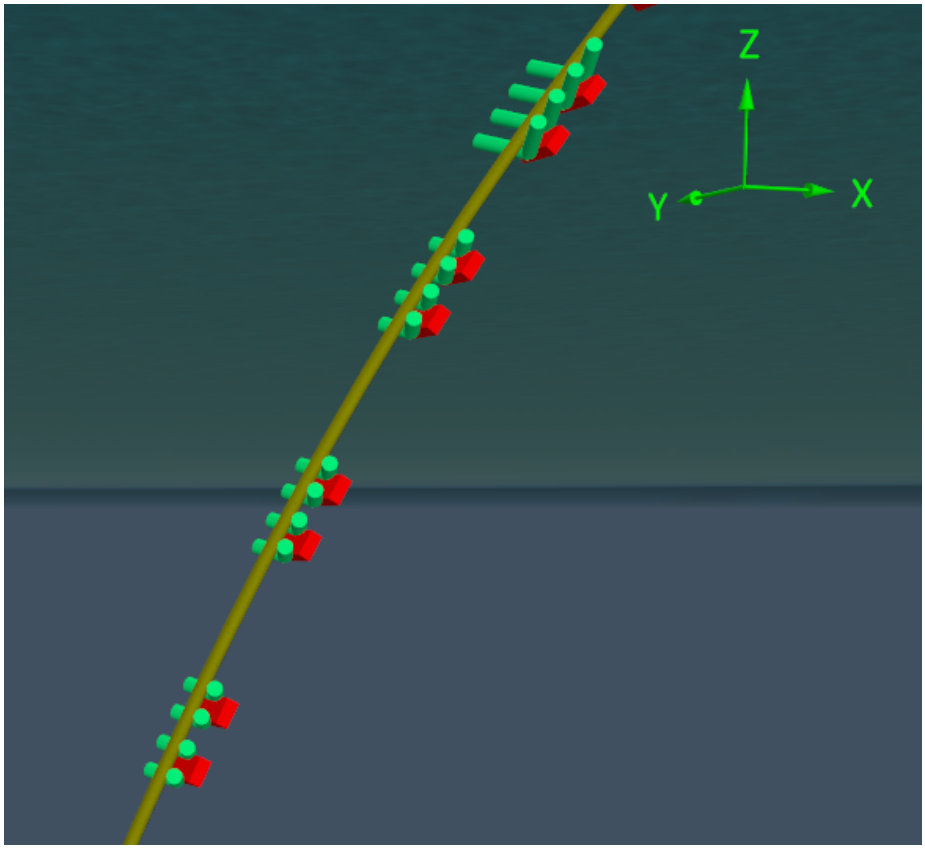


(b) 3D view

Figure B.12: Zoom into the stinger rollerboxes S1-S9 in the OrcaFlex model: a) 2D view b) 3D view

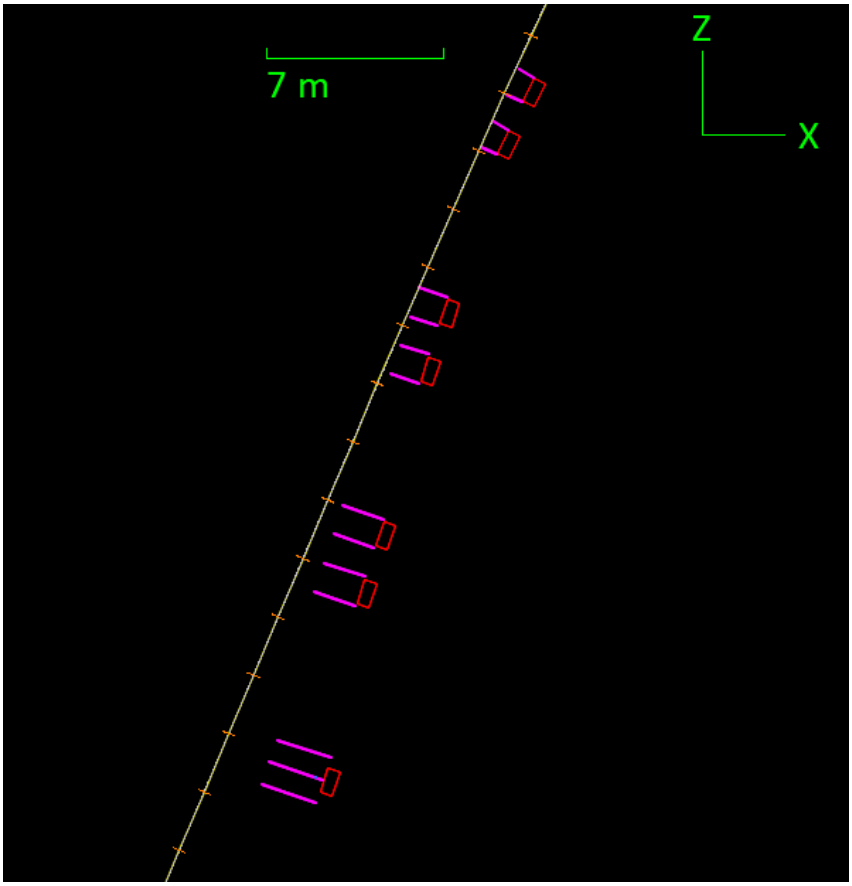


(a) 2D view

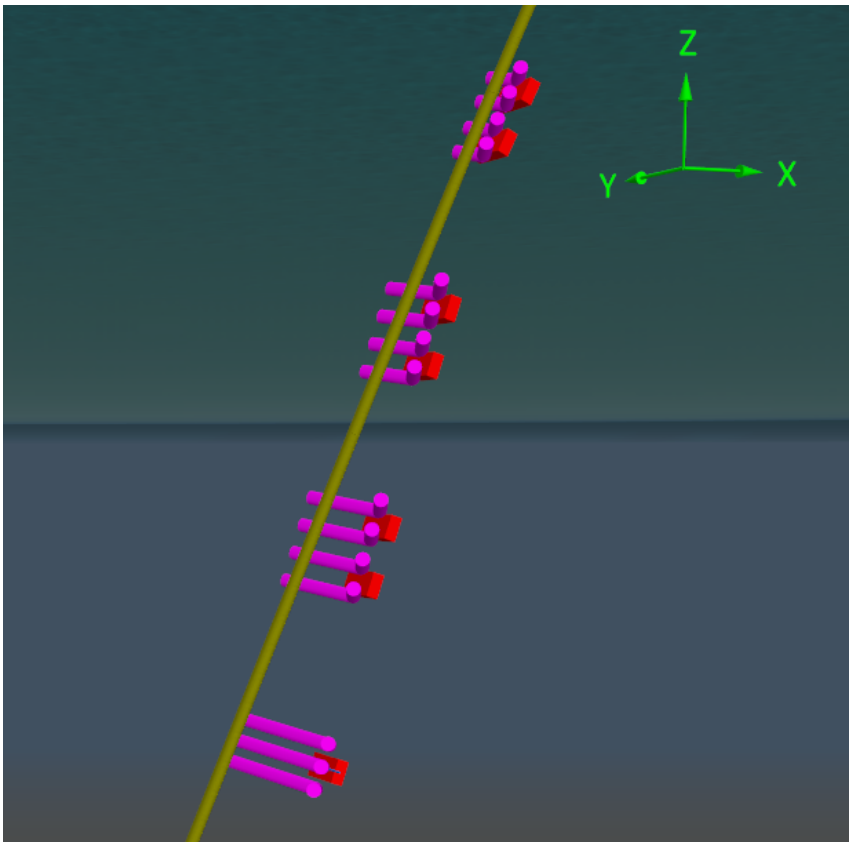


(b) 3D view

Figure B.13: Zoom into the stinger rollerboxes S10-S13 in the OrcaFlex model: a) 2D view b) 3D view

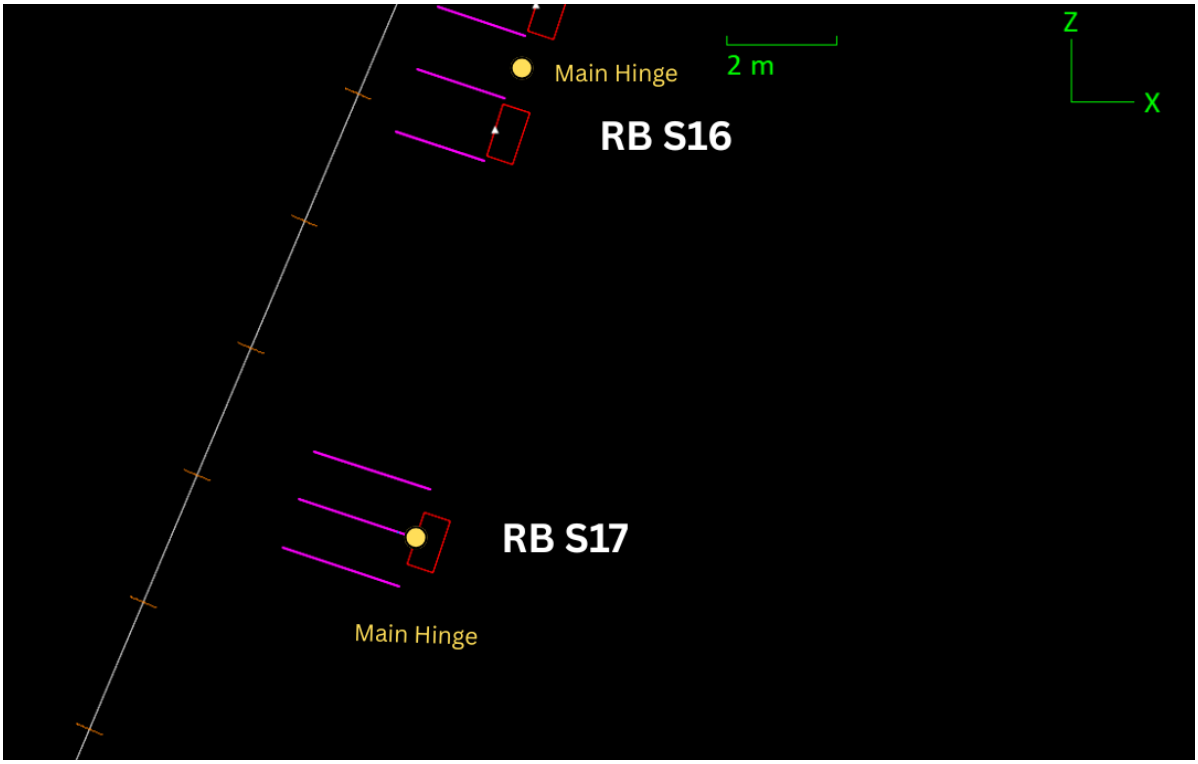


(a) 2D view



(b) 3D view

Figure B.14: Zoom into the stinger rollerboxes S14-S17 in the OrcaFlex model: a) 2D view b) 3D view

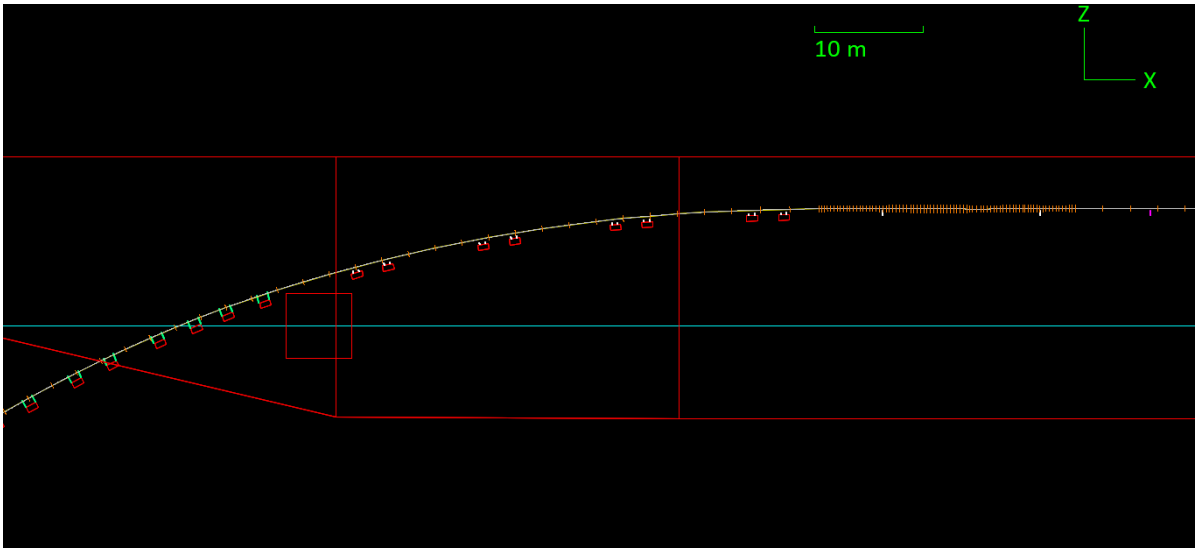


(a) 2D view

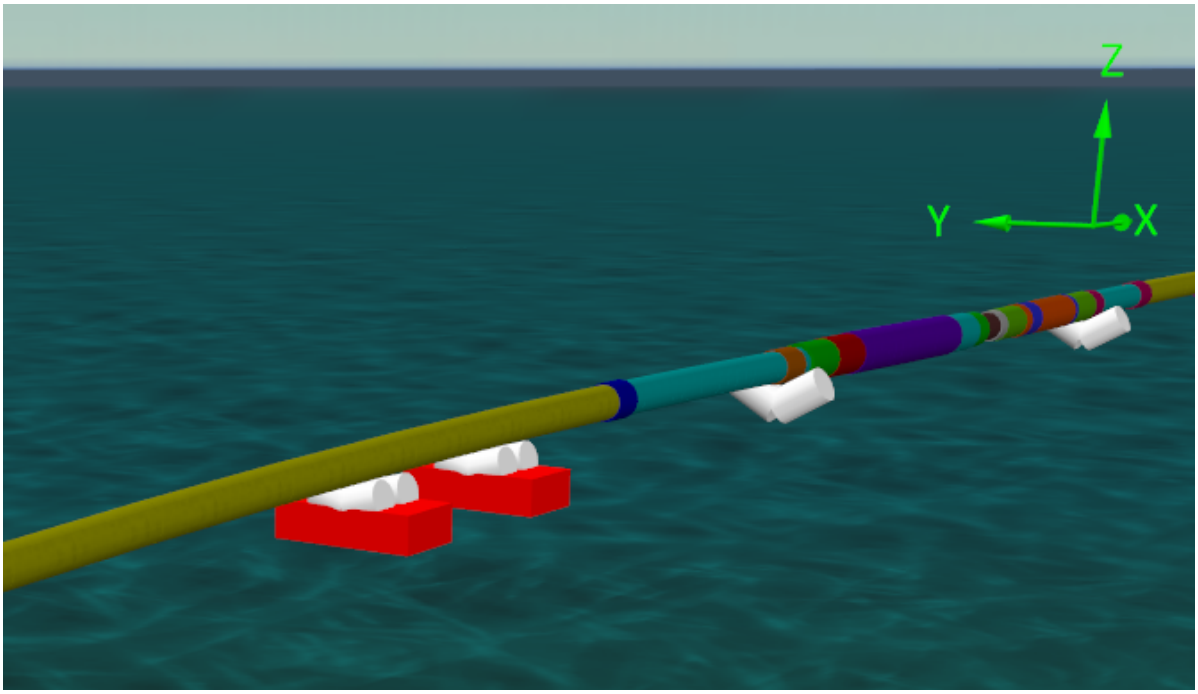


(b) 3D view

Figure B.15: Zoom into the stinger rollerbox S17 in the OrcaFlex model: a) 2D view b) 3D view



(a) 2D view



(b) 3D view

Figure B.16: Zoom into the vessel and inline structure in the OrcaFlex model: a) 2D view b) 3D view

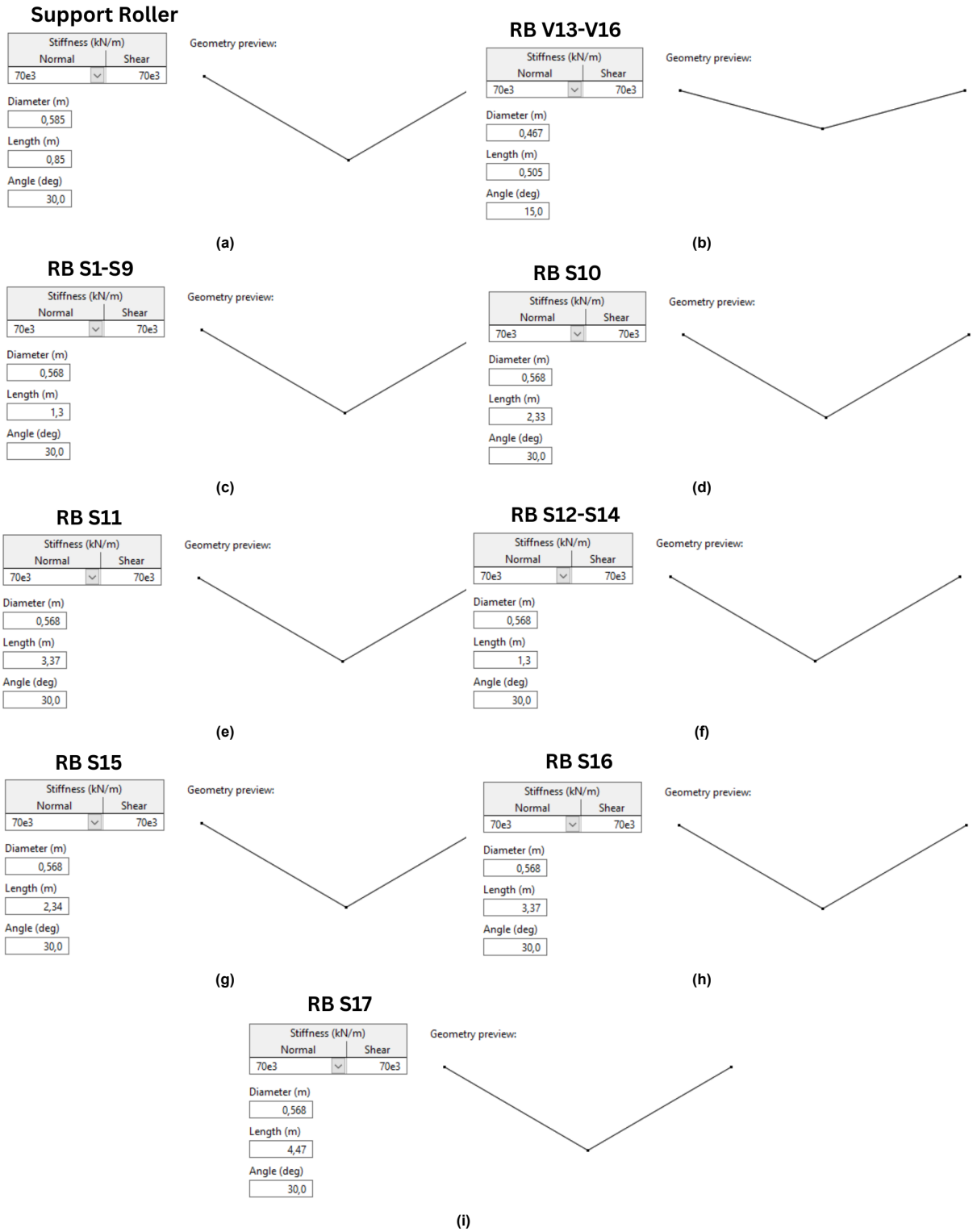


Figure B.17: Rollerbox properties, showing normal and shear stiffness, diameter, length and angle (from the horizontal axis), for: a) Support Roller b) Rollerbox V13 - V16 c) Rollerbox S1 - S9 d) Rollerbox S10 e) Rollerbox S11 f) Rollerbox S12 - S14 g) Rollerbox S15 h) Rollerbox S16 i) Rollerbox S17

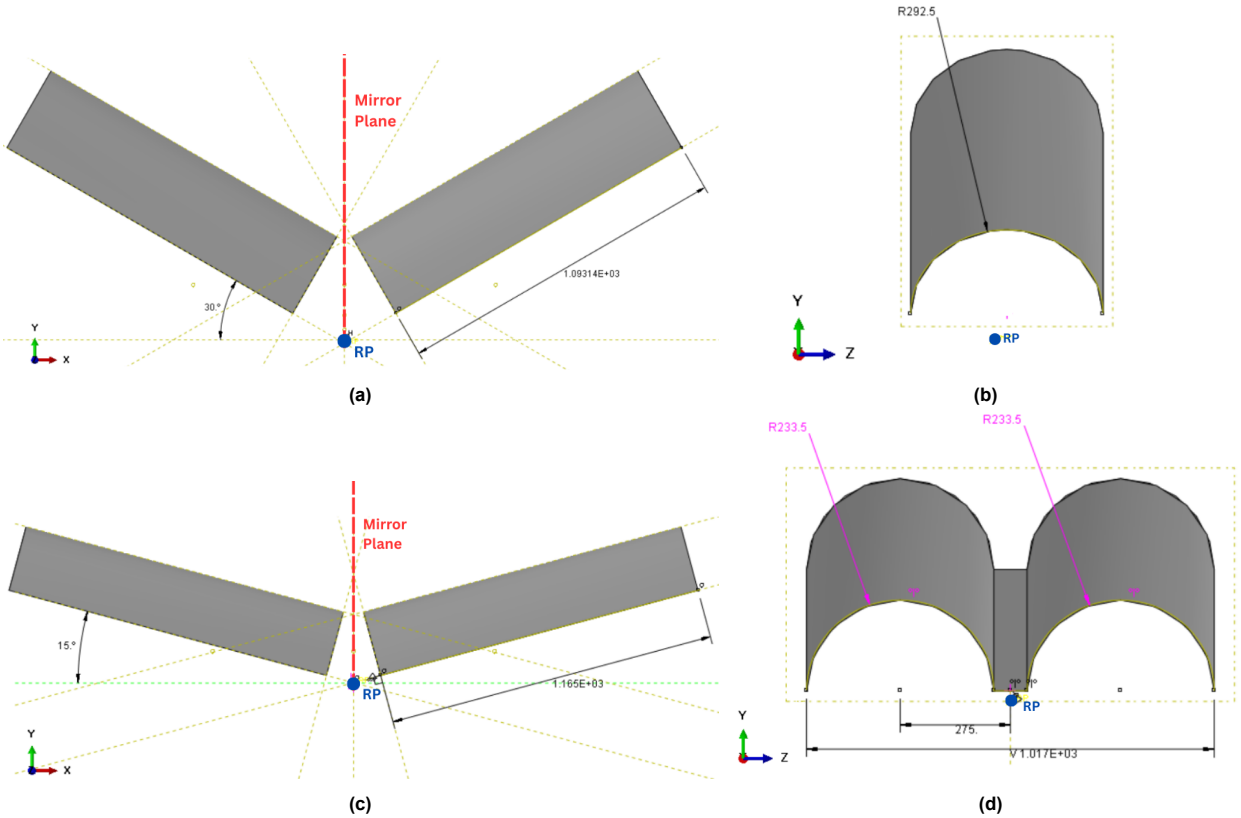


Figure B.18: Support rollers and vessel rollerbox dimension, mirror plane, and reference point: a) Support Roller 1, 2, 3 Front view b) Support Roller 1, 2, 3 Side view c) RB V13-V16 Front view d) RB V13-V16 Side view

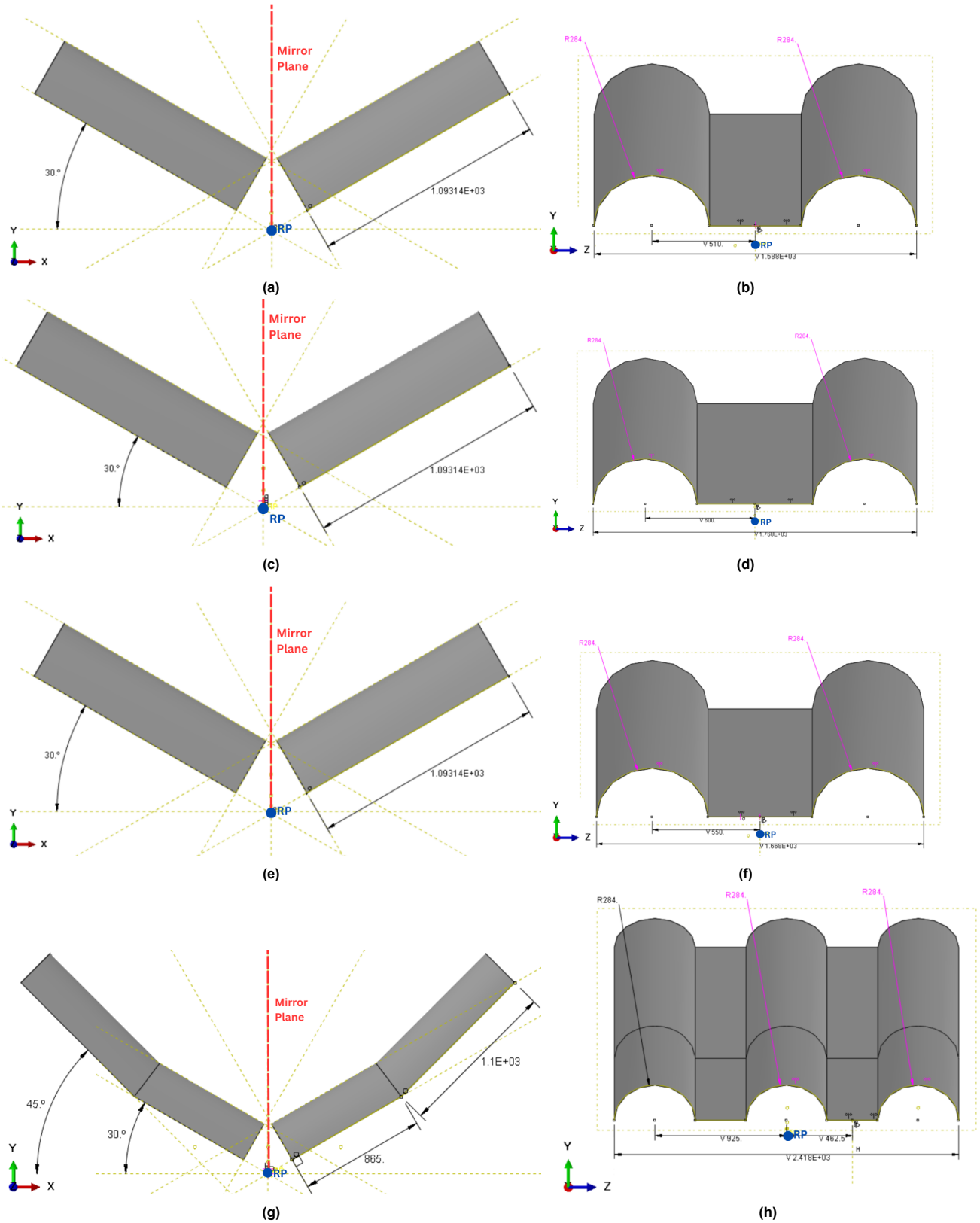


Figure B.19: Stinger rollerbox dimension, mirror plane, and reference point: a) S1-S9 Front view b) S1-S9 Side view c) S10, S11, S15, S16 Front view d) S10, S11, S15, S16 Side view e) S12-S14 Front view f) S12-S14 Side view g) S17 Front view h) S17 Side view

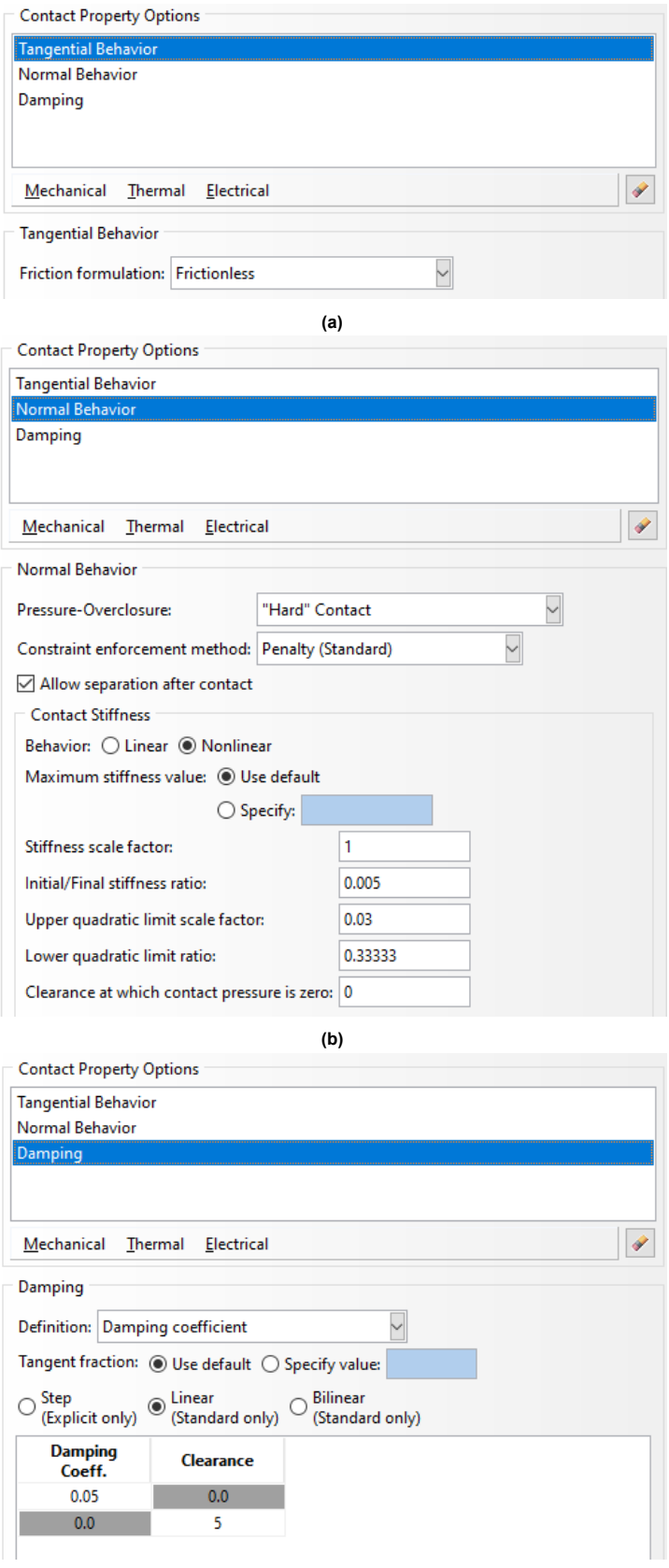


Figure B.20: Interaction properties for frictionless sliding between the pipeline and the rollerboxes: a) Tangential Behaviour b) Normal Behaviour c) Damping

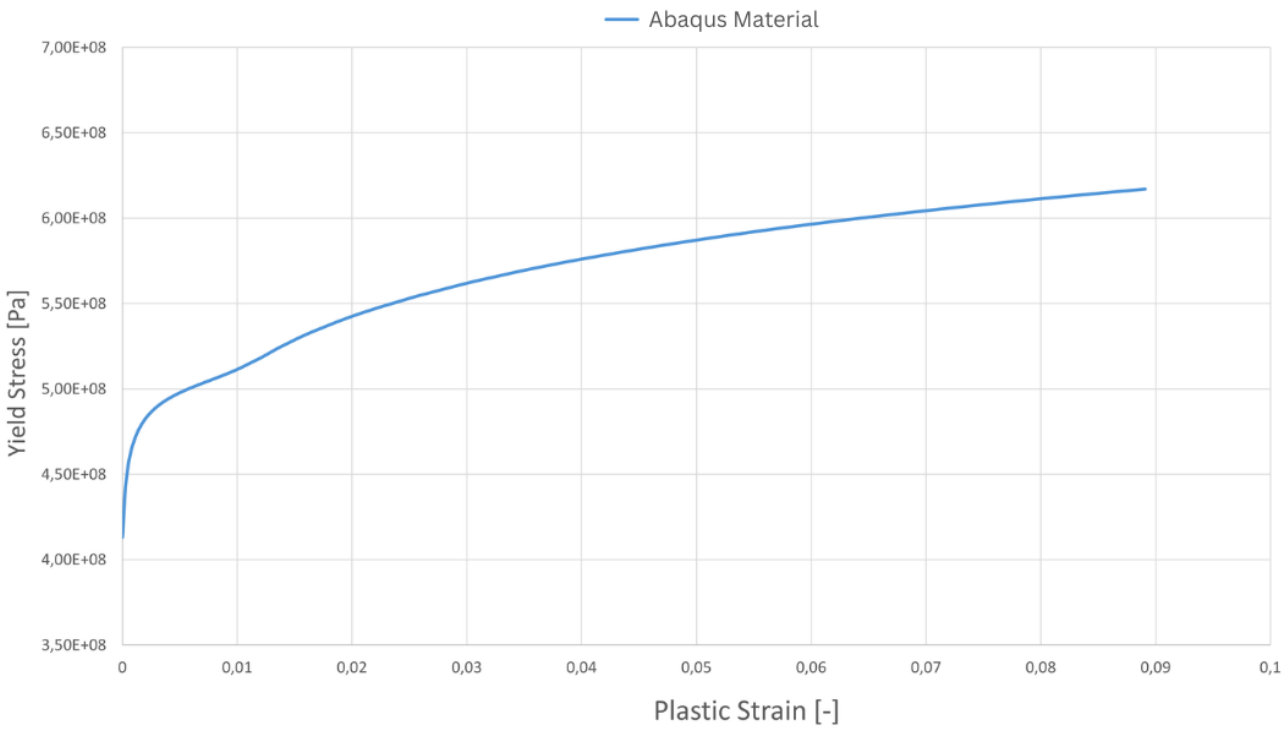


Figure B.21: Stress-Strain curve for the pipeline in Abaqus, for plastic behaviour

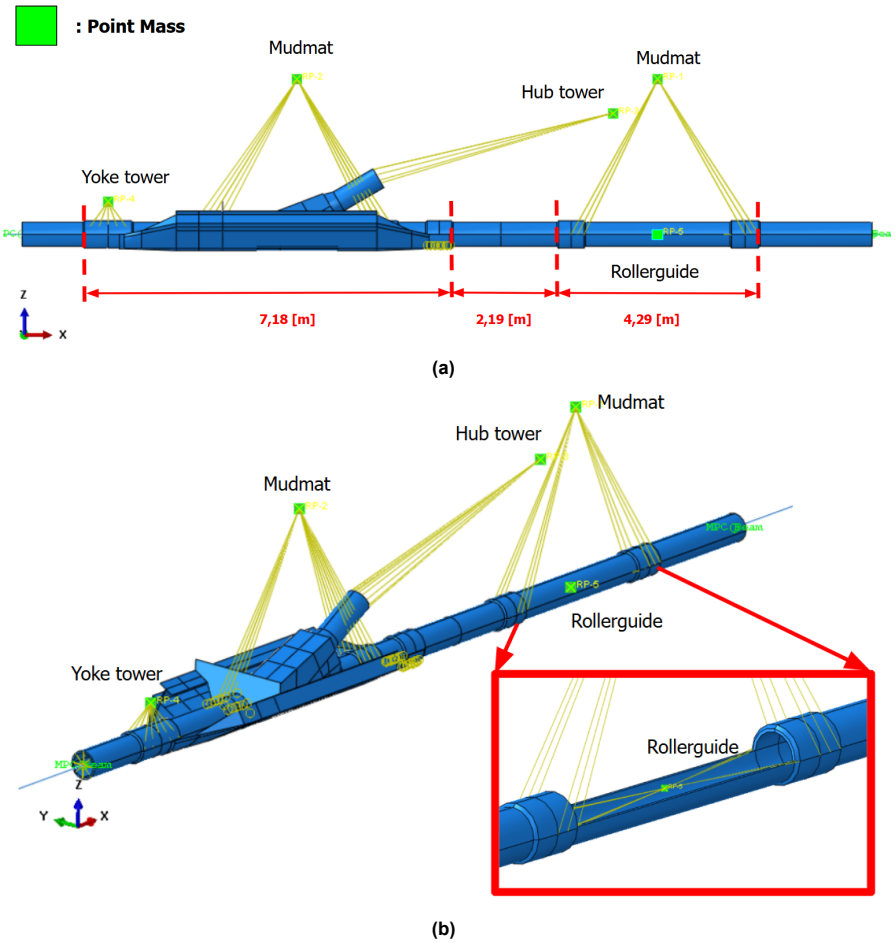


Figure B.22: Simplified Inline Structure in Abaqus, with respective point masses: a) Side view b) Isometric view

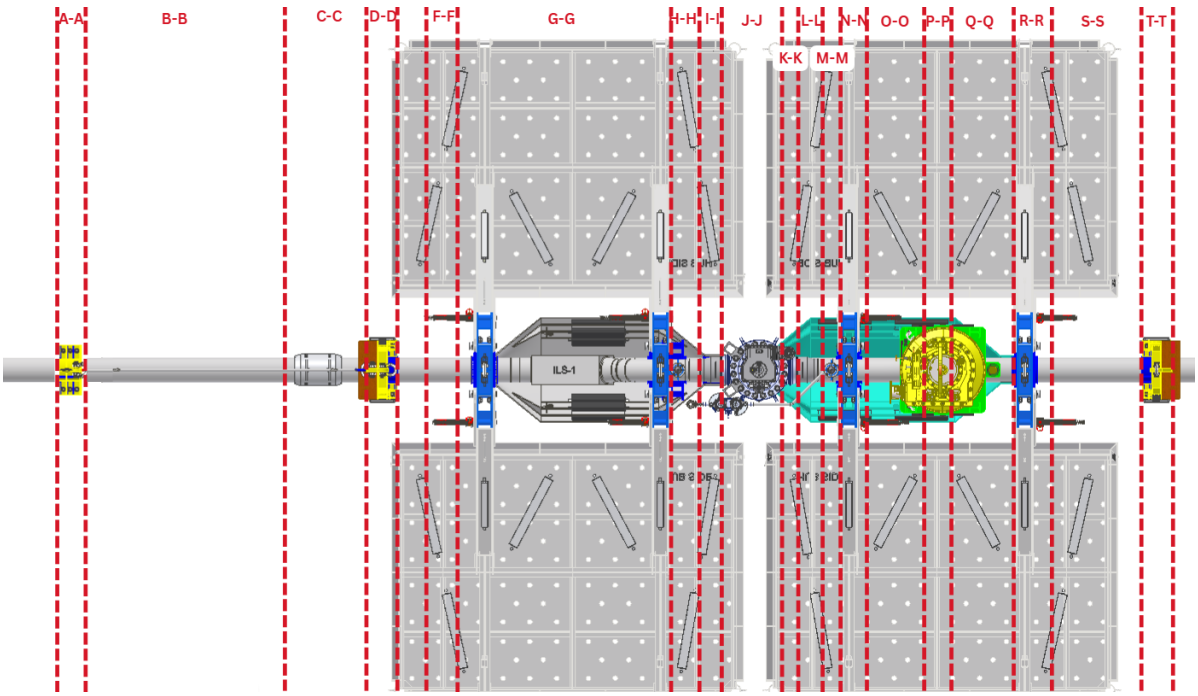


Figure B.23: Top view of the inline structure with OrcaFlex sections

B.1.4. Chapter 4: Results

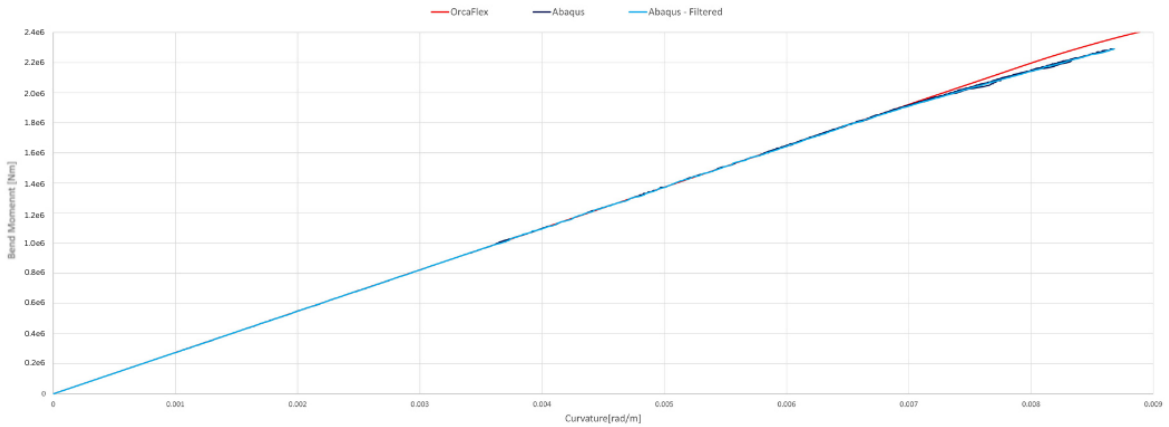


Figure B.24: Material curves comparison, showing the old OrcaFlex and Abaqus material, and the new filtered material

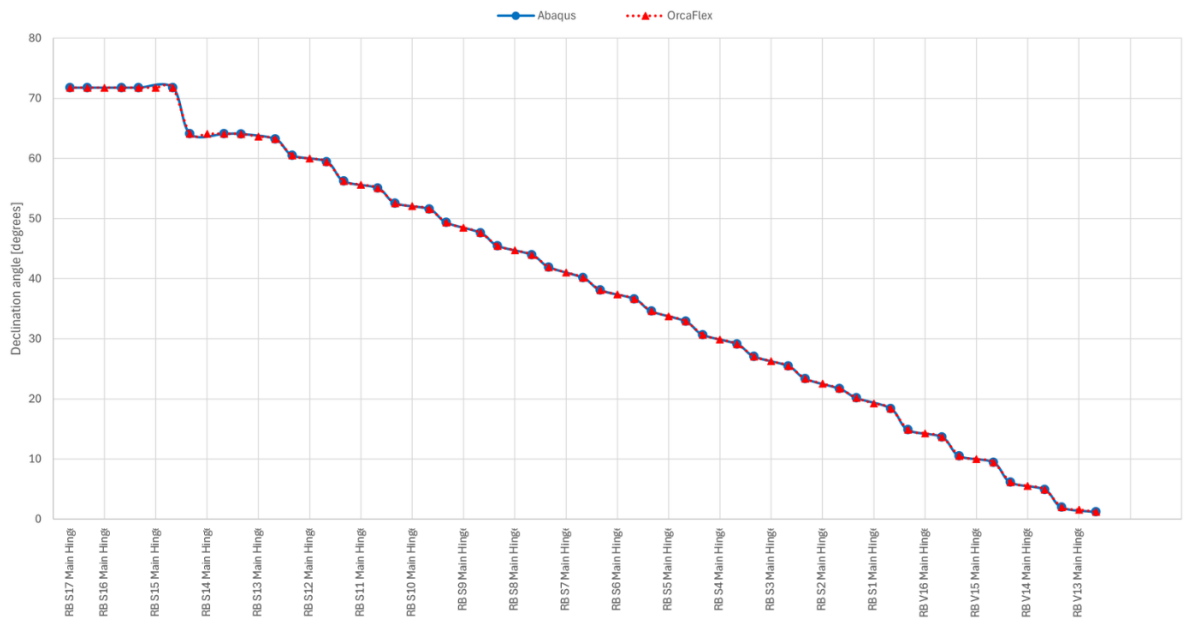
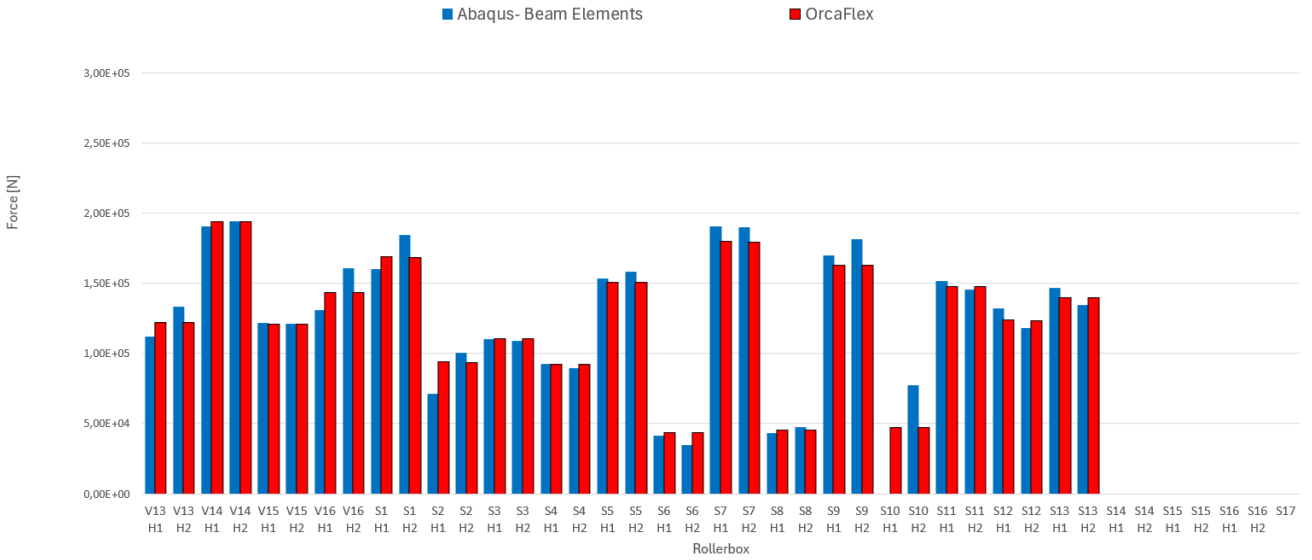
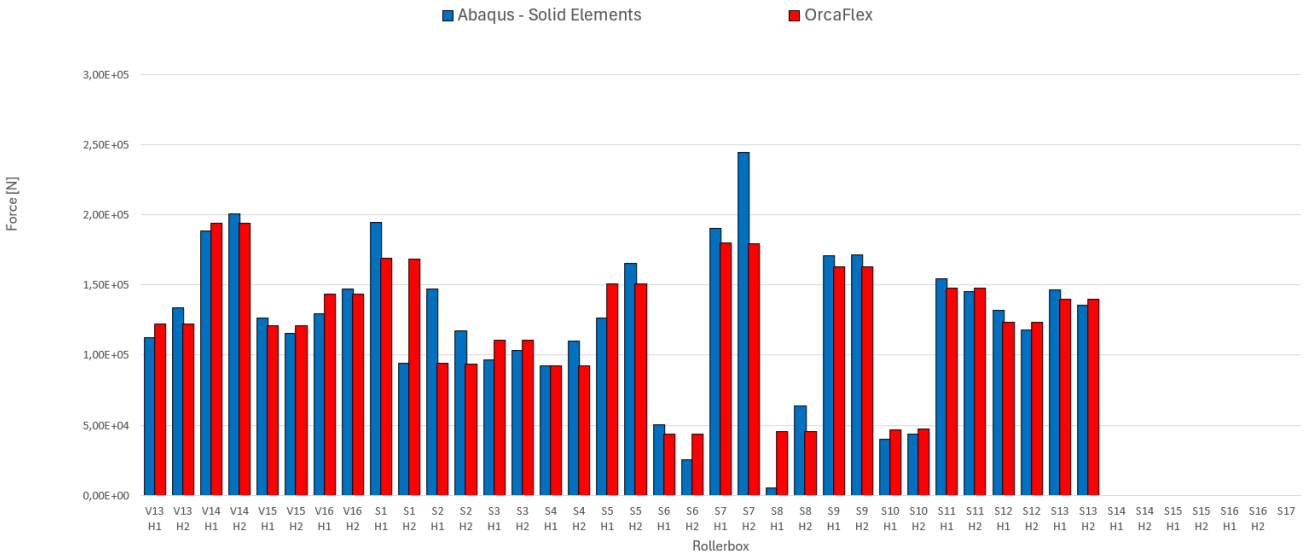


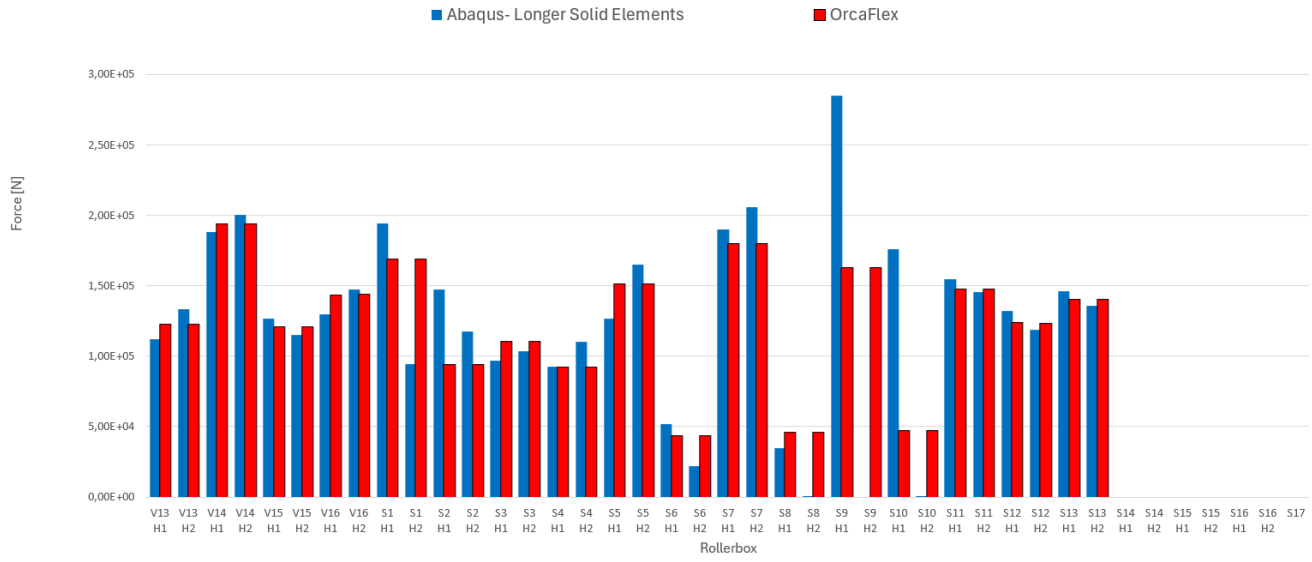
Figure B.25: Declination angle for the rollerboxes used in both Abaqus and OrcaFlex, from RB S17 (leftmost point) to RB V13 (rightmost point) - For the comparison of the homogeneous pipeline



(a)

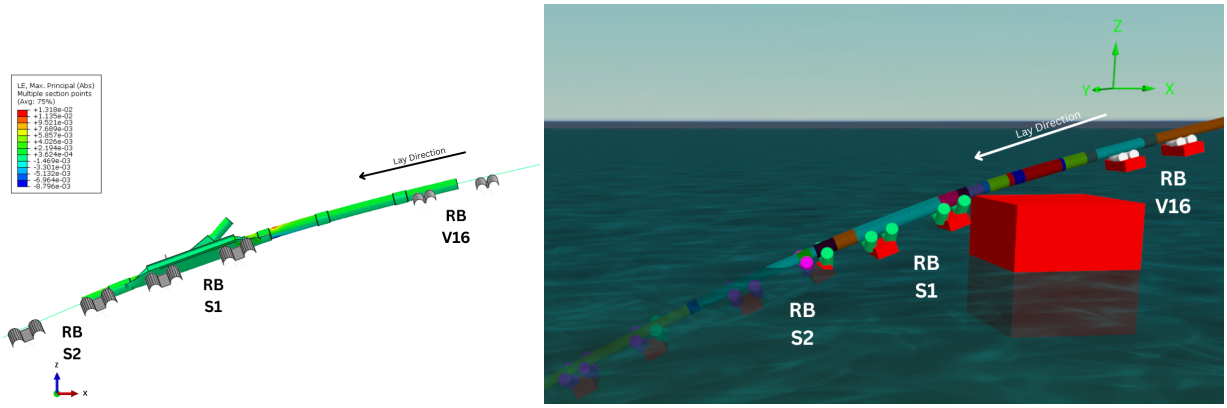


(b)



(c)

Figure B.26: Rollerbox reaction force comparison of OrcaFlex with Abaqus: a) Line-Elements b) Solid-Elements C) Longer Solid-Elements



(a) (b)

Figure B.27: Inline structure illustration for the Max RB reaction force case: a) Abaqus b) OrcaFlex

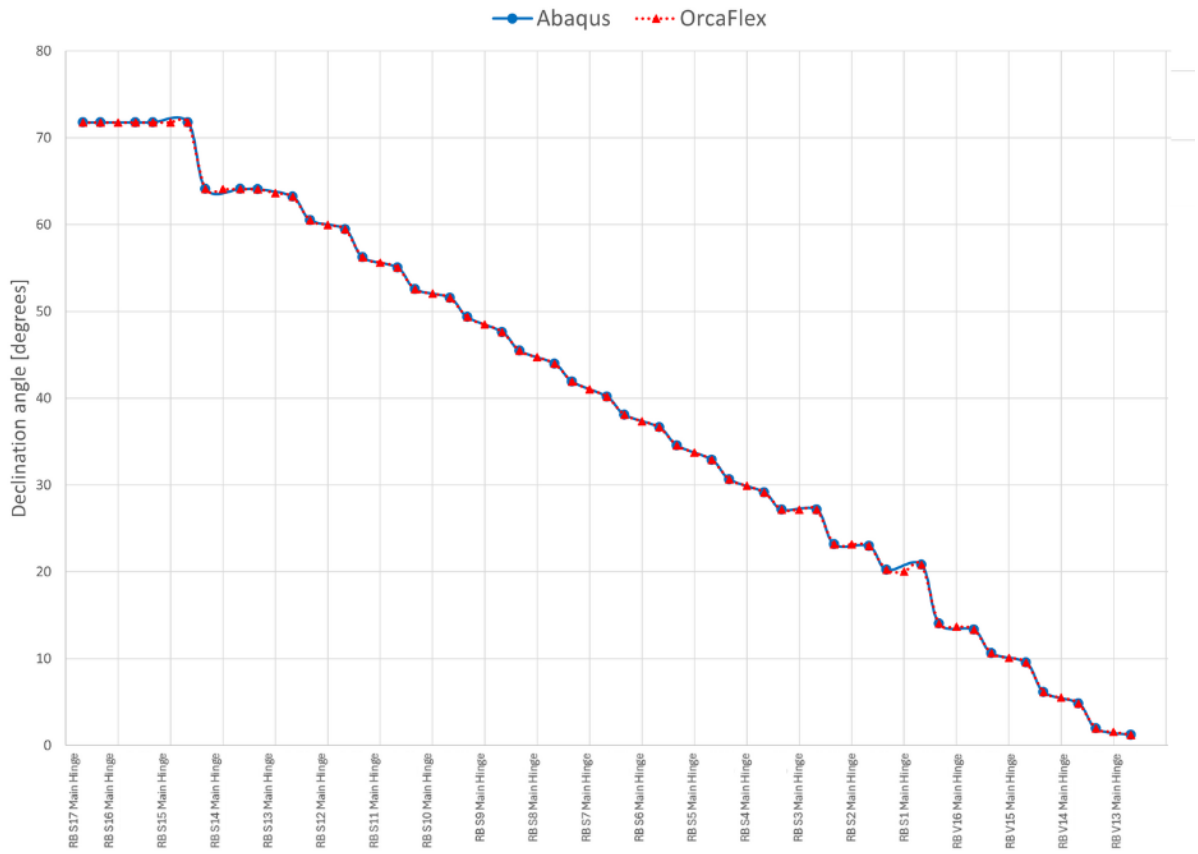


Figure B.28: Declination angle for the rollerboxes used in both Abaqus and OrcaFlex, from RB S17 (leftmost point) to RB V13 (rightmost point) - For the Max RB reaction forces case comparison

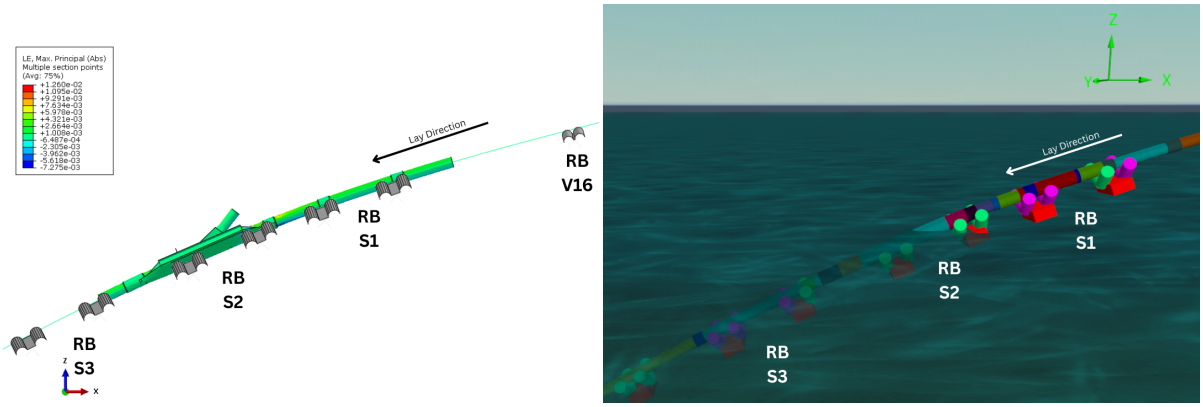


Figure B.29: Inline structure illustration for the Max RB reaction force case: a) Abaqus b) OrcaFlex

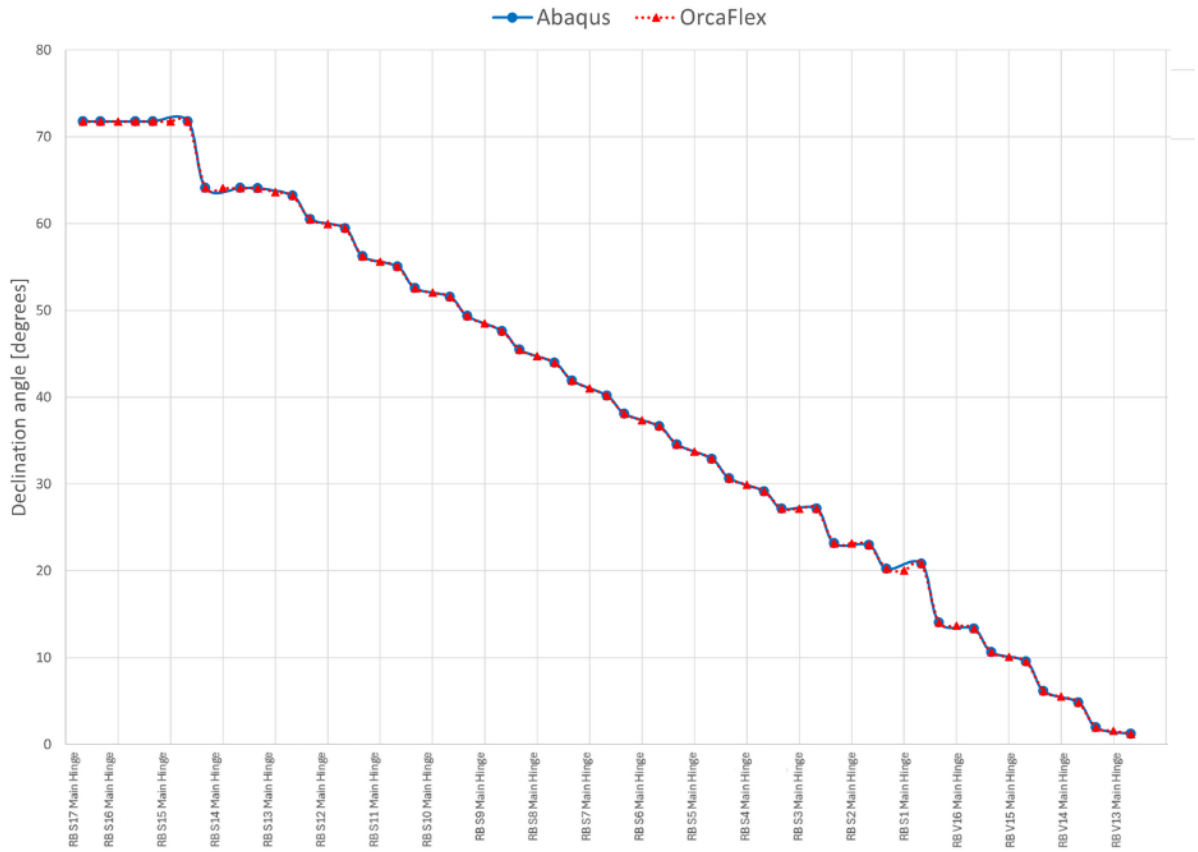


Figure B.30: Declination angle for the rollerboxes used in both Abaqus and OrcaFlex, from RB S17 (leftmost point) to RB V13 (rightmost point) - For the Max bending angle difference case comparison

B.2. Tables

General comparison between Abaqus and OrcaFlex for static simulations

Object	Abaqus	OrcaFlex
Vessel	Not modelled.	Modelled, only used as attachment points for rollers supports, vessel rollerboxes, the stinger, and one end of the pipeline.
Rollerboxes		
Vessel rollerbox body	Modelled as one rigid body per rollerbox, with respective dimension and declination angle.	Modelled as 8 rollers, lying in pairs in a V-shape configuration with V angle of 150 degrees (between rollers). The rollers are fixed on top of two separate flat supports, where each support has 2 roller pairs.
Stinger rollerbox body	Modelled as one rigid body per rollerbox, with respective dimension and a pre-fixed declination angle.	Most stinger rollerboxes have 8 rollers, lying in pairs in a V-shape configuration with V angle of 120 degrees (between rollers). The rollers are fixed on top of a flat support, where some rollerboxes have 2 flat supports with 2 roller pairs, and some have 1 flat support with all 4 roller pairs.
Degree of Freedom	The rollerboxes are modelled as on rigid body, and are completely pre-fixed. There coordinates and declination angles are set and fixed during the creation of the model. The coordinates and declination angle is set via a reference point in the middle of each rollerbox body, where all corresponding roller pairs modelled in a straight line. All roller pairs will thus have the same declination angle and will not translate nor displace during analyses. During modelling for some previous project, the rollerbox which caused the highest contact stress on the structure has been modified to allow just rotation in the plane of bending in order to alleviate these contact stresses. To do so, a 1D rotational spring is added with an arbitrary spring stiffness with a rotational freedom in the plane of bending.	All rollerboxes have rotational freedom in the plane of bending. The amount of rotational points (hinge) depends on how the behaviour of a rollerbox is in reality. Most rollerboxes are modelled with two supports, where these rollerboxes have 3 hinges. The main hinge in the middle of the supports allow for these to translate synchronously. Additionally, each support will have its own hinge, allowing each support to rotate individually. The rollerboxes modelled with only one support will have just a main hinge, placed in the middle of this support.
Individual rollers	Modelled within the rigid body of the rollerboxes.	Modelled by means of cylindrical supports in a V-shape configuration.
Pipeline-Rollerbox Contact Behaviour	A Contact Interaction is created to represent frictionless sliding: Tangential behaviour is set to frictionless, as the roller in reality spin freely and friction is negligible; Normal behaviour is modelled as a nonlinear, as this allows for better contact approximation when penetration occurs.	The rollers have a Normal and Shear linear stiffness of 70 MN/m. This normal stiffness is the reaction force per unit depth of penetration. Despite the value of the shear stiffness, there is no friction coefficient defined, disabling normal friction (sliding of the pipeline in-plane) and shear friction (sliding of the pipeline along the support).
Environment		

Sea	Not modelled, but buoyancy effects are taken into account by reducing the density of a material assigned to corresponding parts. Usually parts below RB S2 are given a different density, accounting for buoyancy. This depends on where the waterline will be.	Modelled with a constant water density of $\rho_{water} = 1025[kg/m^3]$.
Seabed	Not modelled. The end of the pipeline is lowered a certain distance, equivalent to the depth of the waterline to the seabed.	Modelled as a flat rigid body, using elastic seabed model.
Gravity	Modelled as a gravity load, applied on the entire model with a gravitational constant of $g = 9,80665[m/s^2]$. Gravity is applied increasingly over four steps, in order to get better convergence.	Modelled using gravitational acceleration of $g = 9,80665[m/s^2]$.
Pipeline		
Pipe sections	Constructed using non-linear finite elements, namely line elements and/or solid elements.	Constructed using a lumped-mass model.
End points attachments	Fixed at the start of the pipeline (vessel side) and free at the end of the pipeline in order to apply axial force (bottom tension).	Both ends are fixed, at the front of the vessel and on the seabed.
Loads	Axial force is applied in the form of bottom tension, at the end of the pipeline, from the first step of the simulation. The axial force slightly increased in subsequent steps, in order to match the specified bottom tension.	Axial force is not directly applied. This is done by either displacing one of the fixed ends of the pipeline or by adjusting the length of the pipeline accordingly. In the case of displacing an end, either the vessel itself is moved, or the anchor point on the seabed.
Results		
Strain	Logarithmic (True) strain.	"Von Mises Strain", calculated using Axial strain and Hoop strain. Axial strain is the strain caused by direct tensile strain and bending strain. Strain output is measured at 4 points for each line section (at the outer and inner diameters), and the biggest value of those 4 is chosen as output.
Declination Angle of an inline structure	This is extracted by looking at the Ultimate Rotation, using the initial placement as reference.	Measured at both the nodes and between nodes, using the positive vertical axis as reference.

Table B.1: General comparison between Abaqus and OrcaFlex, for static simulation of a pipeline installation

Results of the general comparison between Abaqus and OrcaFlex for static simulations

	Difference	Solution
Vessel	Outside of the scope.	-
Rollerboxes		
Vessel rollerbox body	Having the rollerboxes as one rigid body does not allow for respective rollerboxes to have separate moving parts, in order to adjust to the pipeline.	Each vessel rollerbox has two parts, with each containing 4 rollers (2 roller pairs). The rigid body can be split into two parts, or one part can be copied in order to get two parts per rollerbox. To these, the respective hinge can be set, or the declination has to be set to a specific angle.
Stinger rollerbox body	Having the rollerboxes as one rigid body does not allow for respective rollerboxes to have separate moving side in order to adjust to the pipeline.	Most stinger rollerbox has two parts, with each containing 4 rollers (2 roller pairs). The rigid body can be split into two parts, or one part can be copied in order to get two parts per rollerbox. To these, the respective hinge can be set, or the declination has to be set to a specific angle. Other stinger rollerboxes have are already closely modelled by the single rigid body.
Degree of Freedom	Rollerboxes in Abaqus adapt less to the shape of the pipe by being fixes, leading to the weight of the pipeline not being distributed evenly over the rollerbox. This will then lead to higher reaction forces, and higher contact stress.	The rigid bodies can be set to only rotate in the plane of bending. This will let them adapt a bit more to the pipeline. Additionally, the body of the Abaqus roller can be changed in order to allow extra rotational freedom.
Individual rollers	Outside of the scope.	-
Pipeline-Rollerbox Contact Behaviour	Both are contact behaviour assume no friction.	-
Environment		
Sea	Outside of the scope.	-
Seabed	Outside of the scope.	-
Gravity	Outside of the scope.	-
Pipeline		

Pipe sections	Modelling a pipeline using solid elements yields results that are closest to reality, especially when compared to line elements. Solid elements accurately capture the full geometry of the pipe. They also provide detailed stress and strain distributions across the entire pipe section. In contrast, lumped-mass model simplifies the pipe section to a centerline, with cross-sectional properties at the nodes which are interconnected by massless springs. These provide a single result per cross section and are best suited for global structural behaviour like bending and axial forces, as their simplification significantly reduces its convergence time.	-
End points attachments	There is no difference between Abaqus and OrcaFlex at the start of the pipeline. The end of the pipeline is essentially the same in both Abaqus and OrcaFlex, yet differs in the way it is applied. The end of the pipeline in Abaqus translates in horizontal direction, due to the axial force applied to it. The end is also moved during the simulation in vertical direction towards the seabed. The end of the pipeline is fixed in OrcaFlex, and therefore automatically generates an axial force within the pipeline. The ends differ, yet both methods can cause for there to be a specific tension at the start of the pipeline.	-
Loads	The method in which tension throughout the pipeline is obtained is different, yet the method itself does not cause for a difference in value. The bottom tension in Abaqus is applied by applying the value extracted from OrcaFlex.	-
Analysis Results		
Strain	Abaqus captures logarithmic strain, including not only tensile, bending, and hoop strain as OrcaFlex does, but it also captures torsional and shear strain. Torsion usually excluded, so strain due to shear should be the biggest contributor to the differences.	By having a close representation of the structure in OrcaFlex, the differences in strain can be reduced.
Declination Angle of an inline structure	The way in which the declination angles is measure differs.	This can easily be corrected by subtracting 90 degrees from OrcaFlex.

Table B.2: Results of the general comparison between Abaqus and OrcaFlex, for static simulation of a pipeline installation

In-depth comparison of the pipeline in Abaqus and OrcaFlex for static simulations

	Abaqus	**OrcaFlex**
Material	Density, Young's Modulus, Poisson's Ratio, and Plastic behaviour curve data.	Mass per unit length, Bending Axial and Torsional stiffness, Poisson's ratio. The bending stiffness is usually nonlinear, and given by a Bend Moment – Curvature graph.
Pipe sections	Constructed with non-linear finite elements. Pipe sections are modelled either using solid or line elements. Pipe sections modelled with line elements are those lying in sections where not much detail is not needed, e.g. sagbend section.	Constructed by means of a lumped-mass model, having corresponding contact Outer & Inner diameter, Stress Outer & Inner diameter, and other material properties at the nodes, and axial and torsional stiffnesses at the segments.
Inline structures	Constructed using 3D non-linear finite elements, using a combination of shell and solid elements. Corresponding component interactions and constraints are added to approximate reality. Some components are simplified by adding an isotropic point mass at the corresponding locations.	Constructed using the lumped-mass model, where the structure behaviour is approximated by changing pipe section properties at respective locations: Bending Stiffness, Axial Stiffness, Torsional Stiffness, Stress Outer & Inner diameter, Contact Outer diameter, Mass per Unit Length.
Length	The length of the pipeline modelled is arbitrary, having enough length to pass through all rollerboxes and sag down onto the seabed. The pipeline end is pulled by the tension applied, and subsequently lowered towards the seabed. There is no given position where the end of the pipeline will end, thus having it long enough allows the pipeline to get pulled, sag, and have the correct curvature.	The length of the pipeline modelled is arbitrary once the following requirements are met: 1. The start of the pipeline is fixed to the vessel, and the pipeline end is fixed on the seabed. 2. The pipeline has to pass over the vessel rollerboxes, stinger rollerboxes, sagbend section, and have at least one pipe-section length that is lying down on the seabed. Additionally, the length of the pipeline might also be decreased/increased in order to add/remove tension in the pipeline, respectively.
Pipeline ends		
Pipeline start point (vessel side)	The start of the pipeline in completely fix in all 6 degrees of freedom, and no direct load is applied to it.	The start of the pipeline in connected to the vessel with infinite Bending and Twisting stiffnesses, and no direct load is applied to it.

Pipeline end point (seabed side)	The end of the pipeline is free in only axial direction, and has an axial load applied to it in order to have tension in the pipeline.	The end of the pipeline is connected to the seabed at an arbitrary point with infinite Bending and Twisting stiffnesses, and no direct load is applied to it.
Loads		
Tension	The corresponding tension to the case analysed is added by applying an axial load at the end of the pipeline. The pipeline is pre-tensioned in the first step. The subsequent steps then place the pipeline on the stinger in order to perform the static analysis.	Tension in the pipeline can applied by doing either or all of the following accordingly: 1. Increase/decrease the length of the pipeline. 2. Move the vessel closer to/away from the pipeline end on the seabed. 3. Move the pipeline end closer to/away from the vessel.
Torsion	Pipeline ends are constrained, and additional boundary conditions are placed throughout the pipeline in order to avoid twisting.	Pipeline ends have infinite twisting stiffness and the “include torsion” setting is turned off in the settings for the entire model.
Rollerbox reaction forces	Rollerbox reaction forces are not extracted.	Rollerbox reaction forces are extracted at the Main Hinge of each rollerbox. The magnitude of the rollerbox reaction force at the Main Hinge is an addition of the reaction force at all roller sets.

Table B.3: In-depth comparison between a pipeline in Abaqus and OrcaFlex, for static simulation of a pipeline installation

Results of the in-depth comparison of the pipeline in Abaqus and OrcaFlex for static simulations

	Difference	**Solution**
Material	Outside of the scope.	-
Pipe sections	Majority of the pipeline constructed using the non-linear finite is created by means of line elements in Abaqus, making it mostly identical to the lumped-mass model in OrcaFlex. We assume no/negligible differences.	-
Inline structures	Modelling a structure using the lumped-mass model does not take into account the complete behaviour of components, especially complex components. The behaviour of components are approximated by taking the moment of inertia at a cross section for each component. Multiplying this by the Young's modulus gives a bending stiffness for each pipe section. Additional to this, other properties are entered for each pipe section in order to approximate its behaviour.	By adding multiple pipe sections for each component, the inline structure can be approximated.
Length	Outside of the scope.	-
Pipeline ends		
Pipeline start point (vessel side)	No differences.	-
Pipeline end point (seabed side)	The end of the pipeline is essentially the same in both Abaqus and OrcaFlex, yet differs in the way it is applied. The end of the pipeline in Abaqus translates in horizontal direction, due to a horizontal force applied to it. The end is also moved during the simulation in vertical direction towards the seabed. The end of the pipeline is fixed in OrcaFlex. The ends differ, yet both methods can cause for there to be a specific tension at the start of the pipeline.	-
Loads		

Tension	The way in which the tension is applied wont be of influence on the value of tension throughout the pipeline, as the bottom tension applied to Abaqus is an extracted value from OrcaFlex. However, the tension through the pipeline and the top tension can be slightly different since the pipe sections and structures are modelled differently.	-
Torsion	OrcaFlex fully excludes torsion. Abaqus excludes for the most part, constraining the pipeline at certain sections. Regardless, torsion is assumed to be negligible for both Abaqus and OrcaFlex.	-
Rollerbox reaction forces	For both Abaqus and OrcaFlex the loads are measured at same position, but by the rollerboxes themselves behaving differently, the rollerbox reaction force magnitudes will be different.	By changing the behaviour of the rollerboxes themselves, the differences in magnitude can be reduced.

Table B.4: Results of the in-depth comparison between a pipeline in Abaqus and OrcaFlex, for static simulation of a pipeline installation

In-depth comparison of the inline structure in Abaqus and OrcaFlex for static simulations

	Abaqus	**OrcaFlex**
Material	Density, Young's Modulus, Poisson's Ratio, and Plastic behaviour curve data.	Mass per unit length, Bending Axial and Torsional stiffness, Poisson's ratio. Bending stiffness is given by a Bend Moment – Curvature graph.
Components of the inline structure	Constructed using 3D non-linear finite elements, using a combination of shell and solid elements. Corresponding component interactions and constraints are added to approximate reality. Some components are simplified by adding an isotropic point mass at the corresponding locations.	Constructed using the lumped-mass model, where the structure behaviour is approximated by changing pipe section properties at respective locations: Bending Stiffness, Axial Stiffness, Torsional Stiffness, Stress Outer & Inner diameter, Contact Outer diameter, Mass per Unit Length.
Roller contact	Structures always have a rollerguide at the bottom of it in order to avoid the different components coming into direct contact with the rollers, where the contact forces can be very high. The rollerguide also allows for the structure to pass smoothly over the rollers. When a structure is passing on a certain roller, the contact will then be with the rollerguide, which has a V shape, just like the roller pairs. The contact points then lays along the long of the rollers.	Structures in OrcaFlex are constructed using the lumped-mass model, the same way pipe sections are modelled. Therefore, the roller contact with the structure will be the same as the roller contact with a structure. The only difference is that a structure might be represented with pipe sections with increasing diameter. By having a change in diameter, the contact point on the roller will shift as the diameter increases.
Loads		
Torsion	Pipeline ends are constrained, and additional boundary conditions are placed throughout the pipeline in order to avoid twisting.	Pipeline ends have infinite twisting stiffness and the "include torsion" setting is turned off in the settings for the entire model.
Rollerbox reaction forces	Rollerbox reaction forces are extracted from one point and not from each roller.	Rollerbox reaction forces are extracted at the Main Hinge of each rollerbox. The magnitude of the rollerbox reaction force at the Main Hinge is an addition of the reaction force at all roller sets.

Table B.5: In-depth comparison between an inline structure in Abaqus and OrcaFlex, for static simulation of a pipeline installation

Results of the in-depth comparison of an inline structure in Abaqus and OrcaFlex for static simulations

	Difference	**Solution**
Material	Outside of the scope	-
Components of the inline structure	Modelling a structure only using pipe sections does not take into account the complete behaviour of components, especially complex components. The behaviour of components are approximated by taking the moment of inertia at a cross section for each component. Multiplying this by the Young's modulus gives a bending stiffness for each pipe section. Additional to this, other properties are entered for each pipe section in order to approximate its behaviour. The extent of how well these are approximated remains uncertain.	
Roller contact	Outside of the scope	-
Loads		
Torsion	OrcaFlex fully excludes torsion. Abaqus excludes for the most part, constraining the pipeline at certain sections. Regardless, torsion is assumed to be negligible for both Abaqus and OrcaFlex.	-
Rollerbox reaction forces	Since the rollerbox reaction forces are not extracted from individual rollers, the roller contacts will not lead to a difference in force magnitude. Yet, due to the fact that the rollerboxes themselves behave different in Abaqus and in OrcaFlex, the reaction force magnitudes will be different.	-

Table B.6: Results of the in-depth comparison between an inline structure in Abaqus and OrcaFlex, for static simulation of a pipeline installation

Variable	Value	Unit
Bottom Tension	600	$[kN]$
Gravity	0	$[m/s^2]$
Pipeline End Displacement	0	$[m]$

Table B.7: Values of the variables used in the Abaqus model for pipeline comparison, for the pre-tension step

Variable	Value	Unit
Bottom Tension	600	$[kN]$
Gravity	$2,4517$	$[m/s^2]$
Pipeline End Displacement	334	$[m]$

Table B.8: Values of the variables used in the Abaqus model for pipeline comparison, for the first contact step

Variable	Value	Unit
Bottom Tension	900	$[kN]$
Gravity	$4,9033$	$[m/s^2]$
Pipeline End Displacement	668	$[m]$

Table B.9: Values of the variables used in the Abaqus model for pipeline comparison, for the second contact step

Variable	Value	Unit
Bottom Tension	900	$[kN]$
Gravity	$7,3550$	$[m/s^2]$
Pipeline End Displacement	1002	$[m]$

Table B.10: Values of the variables used in the Abaqus model for pipeline comparison, for the thirds contact step

Variable	Value	Unit
Bottom Tension	1200	$[kN]$
Gravity	$9,80665$	$[m/s^2]$
Pipeline End Displacement	1336	$[m]$

Table B.11: Values of the variables used in the Abaqus model for pipeline comparison, for the final step

Relative distance L for each rollerbox

Rollerbox Type	L $[m]$	Rollerboxes
RB T1	1,475	RB V13, V14, V15, V16
RB T2	1,8	RB S1, S2, S3, S4, S5, S6, S7, S8, S9
RB T3	1,2	RB S10, S11, S15, S16
RB T4	1,15	RB S12, S13, S14

Table B.12: Relative distance from Rollerbox Main Hinge to each Hinge, given per rollerbox

Pipeline Comparison

OrcaFlex Declination Angle	
Rollerbox	**Declination angle [$degrees$]**
MainHingeRB-V13 Out-frame declination	1,54
MainHingeRB-V14 Out-frame declination	5,51
MainHingeRB-V15 Out-frame declination	9,96
MainHingeRB-V16 Out-frame declination	14,22
MainHingeRB-S1 Out-frame declination	19,26
MainHingeRB-S2 Out-frame declination	22,51
MainHingeRB-S3 Out-frame declination	26,25
MainHingeRB-S4 Out-frame declination	29,88
MainHingeRB-S5 Out-frame declination	33,73
MainHingeRB-S6 Out-frame declination	37,35
MainHingeRB-S7 Out-frame declination	41,02
MainHingeRB-S8 Out-frame declination	44,71
MainHingeRB-S9 Out-frame declination	48,47
MainHingeRB-S10 Out-frame declination	52,05
MainHingeRB-S11 Out-frame declination	55,61
MainHingeRB-S12 Out-frame declination	59,96
MainHingeRB-S13 Out-frame declination	63,63
MainHingeRB-S14 Out-frame declination	64,08
MainHingeRB-S15 Out-frame declination	71,76
MainHingeRB-S16 Out-frame declination	71,76
MainHingeRB-S17 Out-frame declination	71,76

Table B.13: Pipeline comparison results: OrcaFlex declination angles for Hinge 1 and Hinge 2

OrcaFlex Declination Angle	
Rollerbox	**Declination angle [$degrees$]**
Hinge1rollerConstr-V13 Out-frame declination	1,17
Hinge2rollerConstr-V13 Out-frame declination	1,94
Hinge1rollerConstr-V14 Out-frame declination	4,91
Hinge2rollerConstr-V14 Out-frame declination	6,11
Hinge1rollerConstr-V15 Out-frame declination	9,43
Hinge2rollerConstr-V15 Out-frame declination	10,5
Hinge1rollerConstr-V16 Out-frame declination	13,63
Hinge2rollerConstr-V16 Out-frame declination	14,82
Hinge1rollerConstr-S1 Out-frame declination	18,39
Hinge2rollerConstr-S1 Out-frame declination	20,13
Hinge1rollerConstr-S2 Out-frame declination	21,66
Hinge2rollerConstr-S2 Out-frame declination	23,33
Hinge1rollerConstr-S3 Out-frame declination	25,4
Hinge2rollerConstr-S3 Out-frame declination	27,04
Hinge1rollerConstr-S4 Out-frame declination	29,09
Hinge2rollerConstr-S4 Out-frame declination	30,65

Hinge1rollerConstr-S5 Out-frame declination	32,89
Hinge2rollerConstr-S5 Out-frame declination	34,56
Hinge1rollerConstr-S6 Out-frame declination	36,61
Hinge2rollerConstr-S6 Out-frame declination	38,08
Hinge1rollerConstr-S7 Out-frame declination	40,13
Hinge2rollerConstr-S7 Out-frame declination	41,90
Hinge1rollerConstr-S8 Out-frame declination	43,96
Hinge2rollerConstr-S8 Out-frame declination	45,45
Hinge1rollerConstr-S9 Out-frame declination	47,61
Hinge2rollerConstr-S9 Out-frame declination	49,31
Hinge1rollerConstr-S10 Out-frame declination	51,55
Hinge2rollerConstr-S10 Out-frame declination	52,54
Hinge1rollerConstr-S11 Out-frame declination	55,04
Hinge2rollerConstr-S11 Out-frame declination	56,19
Hinge1rollerConstr-S12 Out-frame declination	59,44
Hinge2rollerConstr-S12 Out-frame declination	60,48
Hinge1rollerConstr-S13 Out-frame declination	63,19
Hinge2rollerConstr-S13 Out-frame declination	64,05
Hinge1rollerConstr-S14 Out-frame declination	64,08
Hinge2rollerConstr-S14 Out-frame declination	64,08
Hinge1rollerConstr-S15 Out-frame declination	71,76
Hinge2rollerConstr-S15 Out-frame declination	71,76
Hinge1rollerConstr-S16 Out-frame declination	71,76
Hinge2rollerConstr-S16 Out-frame declination	71,76
MainHingeRB-S17 Out-frame declination	71,76

Table B.14: Pipeline comparison results: OrcaFlex declination angles for Hinge 1 and Hinge 2

ILS Max RB Reaction Force Comparison

OrcaFlex Declination Angle

Rollerbox	**Declination angle [$degrees$]**
MainHingeRB-V13 Out-frame declination	1,55
MainHingeRB-V14 Out-frame declination	5,49
MainHingeRB-V15 Out-frame declination	10,08
MainHingeRB-V16 Out-frame declination	13,69
MainHingeRB-S1 Out-frame declination	20,04
MainHingeRB-S2 Out-frame declination	23,16
MainHingeRB-S3 Out-frame declination	27,17
MainHingeRB-S4 Out-frame declination	29,88
MainHingeRB-S5 Out-frame declination	33,72
MainHingeRB-S6 Out-frame declination	37,35
MainHingeRB-S7 Out-frame declination	41,02
MainHingeRB-S8 Out-frame declination	44,71

MainHingeRB-S9 Out-frame declination	48,47
MainHingeRB-S10 Out-frame declination	52,05
MainHingeRB-S11 Out-frame declination	55,62
MainHingeRB-S12 Out-frame declination	59,96
MainHingeRB-S13 Out-frame declination	63,64
MainHingeRB-S14 Out-frame declination	64,08
MainHingeRB-S15 Out-frame declination	71,76
MainHingeRB-S16 Out-frame declination	71,76
MainHingeRB-S17 Out-frame declination	71,76

Table B.15: ILS Max RB reaction force results: OrcaFlex declination angles for Hinge 1 and Hinge 2

OrcaFlex Declination Angle	
Rollerbox	**Declination angle [$degrees$]**
Hinge1rollerConstr-V13 Out-frame declination	1,20
Hinge2rollerConstr-V13 Out-frame declination	1,94
Hinge1rollerConstr-V14 Out-frame declination	4,81
Hinge2rollerConstr-V14 Out-frame declination	6,14
Hinge1rollerConstr-V15 Out-frame declination	9,52
Hinge2rollerConstr-V15 Out-frame declination	10,62
Hinge1rollerConstr-V16 Out-frame declination	13,33
Hinge2rollerConstr-V16 Out-frame declination	14,03
Hinge1rollerConstr-S1 Out-frame declination	20,81
Hinge2rollerConstr-S1 Out-frame declination	20,22
Hinge1rollerConstr-S2 Out-frame declination	22,96
Hinge2rollerConstr-S2 Out-frame declination	23,15
Hinge1rollerConstr-S3 Out-frame declination	27,14
Hinge2rollerConstr-S3 Out-frame declination	27,16
Hinge1rollerConstr-S4 Out-frame declination	29,14
Hinge2rollerConstr-S4 Out-frame declination	30,65
Hinge1rollerConstr-S5 Out-frame declination	32,89
Hinge2rollerConstr-S5 Out-frame declination	34,56
Hinge1rollerConstr-S6 Out-frame declination	36,61
Hinge2rollerConstr-S6 Out-frame declination	38,09
Hinge1rollerConstr-S7 Out-frame declination	40,15
Hinge2rollerConstr-S7 Out-frame declination	41,90
Hinge1rollerConstr-S8 Out-frame declination	43,96
Hinge2rollerConstr-S8 Out-frame declination	45,45
Hinge1rollerConstr-S9 Out-frame declination	47,61
Hinge2rollerConstr-S9 Out-frame declination	49,32
Hinge1rollerConstr-S10 Out-frame declination	51,55
Hinge2rollerConstr-S10 Out-frame declination	52,54

Hinge1rollerConstr-S11 Out-frame declination	55,04
Hinge2rollerConstr-S11 Out-frame declination	56,19
Hinge1rollerConstr-S12 Out-frame declination	59,44
Hinge2rollerConstr-S12 Out-frame declination	60,48
Hinge1rollerConstr-S13 Out-frame declination	63,19
Hinge2rollerConstr-S13 Out-frame declination	64,05
Hinge1rollerConstr-S14 Out-frame declination	64,08
Hinge2rollerConstr-S14 Out-frame declination	64,08
Hinge1rollerConstr-S15 Out-frame declination	71,76
Hinge2rollerConstr-S15 Out-frame declination	71,76
Hinge1rollerConstr-S16 Out-frame declination	71,76
Hinge2rollerConstr-S16 Out-frame declination	71,76
MainHingeRB-S17 Out-frame declination	71,76

Table B.16: ILS Max RB reaction force results: OrcaFlex declination angles for Hinge 1 and Hinge 2

ILS Max Bending Angle Difference Comparison

OrcaFlex Declination Angle	
Rollerbox	**Declination angle [$degrees$]**
MainHingeRB-V13 Out-frame declination	1,54
MainHingeRB-V14 Out-frame declination	5,47
MainHingeRB-V15 Out-frame declination	9,99
MainHingeRB-V16 Out-frame declination	14,21
MainHingeRB-S1 Out-frame declination	19,23
MainHingeRB-S2 Out-frame declination	24,09
MainHingeRB-S3 Out-frame declination	26,75
MainHingeRB-S4 Out-frame declination	30,07
MainHingeRB-S5 Out-frame declination	33,66
MainHingeRB-S6 Out-frame declination	37,37
MainHingeRB-S7 Out-frame declination	41,01
MainHingeRB-S8 Out-frame declination	44,71
MainHingeRB-S9 Out-frame declination	48,47
MainHingeRB-S10 Out-frame declination	52,05
MainHingeRB-S11 Out-frame declination	55,62
MainHingeRB-S12 Out-frame declination	59,96
MainHingeRB-S13 Out-frame declination	63,64
MainHingeRB-S14 Out-frame declination	64,08
MainHingeRB-S15 Out-frame declination	71,76
MainHingeRB-S16 Out-frame declination	71,76
MainHingeRB-S17 Out-frame declination	71,76

Table B.17: ILS Max bending angle difference results: OrcaFlex declination angles for Hinge 1 and Hinge 2

OrcaFlex Declination Angle

Rollerbox	Declination angle [$degrees$]
Hinge1rollerConstr-V13 Out-frame declination	1,20
Hinge2rollerConstr-V13 Out-frame declination	1,93
Hinge1rollerConstr-V14 Out-frame declination	4,83
Hinge2rollerConstr-V14 Out-frame declination	6,17
Hinge1rollerConstr-V15 Out-frame declination	9,43
Hinge2rollerConstr-V15 Out-frame declination	10,52
Hinge1rollerConstr-V16 Out-frame declination	13,66
Hinge2rollerConstr-V16 Out-frame declination	14,78
Hinge1rollerConstr-S1 Out-frame declination	19,91
Hinge2rollerConstr-S1 Out-frame declination	19,23
Hinge1rollerConstr-S2 Out-frame declination	24,47
Hinge2rollerConstr-S2 Out-frame declination	23,87
Hinge1rollerConstr-S3 Out-frame declination	26,71
Hinge2rollerConstr-S3 Out-frame declination	26,75
Hinge1rollerConstr-S4 Out-frame declination	29,42
Hinge2rollerConstr-S4 Out-frame declination	30,72
Hinge1rollerConstr-S5 Out-frame declination	32,79
Hinge2rollerConstr-S5 Out-frame declination	34,53
Hinge1rollerConstr-S6 Out-frame declination	36,64
Hinge2rollerConstr-S6 Out-frame declination	38,09
Hinge1rollerConstr-S7 Out-frame declination	40,14
Hinge2rollerConstr-S7 Out-frame declination	41,89
Hinge1rollerConstr-S8 Out-frame declination	43,97
Hinge2rollerConstr-S8 Out-frame declination	45,45
Hinge1rollerConstr-S9 Out-frame declination	47,61
Hinge2rollerConstr-S9 Out-frame declination	49,32
Hinge1rollerConstr-S10 Out-frame declination	51,55
Hinge2rollerConstr-S10 Out-frame declination	52,54
Hinge1rollerConstr-S11 Out-frame declination	55,03
Hinge2rollerConstr-S11 Out-frame declination	56,21
Hinge1rollerConstr-S12 Out-frame declination	59,43
Hinge2rollerConstr-S12 Out-frame declination	60,50
Hinge1rollerConstr-S13 Out-frame declination	63,19
Hinge2rollerConstr-S13 Out-frame declination	64,06
Hinge1rollerConstr-S14 Out-frame declination	64,08
Hinge2rollerConstr-S14 Out-frame declination	64,08

Hinge1rollerConstr-S15 Out-frame declination	71,76
Hinge2rollerConstr-S15 Out-frame declination	71,76
Hinge1rollerConstr-S16 Out-frame declination	71,76
Hinge2rollerConstr-S16 Out-frame declination	71,76
MainHingeRB-S17 Out-frame declination	71,76

Table B.18: ILS Max bending angle difference results: OrcaFlex declination angles for Hinge 1 and Hinge 2

C

Coding

C.1. Inline Structure Displacement Code

```
"""
Inline Structure (ILS) positions in OrcaFlex:
This code will displace the ILS with an increment over a certain length (overbend section), and will run a static analysis at every position.
"""

import matplotlib.pyplot as plt
import numpy as np
import OrcFxAPI
import os
import pandas as pd
from pathlib import Path

def payout(sectionlength,totalmovement,pipebeforeily,pipeafterily,adjacentpipes,):
    increments = totalmovement / sectionlength #meters to integer
    nfiles = int(increments)
    print(nfiles, " files in total will be created.")

    ilyangle = pd.DataFrame()
    ilycurvature = pd.DataFrame()
    ilystrain = pd.DataFrame()
    rbforces = pd.DataFrame(index = range(21))

    for i in range(nfiles+1):
        print("Currently at step ", i)
        folder = Path(r"D:\IvanDDrive\WorkItem 3")
        file = "410100_SOL_ILS_Detailed_RB_ILS_StartRB1.yml"

        # folder = Path(r"C:\Soft_dev\projects\ivan_study\Phase 3 - ILT models movement\ILY Movement - Python results")
        # file = "410100_SOL_ILS_Detailed_RB_AbaqusComparison_Step " + i + ".dat"

        filepath = folder / file

        model = OrcFxAPI.Model(filepath)
        pipe = model["Pipe"]
        pipe.Length[2] =  pipebeforeily + sectionlength * i
        pipe.Length[-1] = pipeafterily - sectionlength * i
        print(pipe.Length[2])
        ilylength = 23.617
        ilystart = pipe.CumulativeLength[4]
        ilyend = pipe.CumulativeLength[19]
        adjacentpipesbefore = ilystart - adjacentpipes    #Checking pipe sections before ILY - distance in meters
        adjacentpipesafter = ilyend + adjacentpipes       #Checking pipe sections after ILY - distance in meters
        FromArclength = ilystart
```

```
        ToArclength = ilyend
        print("ILY currently starts at ", ilystart, " and ends at ", ilyend)

        rbv13 = model["MainHingeRB-V13"]
        rbv14 = model["MainHingeRB-V14"]
        rbv15 = model["MainHingeRB-V15"]
        rbv16 = model["MainHingeRB-V16"]
        rb1 = model["MainHingeRB-S1"]
        rb2 = model["MainHingeRB-S2"]
        rb3 = model["MainHingeRB-S3"]
        rb4 = model["MainHingeRB-S4"]
        rb5 = model["MainHingeRB-S5"]
        rb6 = model["MainHingeRB-S6"]
        rb7 = model["MainHingeRB-S7"]
        rb8 = model["MainHingeRB-S8"]
        rb9 = model["MainHingeRB-S9"]
        rb10 = model["MainHingeRB-S10"]
        rb11 = model["MainHingeRB-S11"]
        rb12 = model["MainHingeRB-S12"]
        rb13 = model["MainHingeRB-S13"]
        rb14 = model["MainHingeRB-S14"]
        rb15 = model["MainHingeRB-S15"]
        rb16 = model["MainHingeRB-S16"]
        rb17 = model["MainHingeRB-S17"]
        pipe.FullStaticsMaxIterations = 1800

        try:
            model.CalculateStatics()
            length = OrcFxAPI.arSpecifiedArclengths(FromArclength, ToArclength)
            crv = pipe.RangeGraph("Curvature", arclengthRange=length)
            VMStr = pipe.RangeGraph("Max pipelay von Mises strain", arclengthRange=length)
            angle = pipe.RangeGraph("Node declination", arclengthRange=length)

            rbv13RF = rbv13.TimeHistory("In-frame connection force")
            rbv14RF = rbv14.TimeHistory("In-frame connection force")
            rbv15RF = rbv15.TimeHistory("In-frame connection force")
            rbv16RF = rbv16.TimeHistory("In-frame connection force")
            rb1RF = rb1.TimeHistory("In-frame connection force")
            rb2RF = rb2.TimeHistory("In-frame connection force")
            rb3RF = rb3.TimeHistory("In-frame connection force")
            rb4RF = rb4.TimeHistory("In-frame connection force")
            rb5RF = rb5.TimeHistory("In-frame connection force")
            rb6RF = rb6.TimeHistory("In-frame connection force")
            rb7RF = rb7.TimeHistory("In-frame connection force")
            rb8RF = rb8.TimeHistory("In-frame connection force")
            rb9RF = rb9.TimeHistory("In-frame connection force")
            rb10RF = rb10.TimeHistory("In-frame connection force")
            rb11RF = rb11.TimeHistory("In-frame connection force")
            rb12RF = rb12.TimeHistory("In-frame connection force")
            rb13RF = rb13.TimeHistory("In-frame connection force")
            rb14RF = rb14.TimeHistory("In-frame connection force")
            rb15RF = rb15.TimeHistory("In-frame connection force")
            rb16RF = rb16.TimeHistory("In-frame connection force")
            rb17RF = rb17.TimeHistory("In-frame connection force")

            anglearray = [angle.X[0], angle.Mean[0], angle.X[-1], angle.Mean[-1]]
            rbarray = [rbv13RF[0],rbv14RF[0],rbv15RF[0],rbv16RF[0],rb1RF[0],rb2RF[0],rb3RF
                [0],rb4RF[0],rb5RF[0],rb6RF[0],rb7RF[0],rb8RF[0],rb9RF[0],rb10RF[0],rb11RF
                [0],rb12RF[0],rb13RF[0],rb14RF[0],rb15RF[0],rb16RF[0],rb17RF[0]]
            ilyangle[i] = anglearray
            rbforces[i] = rbarray
            ilycurvature[i] = crv.Mean
            ilystrain[i] = VMStr.Mean
            print(ilyangle)

            extrafolder = "ILY Movement - Python results"
            filenumber = "Step %s" % i
            filename_new = "410100_SOL_ILS_Detailed_RB_ILS_StartRB1_" + filenumber + ".sim"
            filepath_new = folder / extrafolder / filename_new
            model.SaveData(filepath_new)
        except Exception as e:
```

```
            print(e)                #prints error
            print("error line 109")
            extrafolder = "ILY Movement - Python results"
            filenumber = "Step %d" %i
            filename_new = "410100_SOL_ILS_Detailed_RB_ILS_StartRB1_" + filenumber + "_FAILED
                .yml"
            filepath_new = folder / extrafolder / filename_new
            model.SaveData(filepath_new)
            try:
                ilycurvature[i] = np.zeros(len(ilycurvature[0]))
                ilystrain[i] = np.zeros(len(ilystrain[0]))
                rbforces[i] = np.zeros(len(rbforces[0]))
                ilyangle[i] = np.zeros(4)

            except Exception as e2:
                print(e2)
                print("error line 122")

        if i % 100 == 0:
            print("Excel Exported")
            ilyangle.to_excel('ILY Angle at step %d.xlsx' %i)
            ilycurvature.to_excel('ILY Curvature at step %d.xlsx' %i)
            ilystrain.to_excel('ILY Strain at step %d.xlsx' %i)
            rbforces.to_excel('ILY Forces at step %d.xlsx' %i)
    ilyangle.to_excel('ILY Angle at step %d.xlsx' % i)
    ilycurvature.to_excel('ILY Curvature at step %d.xlsx' % i)
    ilystrain.to_excel('ILY Strain at step %d.xlsx' % i)
    rbforces.to_excel('ILY Forces at step %d.xlsx' % i)
    return

payout(0.1,40,23.5,3006.43,10)

#pipebeforeILY = No. 2
#pipeafterILY = Last pipe section
```

C.2. Homogenous Pipeline Code

```
"""
Rollerboxes Configuration Creation code for Abaqus:
Copy this code and paste it into the Abaqus Script prompt. Note: the Abaqus Assembly should be empty.
"""
from abaqus import *
from part import *
from material import *
from section import *
from assembly import *
from step import *
from interaction import *
from load import *
from mesh import *
from optimization import *
from job import *
from sketch import *
from visualization import *
from connectorBehavior import *
Model_naam = 'TSJ16-AdjustedRBSpring-ModifiedRB-SmallerRB'
xt=[ 76.431E+03, 66.327E+03, 51.831E+03, 42746.47,39797.53,  30161.19, 27224.81,
    17967.73,15062.27,6307.79, 3448.21, -5004.79,-8403.21 ,-11340.05, -14665.95
    ,-19054.74,-22283.26,-26499.21,-29620.79,-33879,-36873,-40775.11,-43636.89,-47185.89,
-49902.11,-53245.74,-55804.26,-59235.47,-61622.53,-64909.03,-66384.97,-69792.34,-71147.66,
-75003.39,-76154.61,-79321.23,-80342.77,-83015.32,-84020.68,
-86374.4,-87125.6, -88892.4,
-89643.6,-91095]
zt=[ 10.092E+03, 10.092E+03, 10.092E+03,10189.69,10110.31,
9550.61,9267.39,7997.37,7486.63,5618.39,4893.61,2256.86,1069.14,
-53.1,-1430.9,-3421.66, -5014.34,-7263.38, -9056.62,-11692.5,-13691.5,-16486.96,
-18671.04,-21552.67,-23915.33,-27029.71,-29562.29,-33178.6,
    -35873.4,-39855.74,-41748.26,-46396.66,-48377.34,-54508.42,-56499.58,
-62421.66,-64482.34,-70640.68, -72709.32,-79382.3,-81661.7, -88145.3,
-90424.7,-97875]
rt=[ 0, 0, 0,-1.16998054112309,-1.93596193634564,-4.91255012440583, -6.1109893139231
    3,-9.43489353600559,
-10.5045020539608,-13.6356471157755, -14.815119684111,-18.4089054514001,-20.1369414051209
    ,-21.6649295241129,
-23.3284699533887,-25.4642206320804, -27.482658200914, -29.0996055457527, -30.6559450245856,
-32.8927759385977, -34.5623145781752,-36.6172619682684,-38.0887196178748,-40.1342200431248,
    -41.9025542439479 ,
-43.9608117337857 ,-45.4508564589443,-47.6198586688912, -49.3109442670672,-51.5547921701978,
-52.5443426182443,-55.0420495867304, -56.1938517649266, -59.4461805754187, -60.480760801986,
-63.195453196066, -64.0555573219264,-64.0800012507035,
    -64.0800014843804,-71.76,-71.76,-71.76,-71.76,-71.76]
Tag0 = [1, 2, 3] #Support 1,2,3
Tag1 = [4, 5, 6, 7, 8, 9, 10 ,11] #RB V13-16
Tag2 = [12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29]
Tag3 = [30, 31, 32, 33, 40, 41, 42, 43]  #RB10,11,15,16
Tag2222 = [34, 35, 36, 37, 38, 39] #RB12-14
Tag4 = [44]            #RB S17
naam = ["␣", "␣", "␣", "V13H1", "V13H2", "V14H1", "V14H2", "V15H1", "V15H2", "V16H1", "V16H2"
    , "S1H1", "S1H2", "S2H1", "S2H2", "S3H1", "S3H2", "S4H1", "S4H2", "S5H1", "S5H2", "S6H1",
     "S6H2", "S7H1", "S7H2", "S8H1", "S8H2", "S9H1", "S9H2", "S10H1", "S10H2", "S11H1", "
    S11H2", "S12H1", "S12H2", "S13H1", "S13H2", "S14H1", "S14H2", "S15H1", "S15H2", "S16H1",
    "S16H2", "S17"]
p = 0
t = len(xt)
for i in range(0, t):
    p = p + 1
    naam1 = "RB-" + str(p)
    if p in Tag0:
        Rolbx = 'Solitaire_Roller_T0'
        Rolbxx = 'Solitaire_Roller_T0' + '-' + str(p)
        vect = [1.119912814E+03,640.997084,974.98]
        axdir1 = [0.0, 100.0, 0.0]
        axdir2 = [-800.0, 0.0, 0.0]
        axdir3 = [0.0, 1000.0, 0.0]
        axP1 = [-1.119912814E+03,-640.997084,-974.98]
```

```
        axP2 = [-1.119912814E+03,-640.997084,-974.98]
        axP3 = [0,0,0]
    elif p in Tag1:
        Rolbx = 'Solitaire_Roller_T1'
        Rolbxx = 'Solitaire_Roller_' + naam[p-1]
        vect = [1.095359295E+03, 628.449835, 1.585E+03]
        axdir1 = [0.0, 100.0, 0.0]
        axdir2 = [-800.0, 0.0, 0.0]
        axdir3 = [0.0, 1000.0, 0.0]
        axP1 = [-1.095359295E+03, -628.449835, -1.585E+03]
        axP2 = [-1.095359295E+03, -628.449835, -1.585E+03]
        axP3 = [0,0,0]
    elif p in Tag2:
        Rolbx = 'Solitaire_Roller_T2'
        Rolbxx = 'Solitaire_Roller_' + naam[p-1]
        vect = [1.115662814E+03,638.5283,1.585E+03]
        axdir1 = [0.0, 100.0, 0.0]
        axdir2 = [-1200.0, 0.0, 0.0]
        axdir3 = [0.0, 1000.0, 0.0]
        axP1 = [-1.115662814E+03,-638.5283,-1.585E+03]
        axP2 = [-1.115662814E+03,-638.5283,-1.585E+03]
        axP3 = [0,0,0]
    elif p in Tag2222:
        Rolbx = 'Solitaire_Roller_T2222'
        Rolbxx = 'Solitaire_Roller_' + naam[p-1]
        vect = [1.115662814E+03,638.5283,1.585E+03]
        axdir1 = [0.0, 100.0, 0.0]
        axdir2 = [-1200.0, 0.0, 0.0]
        axdir3 = [0.0, 1000.0, 0.0]
        axP1 = [-1.115662814E+03,-638.5283,-1.585E+03]
        axP2 = [-1.115662814E+03,-638.5283,-1.585E+03]
        axP3 = [0,0,0]
    elif p in Tag3:
        Rolbx = 'Solitaire_Roller_T3'
        Rolbxx = 'Solitaire_Roller_' + naam[p-1]
        vect = [1.115662814E+03,638.540098,1.585E+03]
        axdir1 = [0.0, 100.0, 0.0]
        axdir2 = [-10.0, 0.0, 0.0]
        axdir3 = [0.0, 10.0, 0.0]
        axP1 = [-1.115662814E+03,-638.540098,-1.585E+03]
        axP2 = [-1.115662814E+03,-638.540098,-1.585E+03]
        axP3 = [0,0,0]
    elif p in Tag4:
        Rolbx = 'Solitaire_Roller_T4'
        Rolbxx = 'Solitaire_Roller_' + naam[p-1]
        vect = [1.115662814E+03,638.5283,2.0475E+03]
        axdir1 = [0.0, 100.0, 0.0]
        axdir2 = [-921.5, 0.0, 0.0]
        axdir3 = [0.0, 1000.0, 0.0]
        axP1 = [-1.115662814E+03,-638.5283,-2.0475E+03]
        axP2 = [-1.115662814E+03,-638.5283,-2.0475E+03]
        axP3 = [0,0,0]
    mdb.models[Model_naam].rootAssembly.Instance(dependent=ON, name=Rolbxx, part=mdb.models[
        Model_naam].parts[Rolbx])
    mdb.models[Model_naam].rootAssembly.rotate(angle=-90.0, axisDirection=axdir1, \
    axisPoint=axP1, instanceList=(Rolbxx,))
    mdb.models[Model_naam].rootAssembly.rotate(angle=-90.0, axisDirection=axdir2, \
    axisPoint=axP2, instanceList=(Rolbxx,))
    mdb.models[Model_naam].rootAssembly.translate(instanceList=(Rolbxx,), vector=vect)
    mdb.models[Model_naam].rootAssembly.rotate(angle=rt[i], axisDirection=axdir3, \
    axisPoint=axP3, instanceList=(Rolbxx,))
    mdb.models[Model_naam].rootAssembly.translate(instanceList=(Rolbxx,), vector=(xt[i], 0.0,
         zt[i]))
    # mdb.models[Model_naam].rootAssembly.features.changeKey(fromName=Rolbxx, toName=naam1)
    mdb.models[Model_naam].rootAssembly.Surface(name=Rolbxx, objectToCopy=mdb.models[
        Model_naam]. \
    rootAssembly.instances[Rolbxx].surfaces['Top_face'])
```

C.3. Max Rollerbox Reaction Force Case Code

```
"""
Rollerboxes Configuration Creation code for Abaqus:
Copy this code and paste it into the Abaqus Script prompt. Note: the Abaqus Assembly should be empty.
"""
from abaqus import *
from part import *
from material import *
from section import *
from assembly import *
from step import *
from interaction import *
from load import *
from mesh import *
from optimization import *
from job import *
from sketch import *
from visualization import *
from connectorBehavior import *
Model_naam = 'ILY-Shenandoah-MaxReactionForce'

xt=[ 76.431E+03, 66.327E+03, 51.831E+03, 42746.46,39797.54,
30161.23,27224.77,17967.24,15062.76,6311.12,
3444.88,-5012.97,-8395.03,-11348.01,-14657.99,-19067.57,
-22270.43,-26499.34,-29620.66,-33878.91,-36873.09,
-40775.11,-43636.89,-47186.01,-49901.99,-53245.73,-55804.27,
-59235.49,-61622.51,-64909.01,-66384.99,-69792.44,-71147.56,
-75003.31,-76154.69,-79321.31,-80342.69,-83015.32,
-84020.68,-86374.4,-87125.6,-88892.4,-89643.6,-91095]

zt=[ 10.092E+03, 10.092E+03, 10.092E+03,10190.02,10109.98,9550.12,9267.88,8000.11,
7483.89,5604.98,4907.02,2279.77,1046.23,-34.18,-1449.82,
-3396.16,-5039.84,-7263.14,-9056.86,-11692.64,-13691.36,
-16486.96,-18671.04,-21552.53,-23915.47,-27029.72,
-29562.28,-33178.58,-35873.42,-39855.76,-41748.24,-46396.59,
-48377.41,-54508.47,-56499.53,-62421.61,-64482.39,
-70640.68,-72709.32,-79382.3,-81661.7,-88145.3,-90424.7,-97875]

rt=[ 0, 0, 0,-1.2040854930637,-1.93766238829725,-4.81919923276897,
-6.14218073522243,-9.52520853792959,-10.6238084820871,
-13.3312187730581,-14.0344890861685,-20.8149405735342,
-20.2185772800955,-22.9554161222402,-23.1558833134404,-27.1412342882751,
-27.1664268344417,-29.1355817106357,-30.6494586446819,-32.8904930723457,
-34.5539517056782,-36.6185720201438,-38.863527365023,-40.1454419861892,
-41.9040995996431,-43.9638614386199,-45.4505827104793,-47.6102010870534,
-49.3219214119989,-51.5495948403256,-52.546868896098,-55.0303443795269,
-56.2151009108018,-59.4329667594659,-60.4964552640119,-63.1883614299141,
-64.0548476740009,-64.08,-64.08,-71.76,-71.76,
-71.76,-71.76,-71.76]

Tag0 = [1, 2, 3] #Support 1,2,3
Tag1 = [4, 5, 6, 7, 8, 9, 10 ,11] #RB V13-16
Tag2 = [12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29]
Tag3 = [30, 31, 32, 33, 40, 41, 42, 43]  #RB10,11,15,16
Tag2222 = [34, 35, 36, 37, 38, 39] #RB12-14
Tag4 = [44]           #RB S17

p = 0
T0 = 0
T1 = 0
T2 = 0
T2222 = 0
T3 = 0
T4 = 0
t = len(xt)
for i in range(0, t):
    p = p + 1
    naam1 = "RB-" + str(p)
    if p in Tag0:
```

```
        T0 = T0 + 1
        Rolbx = 'RB-T0'
        Rolbxx = 'RB-T0' + '-' + str(T0)
        vect = [1.119912814E+03,640.997084,974.98]
        axdir1 = [0.0, 100.0, 0.0]
        axdir2 = [-800.0, 0.0, 0.0]
        axdir3 = [0.0, 1000.0, 0.0]
        axP1 = [-1.119912814E+03,-640.997084,-974.98]
        axP2 = [-1.119912814E+03,-640.997084,-974.98]
        axP3 = [0,0,0]
        naam1 = Rolbxx
    elif p in Tag1:
        T1 = T1 + 1
        Rolbx = 'RB-T1'
        Rolbxx = 'RB-T1' + '-' + str(T1)
        vect = [1.095359295E+03, 628.449835, 1.585E+03]
        axdir1 = [0.0, 100.0, 0.0]
        axdir2 = [-800.0, 0.0, 0.0]
        axdir3 = [0.0, 1000.0, 0.0]
        axP1 = [-1.095359295E+03, -628.449835, -1.585E+03]
        axP2 = [-1.095359295E+03, -628.449835, -1.585E+03]
        axP3 = [0,0,0]
        naam1 = Rolbxx
    elif p in Tag2:
        T2 = T2 + 1
        Rolbx = 'RB-T2'
        Rolbxx = 'RB-T2' + '-' + str(T2)
        vect = [1.115662814E+03,638.5283,1.585E+03]
        axdir1 = [0.0, 100.0, 0.0]
        axdir2 = [-1200.0, 0.0, 0.0]
        axdir3 = [0.0, 1000.0, 0.0]
        axP1 = [-1.115662814E+03,-638.5283,-1.585E+03]
        axP2 = [-1.115662814E+03,-638.5283,-1.585E+03]
        axP3 = [0,0,0]
        naam1 = Rolbxx
    elif p in Tag2222:
        T2222 = T2222 + 1
        Rolbx = 'T2222'
        Rolbxx = 'T2222' + '-' + str(T2222)
        vect = [1.115662814E+03,638.5283,1.585E+03]
        axdir1 = [0.0, 100.0, 0.0]
        axdir2 = [-1200.0, 0.0, 0.0]
        axdir3 = [0.0, 1000.0, 0.0]
        axP1 = [-1.115662814E+03,-638.5283,-1.585E+03]
        axP2 = [-1.115662814E+03,-638.5283,-1.585E+03]
        axP3 = [0,0,0]
        naam1 = Rolbxx
    elif p in Tag3:
        T3 = T3 + 1
        Rolbx = 'RB-T3'
        Rolbxx = 'RB-T3' + '-' + str(T3)
        vect = [1.115662814E+03,638.540098,1.585E+03]
        axdir1 = [0.0, 100.0, 0.0]
        axdir2 = [-10.0, 0.0, 0.0]
        axdir3 = [0.0, 10.0, 0.0]
        axP1 = [-1.115662814E+03,-638.540098,-1.585E+03]
        axP2 = [-1.115662814E+03,-638.540098,-1.585E+03]
        axP3 = [0,0,0]
        naam1 = Rolbxx
    elif p in Tag4:
        T4 = T4 + 1
        Rolbx = 'RB-T4'
        Rolbxx = 'RB-T4' + '-' + str(T4)
        vect = [1.115662814E+03,638.5283,2.0475E+03]
        axdir1 = [0.0, 100.0, 0.0]
        axdir2 = [-921.5, 0.0, 0.0]
        axdir3 = [0.0, 1000.0, 0.0]
        axP1 = [-1.115662814E+03,-638.5283,-2.0475E+03]
        axP2 = [-1.115662814E+03,-638.5283,-2.0475E+03]
        axP3 = [0,0,0]
        naam1 = Rolbxx
```

```
140     mdb.models[Model_naam].rootAssembly.Instance(dependent=ON, name=Rolbxx, part=mdb.models[
            Model_naam].parts[Rolbx])
141     mdb.models[Model_naam].rootAssembly.rotate(angle=-90.0, axisDirection=axdir1, \
142     axisPoint=axP1, instanceList=(Rolbxx,))
143     mdb.models[Model_naam].rootAssembly.rotate(angle=-90.0, axisDirection=axdir2, \
144     axisPoint=axP2, instanceList=(Rolbxx,))
145     mdb.models[Model_naam].rootAssembly.translate(instanceList=(Rolbxx,), vector=vect)
146     mdb.models[Model_naam].rootAssembly.rotate(angle=rt[i], axisDirection=axdir3, \
147     axisPoint=axP3, instanceList=(Rolbxx,))
148     mdb.models[Model_naam].rootAssembly.translate(instanceList=(Rolbxx,), vector=(xt[i], 0.0,
            zt[i]))
149     mdb.models[Model_naam].rootAssembly.features.changeKey(fromName=Rolbxx, toName=naam1)
150     mdb.models[Model_naam].rootAssembly.Surface(name=naam1, objectToCopy=mdb.models[
            Model_naam]. \
151     rootAssembly.instances[naam1].surfaces['Topface'])
```

C.4. Max Declination Angle Difference Case Code

```
"""
Rollerboxes Configuration Creation code for Abaqus:
Copy this code and paste it into the Abaqus Script prompt. Note: the Abaqus Assembly should be empty.
"""
from abaqus import *
from part import *
from material import *
from section import *
from assembly import *
from step import *
from interaction import *
from load import *
from mesh import *
from optimization import *
from job import *
from sketch import *
from visualization import *
from connectorBehavior import *
Model_naam = 'ILY-Shenandoah-MaxBendingILY'

xt=[ 76.431E+03, 66.327E+03, 51.831E+03,42746.46,39797.54,30161.28,27224.72,
17967.65,15062.35,6307.86,3448.14,-5004.43,-8403.57,-11359.72,-14646.28,-19061.66,
-22276.34,-26502.33,-29617.67,-33877.79,-36874.21,-40775.48,
-43636.52,-47185.83,-49902.17,-53245.78,-55804.22,-59235.53,-61622.47,
-64909.02,-66384.98,-69792.44,-71147.56,-75003.32,-76154.68,
-79321.31,-80342.69,-83015.32,-84020.68,-86374.4,-87125.6,-88892.4,
-89643.6,-91095]

zt=[ 10.092E+03, 10.092E+03, 10.092E+03,10189.86,10110.14,9549.67,9268.33
,7997.8,7486.2,5618.1,4893.9,2255.84,1070.16,-7.39,-1476.61,
-3407.78,-5028.22,-7257.97,-9062.03,-11694.31,-13689.69,-16486.48,-18671.52,
-21552.74,-23915.26,-27029.67,-29562.33,-33178.55,-35873.45,-39855.75,
-41748.25,-46396.59,-48377.41,-54508.47,-56499.53,-62421.61
,-64482.39,-70640.68,-72709.32,-79382.3,
-81661.7,-88145.3,-90424.7,-97875]

rt=[ 0, 0, 0,-1.19958333340793,-1.92700117425911,-4.82902098024286,-6.16643634485618,
-9.4265827637806,-10.5237695163074,-13.6557391708668,-14.7807991619002,
-19.9138996918454,-19.2295068407183,-24.4708081198729,
-23.8733052134974,-26.7081519416912,-26.7514930826147,-29.4161398692906,-30.7281098103153,
-32.7917051004454,-34.5324985251004,-36.6442469957181,-38.0918712224335,
-40.1394472470527,-41.8934053506905,-43.9685082043966,
-45.4528359941454,-47.6135142931258,-49.3224545192047,-51.5505416710344,-52.5482355691493,
-55.0278279482867,-56.2096785315282,-59.4292402556602,-60.4967268914339,-63.1930721632913,
-64.0572541284283,-64.08,-64.08,-71.76,-71.76,-71.76,-71.76,-71.76]

Tag0 = [1, 2, 3] #Support 1,2,3
Tag1 = [4, 5, 6, 7, 8, 9, 10 ,11] #RB V13-16
Tag2 = [12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29]
Tag3 = [30, 31, 32, 33, 40, 41, 42, 43]  #RB10,11,15,16
Tag2222 = [34, 35, 36, 37, 38, 39] #RB12-14
Tag4 = [44]           #RB S17
p = 0
T0 = 0
T1 = 0
T2 = 0
T2222 = 0
T3 = 0
T4 = 0
t = len(xt)
for i in range(0, t):
    p = p + 1
    naam1 = "RB-" + str(p)
    if p in Tag0:
        T0 = T0 + 1
        Rolbx = 'RB-T0'
        Rolbxx = 'RB-T0' + '-' + str(T0)
        vect = [1.119912814E+03,640.997084,974.98]
```

```
        axdir1 = [0.0, 100.0, 0.0]
        axdir2 = [-800.0, 0.0, 0.0]
        axdir3 = [0.0, 1000.0, 0.0]
        axP1 = [-1.119912814E+03,-640.997084,-974.98]
        axP2 = [-1.119912814E+03,-640.997084,-974.98]
        axP3 = [0,0,0]
        naam1 = Rolbxx
    elif p in Tag1:
        T1 = T1 + 1
        Rolbx = 'RB-T1'
        Rolbxx = 'RB-T1' + '-' + str(T1)
        vect = [1.095359295E+03, 628.449835, 1.585E+03]
        axdir1 = [0.0, 100.0, 0.0]
        axdir2 = [-800.0, 0.0, 0.0]
        axdir3 = [0.0, 1000.0, 0.0]
        axP1 = [-1.095359295E+03, -628.449835, -1.585E+03]
        axP2 = [-1.095359295E+03, -628.449835, -1.585E+03]
        axP3 = [0,0,0]
        naam1 = Rolbxx
    elif p in Tag2:
        T2 = T2 + 1
        Rolbx = 'RB-T2'
        Rolbxx = 'RB-T2' + '-' + str(T2)
        vect = [1.115662814E+03,638.5283,1.585E+03]
        axdir1 = [0.0, 100.0, 0.0]
        axdir2 = [-1200.0, 0.0, 0.0]
        axdir3 = [0.0, 1000.0, 0.0]
        axP1 = [-1.115662814E+03,-638.5283,-1.585E+03]
        axP2 = [-1.115662814E+03,-638.5283,-1.585E+03]
        axP3 = [0,0,0]
        naam1 = Rolbxx
    elif p in Tag2222:
        T2222 = T2222 + 1
        Rolbx = 'T2222'
        Rolbxx = 'T2222' + '-' + str(T2222)
        vect = [1.115662814E+03,638.5283,1.585E+03]
        axdir1 = [0.0, 100.0, 0.0]
        axdir2 = [-1200.0, 0.0, 0.0]
        axdir3 = [0.0, 1000.0, 0.0]
        axP1 = [-1.115662814E+03,-638.5283,-1.585E+03]
        axP2 = [-1.115662814E+03,-638.5283,-1.585E+03]
        axP3 = [0,0,0]
        naam1 = Rolbxx
    elif p in Tag3:
        T3 = T3 + 1
        Rolbx = 'RB-T3'
        Rolbxx = 'RB-T3' + '-' + str(T3)
        vect = [1.115662814E+03,638.540098,1.585E+03]
        axdir1 = [0.0, 100.0, 0.0]
        axdir2 = [-10.0, 0.0, 0.0]
        axdir3 = [0.0, 10.0, 0.0]
        axP1 = [-1.115662814E+03,-638.540098,-1.585E+03]
        axP2 = [-1.115662814E+03,-638.540098,-1.585E+03]
        axP3 = [0,0,0]
        naam1 = Rolbxx
    elif p in Tag4:
        T4 = T4 + 1
        Rolbx = 'RB-T4'
        Rolbxx = 'RB-T4' + '-' + str(T4)
        vect = [1.115662814E+03,638.5283,2.0475E+03]
        axdir1 = [0.0, 100.0, 0.0]
        axdir2 = [-921.5, 0.0, 0.0]
        axdir3 = [0.0, 1000.0, 0.0]
        axP1 = [-1.115662814E+03,-638.5283,-2.0475E+03]
        axP2 = [-1.115662814E+03,-638.5283,-2.0475E+03]
        axP3 = [0,0,0]
        naam1 = Rolbxx
    mdb.models[Model_naam].rootAssembly.Instance(dependent=ON, name=Rolbxx, part=mdb.models[
        Model_naam].parts[Rolbx])
    mdb.models[Model_naam].rootAssembly.rotate(angle=-90.0, axisDirection=axdir1, \
    axisPoint=axP1, instanceList=(Rolbxx,))
```

```
139     mdb.models[Model_naam].rootAssembly.rotate(angle=-90.0, axisDirection=axdir2, \
140     axisPoint=axP2, instanceList=(Rolbxx,))
141     mdb.models[Model_naam].rootAssembly.translate(instanceList=(Rolbxx,), vector=vect)
142     mdb.models[Model_naam].rootAssembly.rotate(angle=rt[i], axisDirection=axdir3, \
143     axisPoint=axP3, instanceList=(Rolbxx,))
144     mdb.models[Model_naam].rootAssembly.translate(instanceList=(Rolbxx,), vector=(xt[i], 0.0,
            zt[i]))
145     mdb.models[Model_naam].rootAssembly.features.changeKey(fromName=Rolbxx, toName=naam1)
146     mdb.models[Model_naam].rootAssembly.Surface(name=naam1, objectToCopy=mdb.models[
            Model_naam]. \
147     rootAssembly.instances[naam1].surfaces['Topface'])
```

D

Definitions

In the list of definitions below, examples are used in order to facilitate explanation. These examples are given (but not limited to) using a pipeline.

Axial Force
This is a force along the length of the pipeline, referred to as tension, which is either pulling or compressing the pipeline. This is typically caused by the vessel either moving forward or backwards, or by the tensioner pulling or releasing the pipeline.

Bending Moment
This is a moment causing the pipeline to bend.

Bottom Tension
This is a reference to the axial force within the pipeline at a certain location. Bottom Tension refers the the axial force in the pipeline at the touchdown point on the seabed.

Curvature
This is a quantification of the bending shape, i.e the amount of curve that the pipeline has once it has experienced loads such as a bending moment - Equation 3.10.

Declination Angle
This refers to the angle which quantifies how much an object has rotated from a horizontal position, Figure 3.10. When an object lies horizontally, its declination angle is $0°$. For this research project, once an object starts rotating against the clock, its declination angle increases.

Inflection Point
This is the transition point between the overbend and sagbend section, Figure 1.1.

ILS
This is the abbreviation for Inline Structure (ILS). An inline structure is a type of pipeline structure, which lies in between (in line) pipe section along the pipeline.

Lumped-mass Pipeline
This is a pipeline which is represented by a lumped-mass model, Figure 1.8. A lumped-mass model of a pipeline is a series of masses (nodes) to which most properties, such as diameters are lumped. These masses (nodes) are interconnected by means of springs (segments) which contain properties such as axial stiffness. By means of such representation, the mathematical complexity is simplified, and analyses convergence times are significantly reduced.

Overbend Section
This is the section spanning from the tensioners, up to the inflection point in the pipeline, Figure 1.1. The overbend section includes both vessel rollerboxes and stinger rollerboxes, and can also include a section of the pipeline after the stinger. The shape of the pipeline in the overbend section is given by settings of the rollerboxes and stingers, and can not be modified mid-pipelay.

Roller
This is the cylinder that is in direct contact with the pipeline. These are always seen in pairs (roller pairs), sitting in a V-shape configuration. The angle between the rollers depend on their location on the vessel.

Roller support
This is a set of roller pairs, Figure B.1b, uplifting and guiding the pipeline towards the tensioners of the vessel. These usually have 2 sets of roller pairs (4 rollers in total), and can be found inside of the vessel.

Rollerbox
This is a set of roller pairs in V-shape formation confined within a box-like metal structure. The rollers on these rollerboxes uphold and guide the pipeline from the tensioners into the sea, Section 1.1.5. There are two types of rollerboxes: *Vessel Rollerboxes* (V RB) and *Stinger Rollerboxes* (S RB). Vessel rollerboxes are directly mounted on the vessel, and are the closest to the tensioners. Stinger rollerboxes are mounted on the stinger, and can be found following the vessel rollerboxes.

Sagbend Section
This is the section spanning from the inflection point up to the touchdown point on the seabed, Figure 1.1. The shape of the pipeline in the sagbend section is determined by the amount of tension in the pipeline, Appendix B.4.

Strain
This is a quantification of deformation. Strain tells you how much (a section of) the pipeline has stretched or compressed, compared to its original length.

Stress
This is a quantification of the (internal) forces that develop within an area, e.g. cross-section of the pipeline once subjected to external loads.

Top Tension
This is a reference to the axial force within the pipeline at a certain location. Top Tension refers the the axial force in the pipeline at the start of the pipeline, e.g on the vessel.