



# A dynamic design and analysis of the flexible hose of the deepsea mining system

A. van Bergen



# A dynamic design and analysis of the flexible hose of the deepsea mining system

by

A. van Bergen

to obtain the degree of  
Master of Science in Offshore and Dredging engineering  
at the Delft University of Technology,  
to be defended publicly on, May 10, 2022

Student number: 4601726  
Project duration: October 1, 2020 – May 4, 2022  
Thesis committee: Ir. W. Boomsma, Royal IHC, daily supervisor  
Ir. A. Blanken, Royal IHC, daily supervisor  
Dr. ir. A. Tsouvalas, TU Delft, Chair  
Prof. dr. ir. A. Metrikine, TU Delft, supervisor  
Ir. A. Tsetas, TU Delft, daily supervisor

*This thesis is confidential and cannot be made public until xxxx*

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



# Preface

First of all, I would like to thank everybody that supported and motivated me to keep going through a difficult period, partially because of the Covid pandemic lockdowns. Then I would like to start by thanking my graduation committee for their input, patience, discussions and comments on the work done; Apostolos Tsouvalas, Andrei Metrikine and Athanasios Tsetas (Thanasis).

I also would like to thank my loving parent and sister for supporting me through everything, good and bad times. Helping me to stay on track and reach for my goals. I'm grateful to my family, friends and housemates for their support and understanding. Next, I want to thank my therapist for getting me through my depression and diagnosis.

Finally, many thanks to Royal IHC for the opportunity to graduate at their company during the Covid period. In particular, Wiebe Boomsma and Amir Blanken for their support, motivation and work experience. It was a very interesting and broad topic and the proving of input and data to develop a new design for the flexible connection of the mining system.

Without all that, this report would not have been made possible. I'm looking forward to starting my first career at Royal IHC.

*A. van Bergen  
Rotterdam, May 4, 2022*



# Abstract

Climate change is a recurring topic in the news and goes side by side with the energy transition. The energy transition focuses on using clean energy generated by resources like wind and water. Due to the increasing demand for technology that focuses on generating and storing clean energy, the demand for metals grows. These metals are currently extracted from land-based locations but are also found on the bottom of the ocean. Three types of resources are found on the ocean floor: hydrothermal vents, cobalt crusts, and polymetallic nodules. To extract these resources from the ocean floor, a project called "Blue Nodules" was launched in 2020. This project addresses the challenge of creating a viable and sustainable value chain to retrieve the polymetallic nodules from the ocean floor. Royal IHC is one of the partners on this project and designed the seafloor nodule collector. The total mining system consists of a surface vessel, a vertical transport system, a flexible hose (jumper hose) and a seafloor nodule collector. In combination with a hose developing company, IHC designed the flexible hose. This design was projected on a prior design of the seafloor module collector. Since then, the seafloor module collector design has advanced, but the flexible hose design did not. Therefore the main objective of this master thesis is to design and analyse the flexible hose with its main focus on the dynamic behaviour.

The methodology for this thesis is addressed in the following manner. The old design of the flexible hose was investigated, and all relevant parameters were addressed. A static model was built to verify the parameters that modelled the hose with these parameters. The parameters of interest were: the length of the hose, the distance between crawler and vertical transport system in height and width, number of buoyancy modules, number of buoyancy modules per hose section, starting position of buoyancy nodules. This static model was used to perform a parameter study, resulting in an optimal set of parameters tested against design criteria.

The next stage was implementing the static design in a dynamic software packet called "OrcaFlex". Orcaflex is a software program for static and dynamic analysis of offshore systems. The static model is implemented and tested with this software. This resulted in a failure of the set of parameters and adjustment to the static model. After a re-do of the parameter study, a new set of parameters was extracted. This set is tested for an inline motion of the crawler compared to the vertical transport system and for a perpendicular motion. These motions expand in size until the flexible hose reaches its critical values. These tests resulted in a range around the VTS where the crawler could move in.

Results show that the crawler can move in this range for a selected set of parameters. Throughout the research, it is shown that the parameter set needs further development to complete the flexible connection design. It is recommended that some disregarded motions, parameters and environmental forces are further investigated and that the failure criteria that test the hose need to be expanded.





# Contents

<b>List of Figures</b>	<b>xi</b>
<b>List of Tables</b>	<b>xv</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Background . . . . .	1
1.1.1 History of mining . . . . .	1
1.1.2 The need for an energy transition . . . . .	1
1.1.3 Overall design Blue Nodules project . . . . .	2
1.2 Problem definition . . . . .	3
1.2.1 Objectives. . . . .	3
1.2.2 Scope of the thesis . . . . .	4
1.3 Approach . . . . .	4
1.3.1 Denomination of the components of the subsea harvesting equipment and other components . . . . .	5
<b>2 The ocean and it resources</b>	<b>7</b>
2.1 Classifying the ocean depths . . . . .	7
2.1.1 The ocean bathymetry and plate tectonics . . . . .	8
2.2 Resources in the Ocean . . . . .	9
2.2.1 Hydrothermal vents systems. . . . .	10
2.2.2 Cobalt crust . . . . .	10
2.2.3 Polymetallic nodules. . . . .	11
<b>3 Blue Nodules project characteristics</b>	<b>13</b>
3.1 Environmental parameters and location . . . . .	13
3.1.1 Soil and nodule parameters . . . . .	14
3.1.2 Metocean data . . . . .	14
3.2 Preliminary design of the Blue nodules project . . . . .	15
3.2.1 Subsea Harvesting equipment . . . . .	16
3.2.1.1 <i>Seafloor Mining Tool</i> . . . . .	16
3.2.1.2 <i>Jumper hose</i> . . . . .	16
3.2.1.3 <i>Vertical transport system, Umbilical and SWOE return</i> . . . . .	16
3.2.1.4 <i>Surface operation vessel</i> . . . . .	17
3.2.2 Initial design parameters of the subsea harvesting equipment . . . . .	17
3.2.3 Preliminary jumper hose design . . . . .	17
3.2.3.1 <i>Buoyancy modules</i> . . . . .	18
3.2.3.2 <i>Umbilical</i> . . . . .	18
3.2.3.3 <i>Final design of the Jumper Hose</i> . . . . .	18
3.2.3.4 <i>Prototypes and material</i> . . . . .	19
3.3 Follow-up of the Blue Nodules project - The outline of the thesis . . . . .	19
3.3.1 Premeditation, boundaries and requirements of the thesis research. . . . .	19
3.3.2 Delimited requirements of the design . . . . .	20
3.3.3 Parameters, dimensions and limitations . . . . .	20
3.4 Research question 1 and the related sub-questions . . . . .	21
<b>I Methods and models</b>	<b>23</b>
<b>4 Static analysis method - MatLab model</b>	<b>25</b>
4.1 Base model and description . . . . .	25
4.2 Modelling approach base model jumper hose . . . . .	26
4.2.1 Cable element. . . . .	26
4.2.1.1 <i>Cables under concentrated loads</i> . . . . .	26
4.2.2 Cables under non-uniformly distributed load . . . . .	27
4.2.2.1 <i>Cable under uniformly distributed load</i> . . . . .	27

4.2.2.2	<i>Catenary cable mathematic approach</i>	28
4.2.2.3	<i>Catenary cable</i>	29
4.2.3	Weight of the Jumper hose	30
4.2.4	Adding buoyancy to the model	32
4.3	Geometrically nonlinear system	32
4.3.1	Finite element method and geometrically nonlinear elements	32
4.3.2	Large deformations, static solution for a catenary model, axial extensible	32
4.3.2.1	<i>Linearized stiffness matrix</i>	35
4.4	Research question 2 and the related sub-question	36
<b>5</b>	<b>Mathematical software model - OrcaFlex</b>	<b>37</b>
5.1	The software to enhance the static MatLab model	37
5.1.1	Orcaflex program approach	37
5.2	Orcaflex theoretical background, objects and data	38
5.2.1	Line object and theory	38
5.2.1.1	<i>Line structural model</i>	39
5.2.1.2	<i>Line content flow</i>	39
5.2.1.3	<i>Hydrodynamic and aerodynamic loads, Drag</i>	40
5.2.1.4	<i>Hydrodynamic and aerodynamic loads, Lift</i>	40
5.2.2	Line Attachment buoyancy modules	41
5.2.2.1	<i>Attachment, Clump types</i>	41
5.2.2.2	<i>Attachment, drag chain</i>	41
5.2.2.3	<i>Attachment, flex joint</i>	41
5.2.2.4	<i>Attachment, stiffeners</i>	42
5.2.3	Constraint object	42
5.2.4	Environmental data	42
5.2.5	Sea data	42
5.2.5.1	<i>Seabed</i>	43
5.2.5.2	<i>Current</i>	43
5.2.5.3	<i>Buoyancy variation with depth</i>	44
5.3	Orcaflex theoretical background, Statics	44
5.3.1	Line Statics	45
5.3.1.1	<i>Line statics, step 1 methods</i>	45
5.3.1.2	<i>Line statics, step 2</i>	46
5.3.2	Whole system statics	46
5.4	OrcaFlex theoretical background, dynamic analysis	46
5.4.1	Dynamic approaches	46
5.4.1.1	<i>Frequency-domain analysis</i>	46
5.4.1.2	<i>Time-domain analysis</i>	46
5.4.1.3	<i>Time-domain Explicit solver</i>	47
5.4.1.4	<i>Time-domain Implicit solver</i>	48
5.4.2	Selected approach	48
5.5	Research question 4	48
<b>II</b>	<b>Results and Analysis</b>	<b>49</b>
<b>6</b>	<b>Static analysis - MatLab model</b>	<b>51</b>
6.0.0.1	<i>Calculation structure for static analysis</i>	51
6.1	The shape simulation of the jumper hose model	51
6.1.0.1	<i>Assumptions for the shape simulation model</i>	52
6.1.1	Catenary model, axial extensible	52
6.1.1.1	<i>Formulas to approach catenary shapes</i>	53
6.1.2	Adding buoyancy to the static model concentrated on 1 node	54
6.1.3	Spreading out the buoyancy in the static model	56
6.1.4	Segment size of the static model	57
6.2	Model design for parameter variation	59
6.2.1	Implementing buoyancy	59
6.2.2	Buoyancy implementation of the initial design	60
6.3	Continuation on sub research question 2 and the final static MatLab model	64

<b>7</b>	<b>Parameter study</b>	<b>67</b>
7.1	Parameters selection . . . . .	67
7.1.1	Restrictions of the design . . . . .	68
7.1.2	Failure criteria . . . . .	69
7.1.3	Variation and influence per parameter . . . . .	69
7.1.4	The ranking of the parameters . . . . .	72
7.2	Combinations of parameters . . . . .	72
7.2.1	Parameter sets . . . . .	73
7.3	Multi-criteria analysis . . . . .	74
7.3.1	The criteria for the MCA . . . . .	74
7.3.2	Weighting method . . . . .	75
7.3.3	Performance of the MCA . . . . .	75
7.4	Sensitivity study parameters . . . . .	77
7.4.1	Sensitivity of length parameter . . . . .	78
7.4.2	Sensitivity of height VTS parameter . . . . .	79
7.4.3	Sensitivity of modules parameter . . . . .	80
7.4.4	Sensitivity of distribution modules parameter . . . . .	81
7.4.5	Sensitivity of starting parameter . . . . .	82
7.5	New design parameter set for dynamic analysis . . . . .	83
7.5.1	Research question 3 and summary of the parameter study . . . . .	83
<b>8</b>	<b>Transition from Matlab to OrcaFlex</b>	<b>85</b>
8.1	Matlab Model implemented in OrcaFlex . . . . .	85
8.1.1	Objects Orcaflex model . . . . .	86
8.1.2	Results OrcaFlex model . . . . .	87
8.2	The enlargement of the design parameters . . . . .	87
8.2.1	The buoyancy modules . . . . .	88
8.2.2	The segment length . . . . .	90
8.2.3	Bending stiffness . . . . .	92
8.2.4	The hose body . . . . .	94
8.3	Design Parameters, Static analysis . . . . .	97
8.3.1	Static Analysis . . . . .	98
8.4	Research question 5 and summary of design choices . . . . .	100
<b>9</b>	<b>Dynamic analysis</b>	<b>101</b>
9.1	Static model - base . . . . .	101
9.2	Requirements and criteria . . . . .	102
9.2.1	System dynamics . . . . .	102
9.2.2	Crawler stability . . . . .	102
9.2.3	Jumper hose stability . . . . .	103
9.2.4	Location maximum tension and effective tension . . . . .	103
9.2.5	Summary specified requirements and limitations . . . . .	105
9.3	OrcaFlex simulations - In-line motion . . . . .	106
9.3.1	50 meters in x direction . . . . .	106
9.3.2	100 meters in x direction. . . . .	109
9.3.3	150 meters in x direction. . . . .	112
9.3.4	Analysis conclusion . . . . .	116
9.4	OrcaFlex simulations – Perpendicular motion. . . . .	117
9.4.1	Y displacement at 50 meters. . . . .	118
9.4.1.1	<i>Y displacement at 50 meters, limit range</i> . . . . .	120
9.4.2	Y displacement at 100 meters . . . . .	121
9.4.2.1	<i>Y displacement at 100 meters, limit range</i> . . . . .	123
9.4.3	Y displacement at 150 meters . . . . .	124
9.4.3.1	<i>Y displacement at 150 meters, limit range</i> . . . . .	126
9.4.4	Analysis conclusion . . . . .	127
9.5	Range Crawler and research question 6 and 7 . . . . .	127
<b>10</b>	<b>Conclusion</b>	<b>129</b>
10.1	Conclusion to the research question . . . . .	130

<b>11 Recommendations and Review</b>	<b>131</b>
11.1 Review of the parameter study . . . . .	131
11.2 Heave motion on clump weight . . . . .	132
11.2.1 Heave motion and crawler motion . . . . .	133
11.2.2 Heave motion and crawler motion, perpendicular. . . . .	133
11.3 Pattern crawler for collection nodules . . . . .	134
11.3.1 Crawler speed. . . . .	135
11.4 Displacement of clump weight . . . . .	136
11.5 Displacement of clump weight and crawler . . . . .	137
11.6 Summarized conclusions of investigated parameters . . . . .	139
11.6.1 Other disregarded effects . . . . .	139
11.7 Research question 6 and 7 . . . . .	140
<b>III Appendix</b>	<b>141</b>
<b>A Matlab iteration plots</b>	<b>143</b>
A.1 Matlab iteration steps, catenary . . . . .	143
A.2 Matlab iteration steps, catenary 1 node buoyant . . . . .	145
A.3 Matlab iteration steps, S-shape . . . . .	148
A.4 Matlab element variation plots . . . . .	151
<b>B Jumper hose data</b>	<b>153</b>
<b>C MatLab code</b>	<b>155</b>
C.1 Base model Matlab Static solution. . . . .	155
C.2 Matlab code parameter study . . . . .	164
<b>D Theoretical background OrcaFlex</b>	<b>177</b>
D.1 Calculation stages . . . . .	177
D.1.1 Tension force . . . . .	177
D.1.1.1 Damping coefficient . . . . .	178
D.1.2 Bend moments . . . . .	178
D.1.3 Shear Forces . . . . .	179
D.1.4 Torsion moments . . . . .	179
D.1.4.1 Damping coefficient . . . . .	179
D.2 Content flow types . . . . .	180
D.2.1 Content flow effects . . . . .	180
D.3 Morison equation . . . . .	181
<b>E OrcaFlex models</b>	<b>183</b>
E.1 Catenary check . . . . .	183
E.2 Old models . . . . .	183
E.3 Matlab check-up . . . . .	183
E.4 Parameter update check-up . . . . .	183
E.5 The dynamic study, x motion. . . . .	184
E.6 The dynamic study, y-motion. . . . .	184
E.7 VTS motions . . . . .	185
E.8 Range study zigzag pattern crawler and VTS motion . . . . .	185

# List of Figures

1.1	Schematic overview of the main components of the system [28]	3
2.1	Hypsographic curve of the surface of the Earth.[11]	7
2.2	Oceanic division Pelagic and Benthic zone Earth.[34]	8
2.3	Diverging plate tectonics and convergent plate tectonics[32]	9
2.4	Hydrothermal vent or black smoker.[25]	10
2.5	Cross section ridge system.[5]	10
2.6	Picture of the cobalt-rich ferromanganese crust in the northern pacific ocean [13]	11
2.7	Cross section seamount formation cobalt-rich ferromanganese crust [8]	11
2.8	Nodule field in the Pacific ocean [26]	11
2.9	Formation environment for manganese nodules [7]	11
3.1	Clarion Clipperton zone	13
3.2	Concept design, Apollo 1, Apollo 2, and full scale	16
3.3	Preliminary design subsea harvesting equipment	16
3.4	Schematic concept overview coupling hose segment	17
3.5	buoyant hose segment with 8 buoyancy nodules connected	18
3.6	Sketch hose segment, with a buoyancy module	19
4.1	Schematic drawing subsea harvesting system	25
4.2	Overview example cable under concentrated load	26
4.3	Part of the overview example cable under concentrated load	27
4.4	Overview example cable under non-uniform load	27
4.5	Section of the overview example cable under non-uniform load	27
4.6	Force vectors segment of cable under non-uniform load	27
4.7	Overview example cable under uniform load	28
4.8	Section of the overview example cable under uniform load	28
4.9	Force vectors segment of cable under non-uniform load	28
4.10	Mathematical approach catenary shape	28
4.11	Cable evaluations	29
4.12	Overview example catenary	29
4.13	Section of the overview example catenary	29
4.14	Force vectors segment of catenary	30
4.15	Equilibrium of an infinitesimal element of the inextensible cable under gravitational load [27]	30
4.16	Catenary hose in air, filled with slurry or water	31
4.17	Catenary hose in water, filled with slurry or water	31
4.18	Discretizing of the system by dividing into smaller elements [14]	33
4.19	A reasonable shape [14]	33
4.20	Balancing of forces in 2 nodes [14]	34
4.21	Implementing local stiffness element in global stiffness system [14]	36
4.22	Implementing local stiffness element in global stiffness system for all elements [14]	36
5.1	Line object model OrcaFlex, discretised model [21]	39
5.2	Line object model OrcaFlex, structural model [21]	39
5.3	Overview environment in OrcaFlex [20]	42
5.4	Overview calculation stages OrcaFlex [22]	44
5.5	Overview time-domain analysis [20]	47
6.1	Sections and segments	53
6.2	1st iteration catenary system	54
6.3	5th iteration catenary system	54

6.4	10th iteration catenary system	54
6.5	Initial parabola guess vs iterated catenary shape	54
6.6	1st iteration 1 node buoyant	55
6.7	5th iteration 1 node buoyant	55
6.8	10th iteration 1 node buoyant	55
6.9	Equilibrium 1 node buoyant	55
6.10	Buoyancy distribution over a section	56
6.11	1st iteration	56
6.12	5th iteration	56
6.13	10th iteration	56
6.14	Equilibrium buoyancy spread out over 4 segments	57
6.15	Highest point of the s-shape vs number of segments per section	58
6.16	Convergence spreaded buoyancy	58
6.17	Buoyancy distribution	59
6.18	Segment length with varying buoyancy per section	62
6.19	Schematization 8 modules per section and 5 segments per section	63
6.20	Zoom of schematization 8 modules per section and 5 segments per section	63
6.21	Catenary model of weight Jumper hose	64
6.22	S-shape Jumper hose	64
6.23	Tension per node catenary shape and s-shape	65
7.1	Length Variation	70
7.2	Height variation	70
7.3	Number of modules variation	70
7.4	Modules per segment variation	71
7.5	Starting segment variation	71
7.6	Range variation	71
7.7	Hose substance variation	72
7.8	Optimal parameter set	77
7.9	Sensitivity length hose (water)	78
7.10	Sensitivity length hose (slurry)	78
7.11	Sensitivity height connection VTS (water)	79
7.12	Sensitivity height connection VTS (slurry)	79
7.13	Sensitivity modules (water)	80
7.14	Sensitivity modules (slurry)	80
7.15	Sensitivity modules clamped per hose segment (water)	81
7.16	Sensitivity modules clamped per hose segment (slurry)	81
7.17	Sensitivity starting segment buoyancy modules (water)	82
7.18	Sensitivity starting segment buoyancy modules (slurry)	82
7.19	Optimal parameter set	83
8.1	Design of subchapter 7.5 implemented in OrcaFlex	87
8.2	Implementing buoyancy modules in stead of applied forcing on the nodes	88
8.3	Implementing buoyancy modules in stead of applied forcing on the nodes	89
8.4	Implementing buoyancy modules in stead of applied forcing on the nodes	90
8.5	Base model, 3 segments per section vs 5	91
8.6	Base model, 10 segments per section vs 5	91
8.7	Base model, 23 segments per section vs 5	91
8.8	Hose with pinned nodes vs without pinned nodes	92
8.9	Continuous hose structure vs Matlab model	92
8.10	Bending stiffness incorporated vs MatLab model	92
8.11	Hose with pinned nodes vs without pinned nodes	93
8.12	Continuous hose structure vs Matlab model	93
8.13	Bending stiffness incorporated vs MatLab model	93
8.14	Hose with pinned nodes vs without pinned nodes	93
8.15	Continuous hose structure vs Matlab model	93
8.16	Bending stiffness incorporated vs MatLab model	93
8.17	Hose with pinned nodes vs without pinned nodes	93

8.18	Continuous hose structure vs Matlab model . . . . .	93
8.19	Bending stiffness incorporated vs MatLab model . . . . .	93
8.20	Bending stiffness incorporated vs MatLab model, spline method . . . . .	93
8.21	Hose sections composition vs MatLab model . . . . .	95
8.22	Shape of parameter set of table 8.15, applied forcing . . . . .	97
8.23	Overview OrcaFlex calculations, applied forcing . . . . .	97
8.24	Static OrcaFlex model . . . . .	99
8.25	Static OrcaFlex model, z coordinate . . . . .	99
8.26	Static OrcaFlex model, effective tension . . . . .	100
9.1	OrcaFlex, effective tension [22] . . . . .	104
9.2	Effective tension over the arclength, parameters table 9.1 [22] . . . . .	105
9.3	OrcaFlex model at 0 seconds . . . . .	106
9.4	OrcaFlex model at 100 seconds . . . . .	106
9.5	OrcaFlex model at 200 seconds . . . . .	106
9.6	Timelapse OrcaFlex results . . . . .	107
9.7	Results OrcaFlex model, position node A, End Force . . . . .	107
9.8	Results OrcaFlex model, position node A, Moment in y . . . . .	107
9.9	Results OrcaFlex model, position node A, effective tension . . . . .	108
9.10	Results OrcaFlex model, position node A, end moment . . . . .	108
9.11	Results OrcaFlex model, position node A, end moment . . . . .	108
9.12	Results OrcaFlex model, position node A, end moment . . . . .	108
9.13	Results OrcaFlex model, position node A, effective tension . . . . .	109
9.14	Results OrcaFlex model, position node A, x displacement . . . . .	109
9.15	OrcaFlex model at 0 seconds . . . . .	110
9.16	OrcaFlex model at 200 seconds . . . . .	110
9.17	OrcaFlex model at 320 seconds . . . . .	110
9.18	Timelapse OrcaFlex results . . . . .	110
9.19	Results OrcaFlex model, position node A, End Force . . . . .	111
9.20	Results OrcaFlex model, position node A, Moment in y . . . . .	111
9.21	Results OrcaFlex model, position node A, effective tension . . . . .	111
9.22	Results OrcaFlex model, position node A, end moment . . . . .	111
9.23	Results OrcaFlex model, Effective tension over arclength . . . . .	112
9.24	Results OrcaFlex model, z-position over arclength . . . . .	112
9.25	OrcaFlex model at 0 seconds . . . . .	113
9.26	OrcaFlex model at 300 seconds . . . . .	113
9.27	OrcaFlex model at 420 seconds . . . . .	113
9.28	Timelapse OrcaFlex results . . . . .	114
9.29	Results OrcaFlex model, position node A, End Force . . . . .	114
9.30	Results OrcaFlex model, position node A, Moment in y . . . . .	114
9.31	Results OrcaFlex model, position node A, effective tension . . . . .	115
9.32	Results OrcaFlex model, position node A, end moment . . . . .	115
9.33	Results OrcaFlex model, Effective tension over arclength . . . . .	115
9.34	Results OrcaFlex model, z-position over arclength . . . . .	115
9.35	Results OrcaFlex model, position node A, End Force . . . . .	116
9.36	Results OrcaFlex model, position node A, Moment in y . . . . .	116
9.37	Results OrcaFlex model, position node A, effective tension . . . . .	116
9.38	Results OrcaFlex model, position node A, end moment . . . . .	116
9.39	Results OrcaFlex model, position node A, x displacement . . . . .	116
9.40	Schematic side view range . . . . .	117
9.41	Schematic side view range . . . . .	117
9.42	Schematic side view range . . . . .	117
9.43	Schematic overview range . . . . .	118
9.44	OrcaFlex model at 0 seconds . . . . .	119
9.45	OrcaFlex model at 170 seconds . . . . .	119
9.46	OrcaFlex model at 340 seconds . . . . .	119
9.47	OrcaFlex model at 500 seconds . . . . .	119

9.48 Time laps, x 50 m range 86 m, pan view . . . . .	120
9.49 OrcaFlex model at 0 seconds . . . . .	122
9.50 OrcaFlex model at 130 seconds . . . . .	122
9.51 OrcaFlex model at 260 seconds . . . . .	122
9.52 OrcaFlex model at 360 seconds . . . . .	122
9.53 Time laps, x 100 m range 65 m, pan view . . . . .	123
9.54 OrcaFlex model at 0 seconds . . . . .	125
9.55 OrcaFlex model at 94 seconds . . . . .	125
9.56 OrcaFlex model at 188 seconds . . . . .	125
9.57 OrcaFlex model at 290 seconds . . . . .	125
9.58 Time laps, pan view . . . . .	125
9.59 Floor plan, range crawler . . . . .	127
10.1 Optimal parameter set, static analysis . . . . .	129
10.2 Floor plan, range crawler . . . . .	130
11.1 OrcaFlex model Motion connection point clump weight . . . . .	132
11.2 OrcaFlex model Tension connection point . . . . .	132
11.3 OrcaFlex model Tension node 29 connection point clump weight . . . . .	133
11.4 OrcaFlex model End force connection point . . . . .	133
11.5 OrcaFlex model Effective tension over arce length . . . . .	133
11.6 Path crawler with minimum rotation radius . . . . .	134
11.7 Path crawler with minimum rotation radius, scope crawler . . . . .	135
11.8 OrcaFlex model crawler speed 1 m/s, 75 sec . . . . .	136
11.9 OrcaFlex model crawler speed 1 m/s, 150 sec . . . . .	136
11.10 OrcaFlex model time steps VTS and crawler same speed in x-direction . . . . .	136
11.11 OrcaFlex model, crawler pattern and clump weight pattern . . . . .	137
11.12 OrcaFlex model, crawler pattern and clump weight pattern . . . . .	138
A.1 Iterations Catenary, number 1 to 9 . . . . .	143
A.2 Iterations Catenary, number 10 to 27 . . . . .	144
A.3 Iterations 1 Node, number 1 to 15 . . . . .	145
A.4 Iterations 1 Node, number 16 to 30 . . . . .	146
A.5 Iterations 1 Node, number 31 to 42 . . . . .	147
A.6 Iterations s-shape, number 1 to 15 . . . . .	148
A.7 Iterations s-shape, number 16 to 33 . . . . .	149
A.8 Iterations s-shape, number 34 to 36 . . . . .	150
A.9 Plots number of elements per segment 1 to 15 . . . . .	151
A.10 Plots number of elements per segment 16 to 30 . . . . .	152



# List of Tables

1.1	Terminology of components and other terms [16], [28], [15], [2]	5
2.1	Classification Benthic zone	8
2.2	Classification Pelagic zone	8
3.1	Licence area parameters	13
3.2	Average mineral composition polymetallic nodule	14
3.3	Soil parameters, [28]	14
3.4	Nodules parameters, [28]	14
3.5	Seawater, bottom parameters, [28]	15
3.6	Sea surface parameters, [28]	15
3.7	Initial design parameters components	17
3.8	Parameters and Requirements Jumper hose design	18
3.9	Parameters buoyancy modules by the buoyancy supplier	18
3.10	Environment en design parameters	21
3.11	Base set of parameter jumper hose by the hose supplier	21
4.1	Coordinates VTS and Crawler for model	25
4.2	Parameters	31
4.3	Result own weight jumper hose in air	31
4.4	Result weight jumper hose in water	31
5.1	Velocity and length characteristics for Reynolds number	42
6.1	Preliminary design parameters	52
6.2	Preliminary design parameters	52
6.3	Results constants initial guess parabola	53
6.4	$P_{ext}$ due to weigh jumper hose filled with slurry	54
6.5	$P_{ext}$ due to weigh jumper hose filled with slurry and buoyant node 6	55
6.6	Number of Nodes, DOFs and other data	55
6.7	$P_{ext}$ due to weigh jumper hose filled with slurry and buoyant nodes 4 to 8	57
6.8	Highest node vs corresponding arclength	57
6.9	Variation parameters, influence buoyant sections	60
6.10	Variation parameters, influence buoyant sections	60
6.11	Dimensions spacing between buoyancy modules	61
6.12	Node locations of the segment	61
6.13	Locations centre buoyancy modules	61
6.14	Results calculation of buoyant nodes	62
6.15	Results static model summarized	65
7.1	All parameters of the jumper hose design	68
7.2	Maximum workability values of the failure criteria	69
7.3	Workability criteria	69
7.4	Weigh of the hose per length and number of buoyancy modules	73
7.5	Range width per parameter	73
7.6	Parameter set, composed of parameter combinations	74
7.7	Weight factors of MCA	75
7.8	Ranking parameter sets based on lowest score of a parameter set	76
7.9	Reranking 25 parameter set, first 10 presented	76
7.10	MCA results for sensitivity study length	78
7.11	MCA results for sensitivity study connection point VTS	79
7.12	MCA results for sensitivity study total number of modules	80

7.13 MCA results for sensitivity study clamped modules per segment . . . . .	81
7.14 MCA results for sensitivity study segment buoyancy modules . . . . .	82
8.1 Concept design parameters, subchapter 7.5 . . . . .	85
8.2 Design parameters jumper hose . . . . .	86
8.3 Pinned joint locations . . . . .	86
8.4 Buoyant nodes . . . . .	87
8.5 Design parameters buoyancy modules in OrcaFlex . . . . .	88
8.6 Updated design parameters buoyancy modules in OrcaFlex . . . . .	89
8.7 Parameters . . . . .	90
8.8 Simulation time . . . . .	91
8.9 Colour labels, with corresponding figures . . . . .	92
8.10 Hose structure . . . . .	94
8.11 Hose section, structure weight . . . . .	95
8.12 Weight distribution jumper hose . . . . .	95
8.13 Weight of the hose and buoyancy modules . . . . .	96
8.14 Range design parameters for parameter study . . . . .	96
8.15 Best parameter combination after multicriteria analysis . . . . .	96
8.16 Difference in node locations . . . . .	98
8.17 Difference in node locations . . . . .	98
8.18 MatLab OrcaFlex model vs failure criteria . . . . .	99
8.19 Design parameters for starting point dynamic analysis . . . . .	100
9.1 Static design parameters . . . . .	101
9.2 Failure criteria jumper hose . . . . .	102
9.3 Effective tension check . . . . .	104
9.4 Effective tension check . . . . .	105
9.5 Limitations dynamic design . . . . .	105
9.6 Movement of the crawler node . . . . .	106
9.7 Results OrcaFlex model 50 meters in x direction . . . . .	107
9.8 Movement of the crawler node . . . . .	109
9.9 Results OrcaFlex model 100 meters in x direction . . . . .	111
9.10 Movement of the crawler node . . . . .	113
9.11 Results OrcaFlex model 150 meters in x direction . . . . .	114
9.12 Range calculations for evaluated distances . . . . .	118
9.13 Movement of the crawler node . . . . .	118
9.14 Results OrcaFlex model 50 meters in x direction, 50 m in y-direction . . . . .	120
9.15 Range test, critical values, model 9.48 . . . . .	120
9.16 Range test, critical values, model 9.48 . . . . .	121
9.17 Results OrcaFlex model 50 meters in x direction, range y direction . . . . .	121
9.18 Movement of the crawler node . . . . .	121
9.19 Results OrcaFlex model 100 meters in x direction, 65 m in y-direction . . . . .	123
9.20 Range test, critical values, model 9.53 . . . . .	123
9.21 Range test, critical values, model 9.53 . . . . .	124
9.22 Results OrcaFlex model 10 meters in x direction, range y direction, model 9.53 . . . . .	124
9.23 Movement of the crawler node . . . . .	124
9.24 Results OrcaFlex model 150 meters in x direction, 47 m in y-direction . . . . .	126
9.25 Range test, critical values, model 9.58 . . . . .	126
9.26 Range test, critical values, model 9.58 . . . . .	126
9.27 Results OrcaFlex model 10 meters in x direction, range y direction . . . . .	127
9.28 Range Crawler to VTS . . . . .	127
10.1 Concept design parameters, subchapter 7.5 . . . . .	129
10.2 Final design parameters, subchapter 8.2 . . . . .	130
11.1 Results OrcaFlex model heave motion at connection clump weight . . . . .	132
11.2 Results OrcaFlex model, heave motion and perpendicular motion . . . . .	134
11.3 Design parameters path of the crawler . . . . .	134

---

11.4 Results OrcaFlex model, heave motion and perpendicular motion . . . . .	135
11.5 Results orcaflex model . . . . .	137
11.6 Length path with corresponding clump weight speeds . . . . .	138
B.1 Specific collector initial design parameters . . . . .	153
B.2 Specific Umbilical Design Parameters . . . . .	153
B.3 Specific Jumper Assembly Design Parameters . . . . .	153
B.4 Hose parameters in initial design . . . . .	153
B.5 Parameters Jumper hose design . . . . .	154



# Introduction

Climate change is a hot topic globally, with the Paris agreement treaty legally binding international countries to limit global warming to well below 2 degrees Celsius compared to pre-industrial levels. Due to human activity and the extra amount of greenhouse gases released into the atmosphere, the average temperature on Earth is rising fast. A very rapid transition is needed to sustainable energy generation and consumption to reduce these emissions.

With a growing population, technology developments, and a clean energy future, the demand for the metal incorporated in smartphones, electric cars, solar panels, batteries, and other power storage is increasing rapidly. A famous quote is that "if it is not grown, it's mined". These metals are currently extracted from land-based locations. But the extractable amount is limited, and some of these resources might soon be exhausted. However, there is another source of these precious minerals. They can be found in certain areas, deep down on the ocean floor. There are three main types of deposits on the seafloor, polymetallic nodules, hydrothermal vents and a rich cobalt crust on seamounts.

Deep-sea mining is still in its experimental stage, but scientists are already worried about its impacts on deep-sea marine life and ecosystems. Scientists have mapped more of the surface area of Mars than of the ocean floors on Earth. And in the past 20 years, marine biologists have discovered thousands of new species and vast ecosystems. It's an uncomfortable choice between the potential negative impact or benefit of extracting the metals needed for a more sustainable future. The impacts of deep-sea mining is yet unknown and could be potentially huge. "So the question remains, can we do a better job in the ocean than we do on land?" [31]

## 1.1. Background

### 1.1.1. History of mining

For centuries, people mined raw materials from the Earth, but since the industrial revolution, the focus of survival shifted from primarily producing food to extracting certain materials for automated processes. The industrial revolution was all about using different forms of energy to automate production. For example, the steam engine was invented and optimised. This machine was used for production processes, transportation and automating processes. The steam engine ran on coal, so the demand for coal increased rapidly, resulting in society's industrialisation. The second and third industrial revolution brought the world to our current way of living and, in particular, fossil fuels.

Mining is dangerous and destructive work with risks of explosions, floor collapse, floating and other hazards. Rules and regulations came in order, mostly after major accidents. Society depends greatly on natural resources like metals and precious materials like diamonds. Unfortunately, mining goes hand in hand with exploitation, destroying ecosystems and child labour. Mining takes up a large surface area that leaves a depleted and polluted area after a mine is abandoned. It leaves a scar on the land and a huge impact on the area. Examples are groundwater contamination and worn-out ecosystems, which drive the native inhabitants away.

### 1.1.2. The need for an energy transition

Burning fossil fuels for power generation releases greenhouse gases into the atmosphere. This contributed to the reinforcement of the greenhouse effect, which then added up to global warming. The extraction of fossil fuels harms the environment, and also, the stock of these raw materials is finite, so

it gets harder to extract them. In 2015 a climate agreement was reached to slow down global warming. This agreement is between 55 countries responsible for about 55 % of the production of greenhouse gases. The agreement states that the increase in the global average temperature to well below 2 °C above pre-industrial levels and pursuing efforts to limit the temperature increase to 1.5 °C above pre-industrial levels, recognising that this would significantly reduce the risks and impacts of climate change [1].

An energy transition is needed to live up to the Paris agreement made in 2015. The energy supply has to be changed drastically by replacing fossil fuels with sustainable resources in this transition. But there is also attention on energy saving, storage and decentralized organisation. For the energy transition, a lot of different raw materials are needed. The demand rises because the global population keeps growing. Many of these materials lie deep beneath the Earth's crust and are hard to reach. These raw materials can also be founded on the ocean floor. For example, in the Pacific Ocean is an area located called the Clarion Clipperton Zone. Here billions of potato-sized rocks called polymetallic nodules lie on the ocean floor at around 5 kilometres. These nodules contain metals like cobalt, nickel, iron and manganese, needed for a clean energy future.

### *1.1.3. Overall design Blue Nodules project*

On 1 February, a European consortium launched a new Horizon 2020 project: Blue Nodules. This project addresses the challenge of creating a viable and sustainable value chain to retrieve polymetallic nodules from the ocean floor" [10]. This project is a unique combination of leading academics and the industry to further develop the technology to mine nodules from the seafloor with minimal environmental impact. IHC is one of the partners on this project and has designed the seafloor nodule collector. The project's focus has been on developing a nodule collector and its flow process while considering the impact of sediment plumes, compaction of the seabed, noise and loss of biodiversity. This collector is connected to a vertical transport system with a hose and an umbilical. The nodules are collected with a crawler and flushed through a hose to the vertical transport system. The vertical transport system transports the nodules to the ship. The total operation system for the Blue nodules projects consists of the following items, and these are shown in figure 1.1.

The main components of the mining operation are:

- The Surface operation vessel
- The vertical transport system (VTS)
- The Jumper hose (JH), also referred to as the flexible connection
- The Crawler, also referred to as the Seafloor Mining Tool (SMT)

The design structure is as follows: The subsea harvesting equipment is a seafloor mining tool connected to the offshore production vessel. This connection consists of a flexible and rigid vertical transport system with booster pumps and an umbilical cable for the electrical power supply.

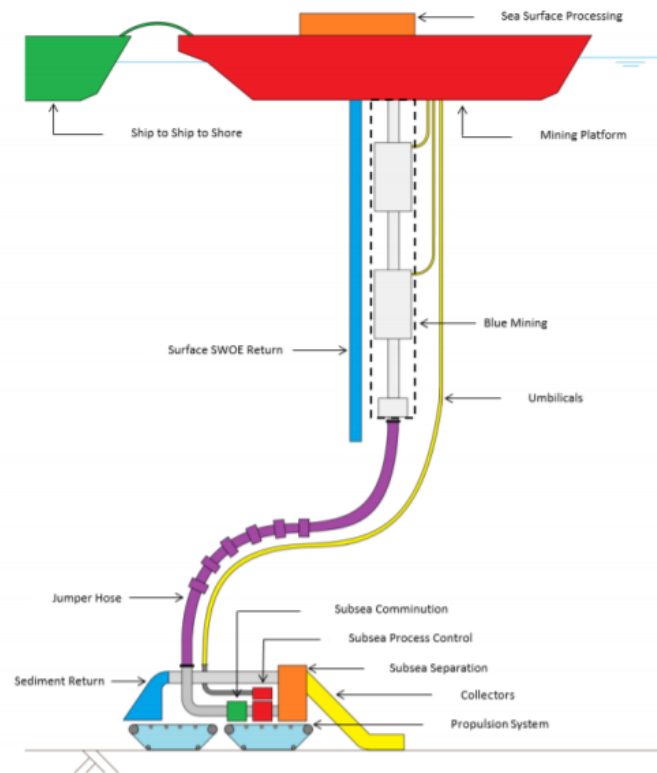


Figure 1.1: Schematic overview of the main components of the system [28]

## 1.2. Problem definition

The Blue nodules project was finished in 2020, delivered a prototype vehicle, and tested many components. Much progress has been made but more remains to be done. Royal IHC continues to develop this deep-sea mining system. One of the topics of interest for further development is the manoeuvrability of the seafloor mining tool in combination with the dynamic behaviour of the flexible connection. In this research study, the dynamical behaviour of the flexible connection is analysed. The study includes the interaction between the flexible connection with a moving seafloor mining tool during different production rates and scenarios.

### 1.2.1. Objectives

This research is focused on the flexible connection between the seafloor mining tool and the vertical transport system of the Blue nodules project. It aims to get more insight into the dynamic behaviour of the flexible connection and its interaction with the seafloor mining tool during production operation.

The first main objective of the research is: to construct a static model of a geometrically non-linear system with large axial deformations. This model examines the current parameters of the flexible connections and determines if a new set of parameters is required. The second objective is the dynamic analysis of the flexible connection. The parameters of the static analyses are implemented into the dynamic software package called "Orcaflex". With this software, the dynamic behaviour of production operations is investigated. Therefore the purpose of this study is to answer the following research questions:

*Determine the dynamical behaviour of flexible hose in three dimensions and its interaction with the moving seafloor mining tool during different scenario's.*

This research has two parts. The first part focuses on the static case, examining and designing the parameters and dimensions of the flexible connection. The second part is focused on the dynamic behaviour of the flexible connection in the operation phase and the interaction with the seafloor mining tool. For the first part, the following sub-questions have been answered:

- What is the preliminary base design of the flexible connection from the Blue nodules project?
  - *What are the design parameters?*
  - *What are the requirements and limitations of the project?*
- How is the static model constructed, and which methodology is used?
  - *Which simplifications and adaptations are necessary for the formulation of the static model?*
- How is the new/optimal set of parameters identified?

And for the second part, the following sub-questions have been answered:

- What is the theory behind the dynamic software used, and does it connect with static analysis?
- What are the results of the static analysis using OrcaFlex?
- Which simulations are performed for the dynamic analysis?
- How manoeuvrable is the seafloor mining tool?

### 1.2.2. Scope of the thesis

To ensure a feasible study, some limitations are imposed on the scope. This research is conducted under the supervision of Royal IHC. Royal IHC has distributed the preliminary design of the flexible connection and the other components of the deep-sea mining system.

The following scope limits the research:

- Royal IHC distributes most of the data used.
- Royal IHC has set some requirements on the design parameters and wants to investigate specific parameter options. These assumptions are clearly stated in the thesis.
- This research is limited to a static and dynamic study on the flexible connection. Design parameters like the diameter of the flexible hose and type of material are outside the scope.

It could be that there are more limitations encountered. They are explicitly clarified in a sub-chapter of the respective chapter.

## 1.3. Approach

To answer the research questions, the following approach is taken. First, the preliminary design of the flexible connection is investigated, and possible new requirements are set up. Also, the environmental data are analysed. This information is all needed to build a simplified static case to determine feasible sets of parameters. With a Multicriteria analysis, the final optimal set of parameters is determined. These parameters are the output of the first part of the thesis. This output is used in the second part. The theory used for the static calculation is compared to the software for the dynamic analysis to see if the optimal parameter set is the starting point for the dynamic design. With the dynamical software, the dynamical behaviour of the flexible connection is determined relative to the movements of the seafloor mining tool. With the dynamic study results, conclusions and recommendations are drawn.



### 1.3.1. Denomination of the components of the subsea harvesting equipment and other components

Different denominations of the subsea harvesting equipment components in prior research are used. Throughout this thesis, it is attempted to keep the terminology consistent; however, different terms may appear. These components can be referred to with different terminology. In table 1.1, the terminology for the mining operation is displayed. The abbreviations are also found in the Nomenclature chapter.

Introduced component term	The most commonly used component term	Other terminology
Seafloor Mining Tool (SMT)	Seafloor nodule collector	Crawler
Flexible connection	Jumper hose (JH)	
Vertical Transport System (VTS)		Riser system
Surface operation vessel		Ship
Polymetallic nodules	Nodules	Manganese nodules
Subsea harvesting equipment	Total operation system	

Table 1.1: Terminology of components and other terms [16], [28], [15], [2]



# The ocean and its resources

This chapter consists of a brief introduction to the deep sea, focussing on the resources found on the seabed. First, the ocean's depths classification is documented with the bathymetry and influence of plate tectonics. Next, the different resources of the ocean floor are presented like hydrothermal vents, cobalt crust and polymetallic nodules. This chapter gives the background of the working environment of the Blue nodules project.

## 2.1. Classifying the ocean depths

As mentioned in the introduction, the metal needed to transition to a fossil-free future is currently extracted from the land. These resources are slowly getting exhausted, and the US geological survey has shown that the deep sea contains more of these metals than all land-based reserves combined. In figure 2.1, the hypsographic curve of the earth is shown. This hypsographic curve shows the relationship between the height of the land surface and the depth of the oceans. This cumulative curve shows that around 29 % of the surface is above sea level. The rest of the earth's surface is below sea level, and only around 15% of this area is mapped.

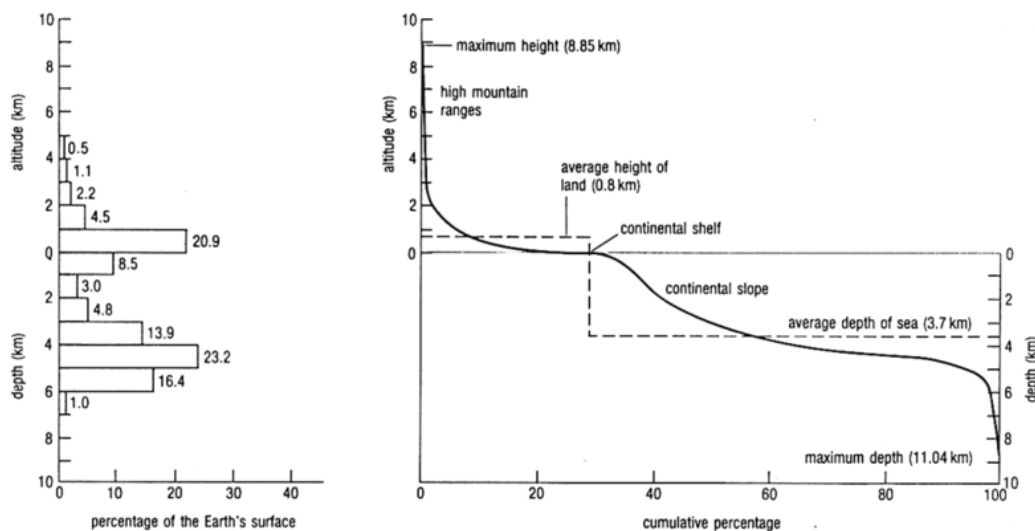


Figure 2.1: Hypsographic curve of the surface of the Earth.[11]

The hypsographic curve shows that the ocean's depth varies from 0 to 11 km below the sea surface. The ocean depths can be classified based on depth ranges and geological bathymetry. There are two classification systems based on depth range. The first applies to organisms floating in the pelagic zone's water column, and these pelagic zones are expressed in ocean depth. The second classification applies to organisms living on the ground, the Benthic zone. This classification is not described in water depth but with ranges over the seafloor. A schematic picture is displayed in figure 2.2. Table 2.1 describes the classification of the benthic zone and table 2.2 describes the classification of the pelagic

zone.

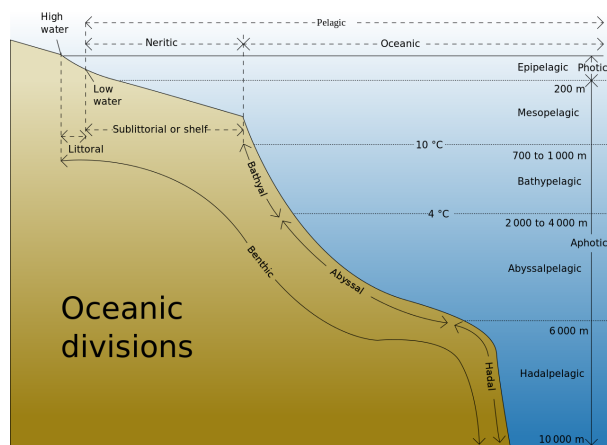


Figure 2.2: Oceanic division Pelagic and Benthic zone Earth.[34]

Benthic zone

Zone	Classification
Supralittoral zone	Above high tide line
Littoral zone	Between high and low tide line
Sublittoral zone	Between low tide line and shelf break
Bathyal zone	From shelf break up to 4000 meters depth, describing the continental slope and rise
Abyssal zone	Describes most of the abyssal plains
Hadal zone	Deeper than 6000 meters, describing the deep ocean trenches

Table 2.1: Classification Benthic zone

Pelagic zone

Province	Description	Zone	Depth [m]	Characteristics
Neritic	All water above the low tide line to the shelf break		0-200	The shelf break varies from 20 meters depth to 550 meters. 200 meters is the average depth.
		Epipelagic zone	0 – 200	The Epipelagic zone of the oceanic province is thoroughly penetrated by light, and so photosynthesis is possible
Oceanic	All water in the open ocean region		0 - 11000	
		Mesopelagic zone/ Twilight zone	200 – 1000	The Mesopelagic zone of the oceanic province is partly penetrated by light, not enough for photosynthesis
		Bathypelagic zone/ Midnight zone	1000 - 4000	In the Bathypelagic zone depth range of the oceanic province is no light
		Abyssopelagic zone	4000 - 6000	The Abyssopelagic zone of the oceanic province exists primarily of abyssal plains
		Hadopelagic zone	Below 6000	The Hadopelagic zone of the oceanic province describes the water in the deep ocean trenches

Table 2.2: Classification Pelagic zone

### 2.1.1. The ocean bathymetry and plate tectonics

Bathymetry is the depth of the landforms below sea level, and it ties very closely to plate tectonics. “The theory of plate tectonics states that the earth’s solid outer crust, the lithosphere, is separated into plates that move over the asthenosphere, the molten upper portion of the mantle. Oceanic and continental plates come together, spread apart, and interact at boundaries all over the planet. Each type of plate boundary generates distinct geologic processes and landforms. At divergent boundaries, plates separate, forming a narrow rift valley. Here, geysers spurt super-heated water, and magma, or molten rock, rises from the mantle and solidifies into basalt, forming a new crust. Thus, at divergent boundaries, oceanic crust is created. At convergent boundaries, plates collide with one another. The collision buckles the edge of one or both plates, creating a mountain range or subducting one of the plates under the other, creating a deep seafloor trench. At convergent boundaries, continental crust is created, and oceanic crust is destroyed as it subducts, melts, and becomes magma” [4].

Considering the seafloor, the region that describes the transition from land to the deep seafloor depends on the continental margin's tectonic plate movement. It can be distinguished into two types, active continental margin and passive continental margin. Figure 2.3 depicts these two margins. An active continental margin occurs at a tectonic plate boundary where the ocean tectonic plate and continental tectonic plate move towards each other. The oceanic plate moves underneath the continental plate. The seafloor close to the surface above the water level is characterised by a narrow offshore shelf transitioning from a steep slope into a deep trench. These are the deepest parts of the oceans. The trench marking the boundary of the plate and the deep ocean floor starts at the seaward side of the trench. "A passive continental margin occurs where the transition from land to sea is not associated with a plate boundary" [33]. This margin is divided into three regions, the continental shelf, the shelf break and the continental slope. The bottom of the continental slope is called the continental rise, where the transition from continental crust to oceanic crust meets, eventually becoming the deep ocean floor.

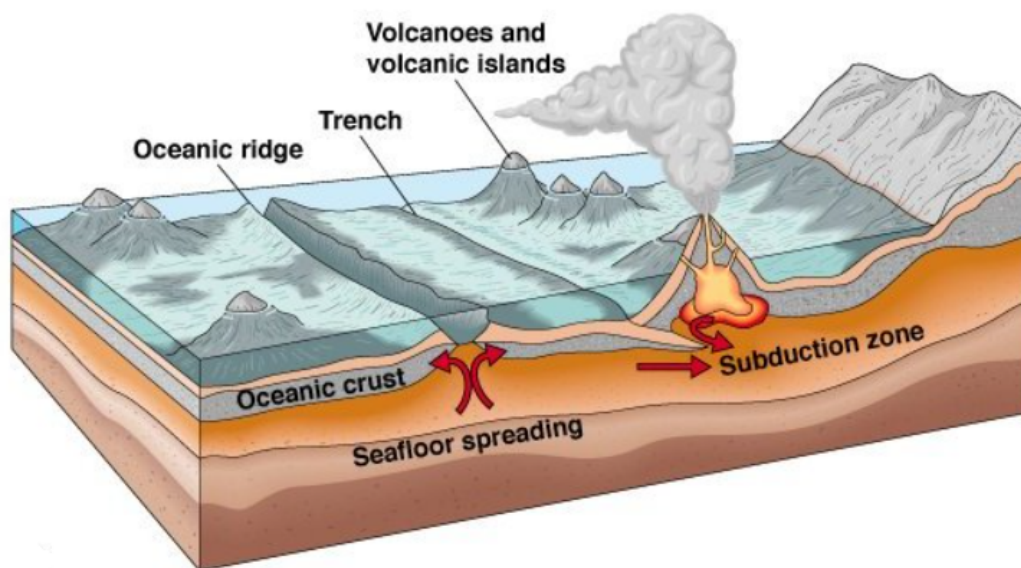


Figure 2.3: Diverging plate tectonics and convergent plate tectonics[32]

The deep ocean floor covers about 42 per cent of the earth's surface, with average depths ranging from 4500 to 6000 meters. This deep ocean floor is covered with large regions covered by abyssal plains. Abyssal plains are large flat regions covered by a thick layer of sediment called marine snow. At a diverging boundary, where the tectonic plates move away from each other. This creates a new oceanic crust due to volcanic activity, a ridge system. There are other particular bathymetric features in the ocean like seamounts, abyssal hills or geodes. Due to individual volcanic activity, they rise off the seafloor and are formed over millions of years. These are classified by height. If an undersea volcano is less than a thousand meters high, measured for the seafloor, it's called an abyssal hill. If they rise over a thousand meters from the seafloor, it's called a seamount. These seamounts can have a flattened top due to erosion by waves at some point in time, and then it's called a geode.

## 2.2. Resources in the Ocean

It's been known that metals are present on the deep-sea ocean floor. In 1874 the first polymetallic nodules were found during the Challenger expedition. "The US Geological Survey stated that the deep sea contains more nickel, cobalt and other rare earth metals than all land-based reserves combined" [23]. There are three main resources on the seafloor: hydrothermal vents at plate boundaries, a cobalt-rich ferromanganese crust on seamounts and polymetallic nodules in nodule fields on abyssal plains.

### 2.2.1. Hydrothermal vents systems

When two tectonic plates diverge from each other, they create underwater ridges (figure 2.4 and 2.5). This new oceanic floor has faults like cracks and porous sediment, allowing seawater to seep into the crust deep into the earth. When the seawater is underground, it leaches metals from the surrounding ground, full of vast amounts of sulphur. When this water comes close to the magma chambers, it gets heated and rises quickly, under tremendous pressure, towards the ocean floor, bursting out through the gaps called vents. So if the hot seawater bursts out of the vents and interacts with the cold water, the minerals will precipitate and form tiny particles around the vents. These form chimney-like structures and are called hydrothermal vents. Chimneys are also called massive sulphide deposits due to that the vent fluid mainly exists of sulphide, which is a combination of sulphur with a metal.

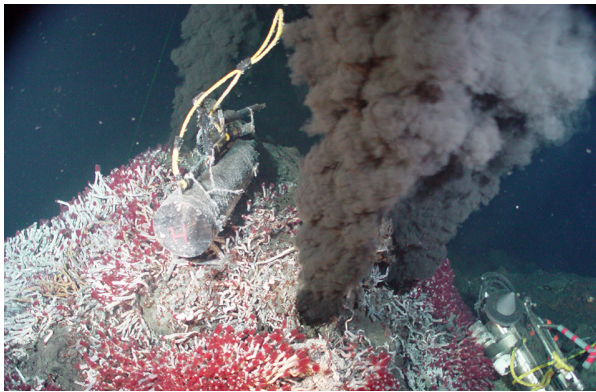


Figure 2.4: Hydrothermal vent or black smoker.[25]

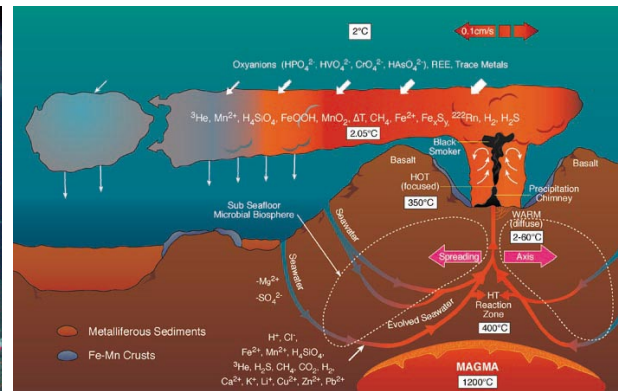


Figure 2.5: Cross section ridge system.[5]

Hydrothermal vents can spit out sulphides for thousands of years. When a vent system spits out dense vent fluid, it's called an active system. When a system is no longer spitting out fluids, it's called an inactive system. They are primarily found at the plate boundaries up to 5800 meters of water depth but can also be found at volcanoes or seamounts of subduction zones. They rarely appear in the middle of a plate. If they do, they are called hotspots.

Hydrothermal vents are also one of seawater's original iron and manganese resources. These minerals are the base for forming the other two resources found on the ocean floor.

### 2.2.2. Cobalt crust

Another type of deposit is the cobalt-rich ferromanganese crust. This is found at seamounts, ridges, sea knolls, and plateaus left by extinct volcanoes, at depths ranging from 600 to 6000 meters. The name cobalt crust is a little misleading because the crust composition is mainly manganese and iron. In some areas, the crust contains around 2% cobalt, making them economically relevant. These compositions are found at 800 to 2500 meters.

The cobalt crusts are primarily found in high turbidity areas, which occur at the peaks of seamounts and other elevations of the ocean floor. These eddies can trap nutrients and metals and eventually are deposited on the rock. The manganese and iron oxide attach to the rock's surface and, due to chemical processes, form compounds, which form a sponge-like network. The nutrient and metals deposited from the eddies fall on the sponge-like surface and are trapped in the pores. Most of the elements found in the crust are eroded initially material from the continents, washed out by rivers and transported to the ocean (figure 2.6 and 2.7).

The economically relevant crust is primarily found in the western Pacific. This area is called the prime crusts zone. The growth rate of the crust is extremely slow and is about one to five millimetres in a million years. This growth also depends on the strength and direction of the ocean currents.



Figure 2.6: Picture of the cobalt-rich ferromanganese crust in the northern pacific ocean [13]

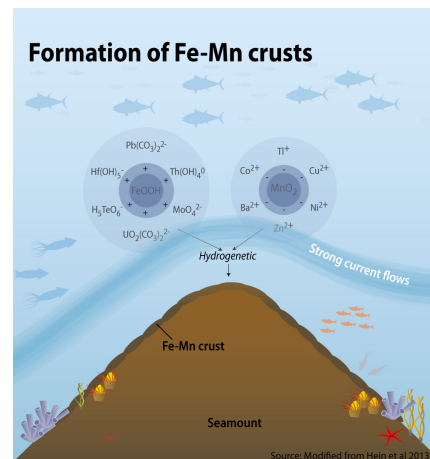


Figure 2.7: Cross section seamount formation cobalt-rich ferromanganese crust [8]

### 2.2.3. Polymetallic nodules

Polymetallic nodules or manganese nodules are lump rocks that are mostly found on the seafloor at depths of 3500 to 6000 meters below sea level. These accumulations contain smaller amounts of manganese, iron, cobalt, and other metals like titanium and barium. The nodules vary in size from macroscopically small to up to 20 cm in diameters, and their shapes are quite irregular. However, on average, the shape looks like a potato with a size between 3 to 8 centimetres.

The formation of polymetallic nodules is one of the slowest geological phenomena. They grow around a small particle, like a shark tooth, shell or a small rock fragment. Then as time passes, manganese and iron oxides are attached to the particle and form concentric layers around it. This growth process is prolonged, around a centimetre over several million years (figure 2.8 and 2.9).

These nodules are found in concentrated locations on the deep seafloor worldwide. They lie loosely on the seabed, partly or entirely buried in a sediment layer called marine snow. These nodule fields can cover up to over 70% of the seafloor. Primarily found in the north-eastern pacific extending from the west coast of Mexico to Hawaii. This place is called the manganese nodules belt.



Figure 2.8: Nodule field in the Pacific ocean [26]

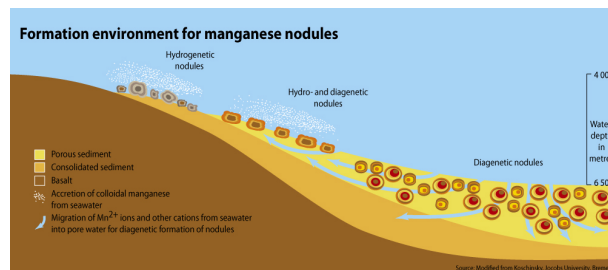


Figure 2.9: Formation environment for manganese nodules [7]





## Blue Nodules project characteristics

This chapter aims to collect and schematise all information relevant to the Blue Nodule project and the jumper hose. Subchapter 3.1 specifies the environmental parameters and subchapter 3.2 focuses on the Blue nodule's preliminary design. Subchapter 3.3 reflects the follow-up on the preliminary jumper hose design, which also describes the outline of the thesis. The first subsection describes the recommendations for the design of the jumper hose that could be examined for a prospective design and the design adjustment during the project. The second subsection focuses on the requirements for this research, and the third subsection describes the parameters of the base design of the jumper hose. Subchapter 3.4 an answer and overview is given of the first research question, 'What is the preliminary base design of the flexible connection from the Blue nodules project?' with sub-questions 'What are the design parameters?' and 'What are the requirements and limitations of the project?'.

### 3.1. Environmental parameters and location

The Blue Nodules project is focused on mining polymetallic nodules in the Clarion Clipperton Zone, CCZ, for short. The CCZ is located in the central Pacific Ocean, between Hawaii and Central America, and it is a geological submarine fracture zone with abyssal plains. This area lies in international waters, and all mineral resources related activities are controlled by the International Seabed Authority (ISA). The ISA has expended 30 licences for exploration in the CCZ to explore their critical raw materials. In table 3.1 the date of a licence area is shown, and figure 3.1 shows the area of the CCZ with all the licence areas.

Item	value	unit
License area	75000	[km <sup>2</sup> ]
Minimum water depth	3000	[m]
Maximum water depth	6000	[m]
Design working water depth	5500	[m]

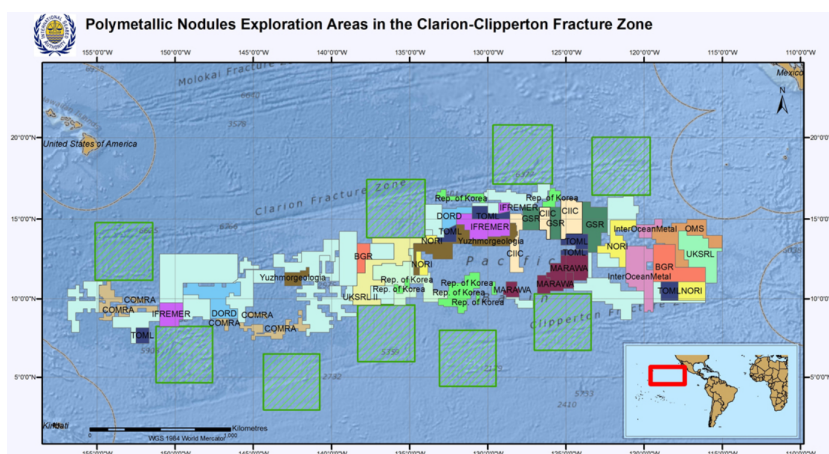


Table 3.1: Licence area parameters

Figure 3.1: Clarion Clipperton zone

### 3.1.1. Soil and nodule parameters

The polymetallic nodules are situated in nodules fields found on the abyssal plains of the CCZ. These polymetallic nodules consist of various rare earth minerals. The chemical composition of the minerals present in a nodule is shown in table 3.2. The geotechnical and geological data of the blue nodules project, specified on the nodules and soil composition, are displayed in table 3.3 and 3.4, respectively.

The jumper hose and VTS transport a mixture of nodules, sediment not derived at the seabed, nodule fragments and water. This mixture is then transported and referred to as slurry or wed bulk density.

Mineral	Percentage
Manganese	29 %
Iron	6 %
Silicon	5 %
Aluminium	3 %
Nickel	1,4 %
Copper	1,3 %
Cobalt	0,25 %
Oxygen	1,5 %
Hydrogen	1,5 %
Sodium	1,5 %
Calcium	1,5 %
Magnesium	0,5 %
Potassium	0,5 %
Titanium	0,2 %
Barium	0,2 %

Table 3.2: Average mineral composition polymetallic nodule

Item	Value	Unit
Grain size D50 Classification	8 - 16 Clay/ siliceous ooze	[ $\mu\text{m}$ ] [-]
Water content	200	[%-by mass]
Wet bulk density	1250 - 1350	[ $\text{kg}/\text{m}^3$ ]
Top layer	0 - 10	[cm below seafloor]
Undrained shear strength top layer	0 - 1,5	[kPa]
Lower layer	10 - 30	[cm below seafloor]
Undrained shear strength lower layer	2 - 4	[kPa]

Table 3.3: Soil parameters, [28]

Item	Value	Unit
Abundancy (wet density)	15 - 20	[ $\text{kg}/\text{m}^2$ ]
Abundancy (wet density) average	18	[ $\text{kg}/\text{m}^2$ ]
Diameter	0 - 15	[cm]
Burial depth	0 - 10	[cm]
Wet bulk density	2030	[ $\text{kg}/\text{m}^3$ ]
Dry bulk density	1500	[ $\text{kg}/\text{m}^3$ ]
Moisture content	37	[%-by mass]
Adhesion with soil	100 - 2000	[Pa]

Table 3.4: Nodules parameters, [28]

### 3.1.2. Metocean data

The primary sources for collecting metocean data are ARGOSS, NOAA, IHC, research data, and publicly available sources. The collected data is described in the "Blue Nodules Deliverable Report" [28]. The required data of the metocean data are summarised in table 3.5 and 3.6.

Item	Value	Unit	Item	Value	Unit
Bottom mean current	0,1	[m/s]	Air temperature	10-35	[°C]
Bottom water salinity	34,6 - 34,7	[‰]	Sea surface temperature average	26	[°C]
Bottom water density	1025 - 1045	[kg/m <sup>3</sup> ]	Surface current average	0,3	[m/s]
Average bottom water density	1035	[kg/m <sup>3</sup> ]	Surface current extreme	0,7	[m/s]
Bottom water temperature	1 - 4	[°C]	Regular wave maximal height	5,73	[m]
Working depth	3000 - 5500	[m]	Regular wave maximal period	7,34	[s]
Depth rating	6000	[m]	Wave return period	1	[year]
			Current return period	10	[year]
			Wind speed average	8	[m/s]
			Wind speed extreme	17	[m/s]

Table 3.5: Seawater, bottom parameters, [28]

Table 3.6: Sea surface parameters, [28]

## 3.2. Preliminary design of the Blue nodules project

"The blue nodules project has designed the layout of the entire logistics chain while tackling the complex problem of environmental protection up to technology readiness level 6 (TRL 6)" [3]. TRL is a level scale for innovative technologies divided into 9 levels and indicates how close the innovation is to the commercial market. The Blue Nodules project is at a level 6, indicating the end of the development phase and entering the demonstration phase.

The structure of the Blue Nodules project is divided into six work packages, with a work package being an element of the project that focuses on a specified task. The following work packages are part of the blue nodules project<sup>1</sup>:

- WP1 Value chain requirement for industrial viability
- WP2: Subsea harvesting equipment and control technology
- WP3: In situ seafloor processing of polymetallic nodules
- WP4: Sea surface and land operations/processes
- WP5: Environmental impact assessment
- WP6: Dissemination and exploitation

The preliminary design of the mining operation is part of WP2. The total system is called the seafloor harvesting equipment and consists of the following components (figure 1.1):

- Seafloor Mining Tool
- Jumper Hose
- Vertical transport system
- Umbilical (provides the seafloor mining tool off power)
- Surface operation vessel
- The process and handling of unwanted liquid and solid products, which is a byproduct of the nodules offshore mining and mineral processing. The byproduct is called the 'Sediment, Wastes and Other Effluents' (SWOE).

<sup>1</sup>Most of this information is classified and does not contribute to the Jumper hose design

### 3.2.1. Subsea Harvesting equipment

The subsea harvesting equipment is a seafloor mining tool that is connected to the offshore production vessel. This connection consists of the jumper hose (flexible connection) and a rigid vertical transport system with booster pumps and an umbilical cable for the electrical power supply.<sup>2</sup>

#### 3.2.1.1. Seafloor Mining Tool

The first vehicle developed was the Apollo 1 to test the functioning of the propulsion system. Its successor, the Apollo 2, was developed to test the collection system's general functioning. The Seafloor nodule collector is designed to minimise the environmental impact (figure 3.2). For example, it has 4 tracks instead of 2 to reduce the sediment plume created while driving. Also, a new method for separating the nodules from SWOE was designed. It separates the nodules from the segment inside the collector with an inclined tube. This process separates a major part of the SWOE from the nodules and happens inside the collector.

With computational models and simulations of the collector's plume, noise generated by the vehicle and the seafloor substrate alteration is reduced to a minimum. The full-scale vehicle will have all the basic features of the Apollo 2, including being all-electric.

The SMT is designed by royal IHC and is fully electrical, powered by the umbilical. This SMT removes the nodules from the seafloor by blasting them free and sucking them into the mining tool. A new method is designed to separate the sediment and the small aquatic species (attached to the nodules) from the nodules inside the collector itself. The sediment is deposited behind the collector, and the nodules plus some sediment not removed are flushed inside the jumper hose.

#### 3.2.1.2. Jumper hose

The hose supplier<sup>3</sup> has developed the jumper hose. The jumper hose forms a flexible link between the seafloor nodule collector and the vertical transport system, allowing the vehicle to move on a pre-determined track. Various tests were conducted on bending, bending stiffness, axial pulling and full scale wear to validate operational parameters. In appendix B, the results of these tests are presented.

#### 3.2.1.3. Vertical transport system, Umbilical and SWOE return

The vertical transport systems contain pumps that transport the nodules, water and sediment to the surface. These pumps are located on every kilometre of the riser (figure 3.3). A small fraction of abrasion and fragmentation of the nodules is lost during the transport. An electrical cable supplier<sup>4</sup> has developed the umbilical system that connects the ship to the mining vehicle and delivers power, communications and lifting ability.

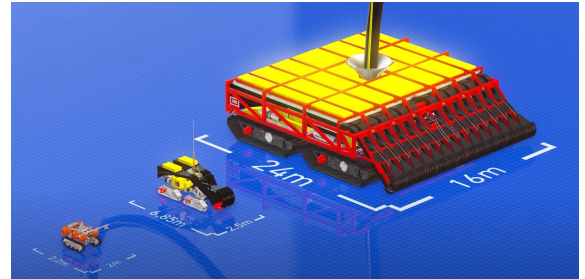


Figure 3.2: Concept design, Apollo 1, Apollo 2, and full scale

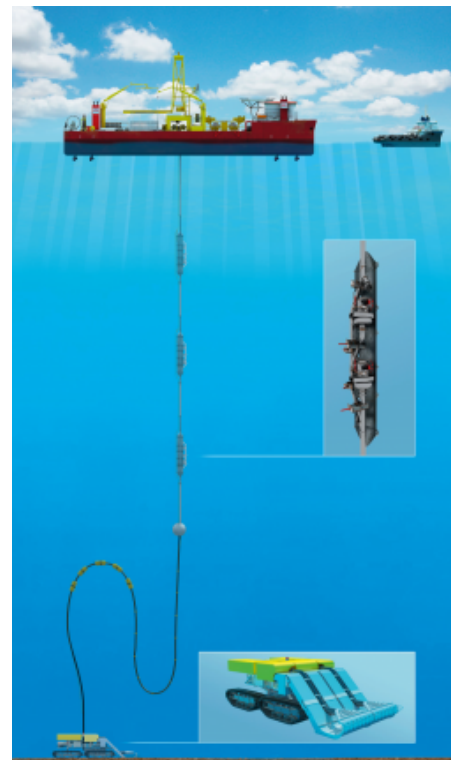


Figure 3.3: Preliminary design subsea harvesting equipment

<sup>2</sup>For more information visit [www.blue-nodules.eu](http://www.blue-nodules.eu)

<sup>3</sup>Confidential

<sup>4</sup>Confidential

### 3.2.1.4. Surface operation vessel

The material that arrives at the surface operation vessel consists of water derived near the seabed, nodules, a small number of nodule fragments and sediment not removed at the seabed. The nodules are offloaded to a bulk carrier. This material needs to be dewatered to store the nodules in a dry state. The SWOE is returned to the ocean and can form a sediment plume. It is released close to the seabed because it is estimated to have the least impact.

### 3.2.2. Initial design parameters of the subsea harvesting equipment

The subsea harvesting equipment's initial design requirements and parameters are shown in the tables below (table 3.7 and in the appendix table B.1, B.2, B.3). The parameters apply to the design of the jumper hose or to components that affect the jumper hose. These components include the SMT and the VTS because the jumper hose connects these components. Another component is the umbilical because it is secured to the jumper hose.

Parameter	Value	Unit
Jumper hose length		[m]
Height clump weight above seafloor		[m]
Full scale mining vehicle mass		[tonne]
Full scale mining vehicle submerged weight		[kN]
Full scale mining vehicle height		[m]
Vessel trailing velocity		[m/s]
Harvesting system velocity		[m/s]

Table 3.7: Initial design parameters components

### 3.2.3. Preliminary jumper hose design

*The preliminary jumper hose design tables are relocated in appendix B*

The hose supplier<sup>5</sup> designed a prototype jumper hose which is a component of the subsea harvesting equipment of the blue nodules project. A preliminary hose design was created as a base for the configuration analysis, where the critical parameters of the design were determined. Multiple models with altered hose parameters were created with specialist software, and the bending stiffness of the hose has been added to the design parameters besides the initial requirements.

The jumper hose comprises 22,86 meters (75 feet) long hose segments. The initial design parameters of these hose segments and the coupling are displayed in appendix table B.4. At the end of the segments, there is a coupling piece with a neck reinforcement, as shown in figure 3.4. The coupling part of the segment doesn't influence the shape of the jumper hose.

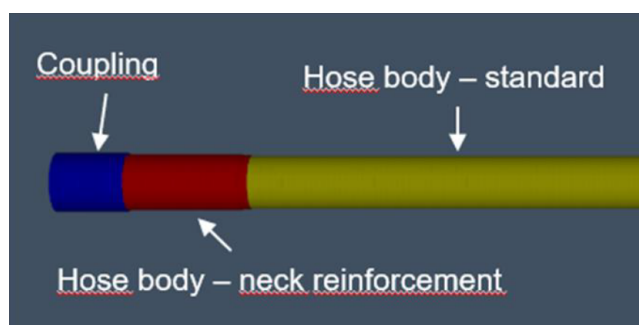


Figure 3.4: Schematic concept overview coupling hose segment

<sup>5</sup>Confidential

After optimisations on the bending stiffness and the weight of the hose segments, a final parameter set is presented in table 3.8. It was also concluded that bending stiffness is the most sensitive parameter of the jumper hose assembly. It has significant structural effects on diameter, weight, and construction cost.

Parameter	Value	Unit
Total length Jumper hose		[m]
Segment length hose		[m]
Height Clump weight from seafloor		[m]
Minimum Bending radius		[m]

Table 3.8: Parameters and Requirements Jumper hose design

### 3.2.3.1. Buoyancy modules

One of the jumper hose design requirements is that the hose should not hit or rest on the seafloor. Buoyancy modules are attached to the hose to prevent a confrontation between the jumper hose and the seafloor while the crawler is manoeuvring on a predetermined track. The parameters of these modules are displayed below:

Parameter	Value	Unit
Design water depth		[m]
Uplift per buoyancy module		[kg]
Approx. module length		[mm]
Approx. OD		[mm]

Table 3.9: Parameters buoyancy modules by the buoyancy supplier

### 3.2.3.2. Umbilical

The seafloor harvesting equipment is clamped to the jumper hose. The estimations of the umbilical are displayed in Appendix table B.2.

### 3.2.3.3. Final design of the Jumper Hose

The jumper hose comprises 13 hose segments, with a segment length of 22,86 meters. All the segments have the same construction, and on 7 segments, 8 buoyancy modules are attached. Figure 3.5 shows a sketch of a buoyant segment, and figure 3.6 shows the main dimensions of a buoyant segment. The parameters of a hose segment are displayed in appendix table B.5.

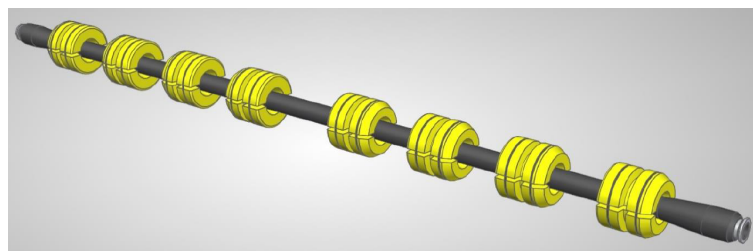


Figure 3.5: buoyant hose segment with 8 buoyancy nodules connected



### 3.3.2. *Delimited requirements of the design*

There are three types of requirements for the design of the jumper hose. The first is about the design of the jumper hose and failure

- The jumper hose is not allowed to touch or recline on the seafloor.
- The jumper hose cannot lift the crawler from the seafloor or pull over the crawler.
  - Safety factor of 2 between maximal pulling tension and the weight of the SMT.
- The jumper hose should never exceed the minimum bending radius.

The following requirements are to delimit the dimensions of the jumper hose and the new set of parameters, and these are also part of the scope of the thesis:

- For the Blue nodules project research is done by the hose supplier on the construction and material of the jumper hose. These material parameters are used for the design of the jumper hose. The structures assembly and material selection are outside the scope of this thesis.
- The environmental parameters collected during the Blue nodules project are used for the base design.
- The connection of the jumper hose with the SMT and the VTS is evaluated globally. Later in the document, the assumptions for the design of the parameters are described.
- The design of the SMT is disregarded. The parameters of the full-scale design of the crawler are used.
- The design of the VTS is disregarded and the parameters supplied by IHC are used.
- The hose parameters of table 3.9 are used for the dimensions of the hose.
- The buoyancy modules parameters of the buoyancy supplier are used.
- The construction, launch and recovery of the total seafloor harvesting system and the jumper hose are disregarded.
- The umbilical and its attachment to the jumper hose are disregarded.
- If a parameter is not addressed explicitly in the research, the estimated parameters of the initial design are used.

### 3.3.3. *Parameters, dimensions and limitations*

The following parameters, dimensions and limitations are used for the optimisation and dynamical design, table 3.10. If parameters change, it is described and specified in the relevant chapters. These changes are then summarised at the end of the chapters.



Parameter	Value	Unit
ID hose		[m]
OD hose		[mm]
Weight hose in air		[kg/m]
Weight slurry in air		[kg/m]
MBR		[m]
Axial load ability		[kN]
Bending stiffness		[kN/m <sup>2</sup> ]
Working depth		[m]
Bottom mean current		[m/s]
Sea water density		[kg/m]
Submerged weight crawler		[kN]
Velocity surface operation vessel		[m/s]
Maximum velocity crawler		[m/s]
Lift force buoyancy module		[kg]
Full scale mining vehicle height		[m]

Table 3.10: Environment en design parameters

In consultation with Royal IHC, the following parameters are used as a starting point for the Jumper hose design, table 3.11.

Parameter	Value	Unit
Length element jumper hose		[m]
Total length jumper hose		[m]
Number of elements jumper hose		[pieces]
Height clump weight above seafloor		[m]
Number of buoyancy modules		[pieces]
Modules clamped per hose segment		[pieces]
Starting point modules		[hose segment]

Table 3.11: Base set of parameter jumper hose by the hose supplier

### 3.4. Research question 1 and the related sub-questions

This chapter provides a clear answer to the first research question: *What is the preliminary base design of the flexible connection from the Blue nodules project?* The preliminary base design parameters are described in subchapter 3.2. These parameters are used for the static model and are shown in table 3.10 and 3.11.

The main design parameters which are varied for the jumper hose design exist of:

- Length of the jumper hose
- Total number of buoyancy modules
- Number of buoyancy modules per hose section
- Starting section of the buoyancy modules
- Height VTS above the seafloor
- Range of the crawler
- Content flow in the jumper hose

These parameters arise from the requirements and limitations that are described and set in sub-section 3.3.1.





# Methods and models



# Static analysis method - MatLab model

The starting point for the dynamic model starts with the construction of the static base model and is presented in this chapter. This chapter concerns the underlying theory to construct the first static model of the jumper hose, which is interpreted as a catenary cable subjected to its weight. The basic calculations like the total weight of the jumper hose and the number of buoyancy modules are carried out. Subsection 4.2.4 elaborates the implementation of buoyancy in the first static model. The final static model is constructed in chapter 6, which is the foundation of the parameter study for the jumper hose design. Finally, an answer is given to research question 2, “*How is the static model constructed, and which methodology is used?*”. Also, the sub-question “*Which simplifications and adaptations are necessary for the contemplation of the static model?*” is answered. The execution of the model is presented in chapter 6

## 4.1. Base model and description

As a visualisation of the jumper hose model, a schematic overview is drawn to appoint: the layout of the jumper hose, the boundary conditions at the connection points and other model input. The design parameters as stated in table 3.10 and 3.11. The global coordinate system is placed on top of the crawler, which has a height of 7.5 meters (figure 4.1 and table 4.1).

For the static analysis, the hose model is simplified to a simple known system. Then the model is extended by adding complex elements to approach and represent the behaviour of the hose. The connection points are modelled as nodes fixed to the global coordinate axis for the first simple known model. The hose is modelled as a limp cord without properties. The buoyancy modules are clamped on the jumper hose and are modelled as a point force in the positive z-direction. The environmental conditions are disregarded for the static analysis. These are the characteristics of the first static model of the jumper hose.

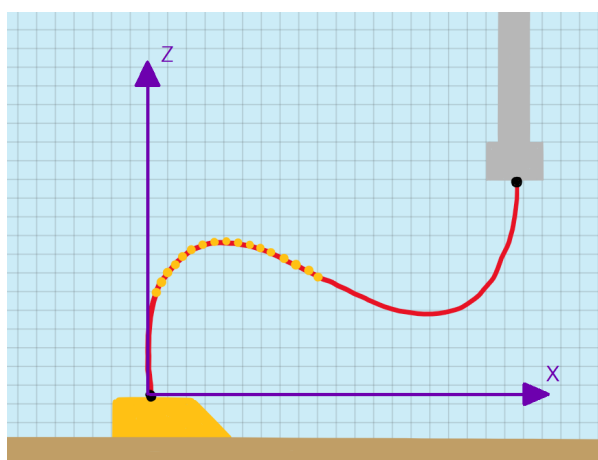


Figure 4.1: Schematic drawing subsea harvesting system

	VTS	Crawler
X	100 m	0 m
Z	62,5 m	0 m

Table 4.1: Coordinates VTS and Crawler for model

## 4.2. Modelling approach base model jumper hose

The preliminary design of the jumper hose is complicated. The hose is simplified into basic elements to assess the static behaviour and shape of the jumper hose. The static characteristics of these elements are known. A cable model does this first approach of the shape of the hose. The shape of the cable results from the mass of the cable, and one of the conceivable options is a catenary shape. This shape represents the lazy s-shape of the hose and includes the weight of the hose.

Another similar structure is a mooring line. A mooring line takes a particular shape due to the weight of the mooring line, and this shape is only influenced by connection points and parameters of the mooring line. This type of behaviour is called a catenary system. A catenary is both a physical and mathematical phenomenon.

### 4.2.1. Cable element

A cable is a structure type in mechanical engineering. Cables are flexible, and if they are compressed, they do not stay in their shape. A cable is always under tension and keeps its shape due to internal forces. The internal forces act in the direction of the cable. The tension in the cable is composed of an x and a y component. A property of a cable structure is that the horizontal x component of the tension is always constant in each element. The vertical component depends on the steepness of the cable. The steeper the cable, the higher the tension.

The following four situations are described for the model approach to model a simplified version of the jumper hose. The four cable types are:

- Cables under concentrated load
- Cables under non-uniformly distributed load
- Cables under uniformly distributed loads
- Catenary cables

#### 4.2.1.1. Cables under concentrated loads

An example of cable under concentrated loads is shown in figure 4.2. It's considered that the cable is homogeneous, flexible, non-extendible and of negligible weight. The shape is not parabolic but formed due to the concentrated loads. The cable is divided into segments that are connected by the connection points. These points are located at the concentrated loads. It's considered that the segment is straight.

From Newton's law, if the system is in equilibrium, the sum of the external forces vanishes, and the sum of their moments about any point also vanishes [30], (equation 4.1).

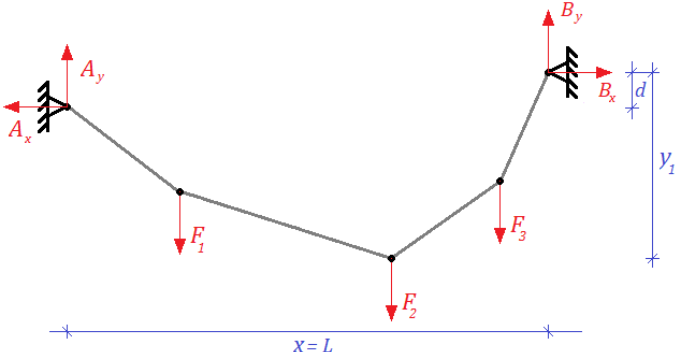
$$\begin{aligned} \sum F_x &= 0 \\ \sum F_y &= 0 \\ \sum M_{point} &= 0 \end{aligned} \quad (4.1)$$


Figure 4.2: Overview example cable under concentrated load

The reaction forces are carried by the attached points and are calculated with Newton's law (equation 4.2). Each segment is under tension and varies due to the angle of the segment. If an s is evaluated, the direction of the tension is in the next cable segment. The tension in a segment is constant over the length of the segment (figure 4.3).

$$\begin{aligned}
 \sum F_x &= 0 \rightarrow \\
 \sum F_x &= -A_x + T_x = 0 \rightarrow A_x = T_x \\
 \sum F_y &= 0 \rightarrow \\
 \sum F_y &= A_y - T_y - F_1 = 0 \rightarrow A_y = T_y + F_1 \\
 \sum M_1 &= 0 \rightarrow \\
 \sum M_1 &= A_x * y_1 - A_y * x_1 = 0
 \end{aligned}
 \tag{4.2}$$

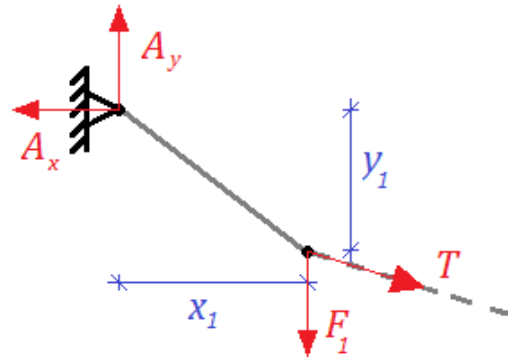


Figure 4.3: Part of the overview example cable under concentrated load

### 4.2.2. Cables under non-uniformly distributed load

Under a non-uniformly distributed load, a cable has forces that vary along the cable, figure 4.4. An important point is the lowest point of the cable because the horizontal component of the tension is the same throughout the cable and corresponds to the horizontal reaction forces at the attached points. A segment is evaluated (C to D, figure 4.5). The forcing on the segment varies and can be evaluated due to these forces. At point D, the angle between the horizontal line and the direction of the cable determines the tension (figure 4.6, equation 4.3).

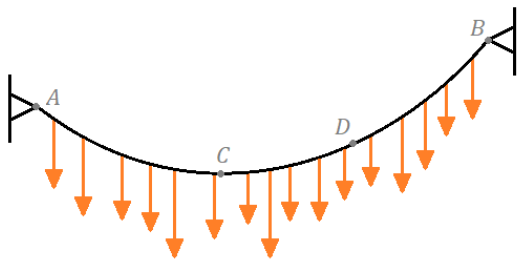


Figure 4.4: Overview example cable under non-uniform load

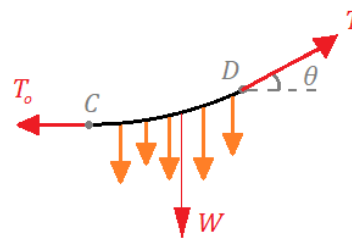


Figure 4.5: Section of the overview example cable under non-uniform load

The following relations are derived:

$$\begin{aligned}
 T &= \sqrt{T_0^2 + W^2} \\
 \tan\theta &= \frac{W}{T_0} \\
 T_0 &= T * \cos\theta \\
 W &= T * \sin\theta
 \end{aligned}
 \tag{4.3}$$

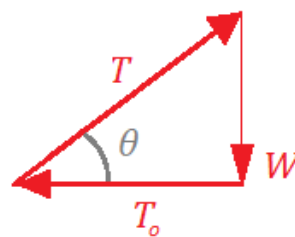


Figure 4.6: Force vectors segment of cable under non-uniform load

#### 4.2.2.1. Cable under uniformly distributed load

A cable under a uniformly distributed load is similar to a non-uniformly distributed load, figure 4.7. The only difference is in the loading, and the load is equally distributed over the cable. The same approach is applied, and a segment is evaluated (figure 4.8). The resultant load is exactly in the middle of the segment. The same relations and force triangle hold for this cable (figure 4.9).

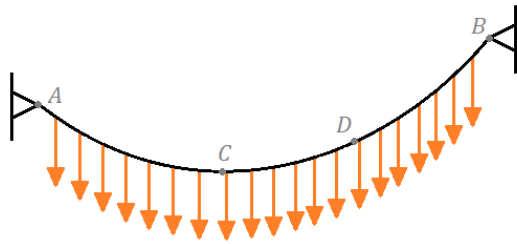


Figure 4.7: Overview example cable under uniform load

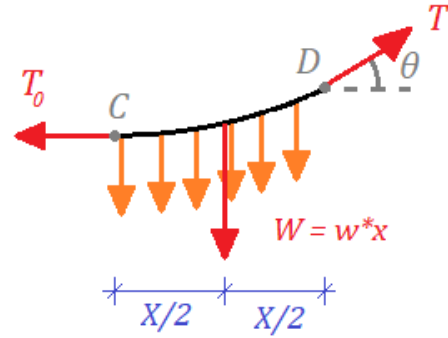


Figure 4.8: Section of the overview example cable under uniform load

The length of the cable can also be evaluated with a segment by taking a small part. The relations and moment around point D are evaluated (equation 4.4):

$$\begin{aligned}
 T &= \sqrt{T_0^2 + W^2} \\
 \tan\theta &= \frac{W}{T_0} \\
 T_0 &= T * \cos\theta \\
 W &= T * \sin\theta
 \end{aligned}
 \tag{4.4}$$

$$\begin{aligned}
 \sum M_d &= 0 \rightarrow \\
 \sum M_d &= -T_0 * y_1 + (w * x) * \frac{x}{2} = 0 \\
 y &= \left(\frac{W}{2 * T_0}\right) * x^2
 \end{aligned}$$

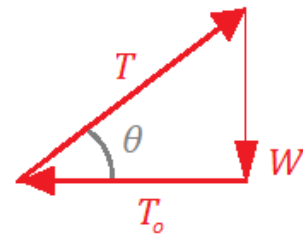


Figure 4.9: Force vectors segment of cable under non-uniform load

4.2.2.2. Catenary cable mathematic approach

The catenary is a plane curve whose shape corresponds to a hanging homogeneous flexible chain supported at its ends and sagging under the force of gravity [29]. The mathematical equation for a catenary is well-defined, equation 4.5. Where a is the wideness of the curve. The catenary passes through P1 and P2 and has a horizontal width of H. When P1 and P2 are on the same level, the vertical distance between P1 and P2 is the sag (figure 4.10). The height difference is expressed by the parameter mu, equation 4.6 and the length of the curve between P1 and P2 is expressed by parameter s, equation 4.7.

$$y = a * \cosh\left(\frac{x}{a}\right) \tag{4.5}$$

$$\mu = a * \cosh\left(\frac{x_2}{a}\right) - a * \cosh\left(\frac{x_1}{a}\right) \tag{4.6}$$

$$s = a * \sinh\left(\frac{x_2}{a}\right) - a * \sinh\left(\frac{x_1}{a}\right) \tag{4.7}$$

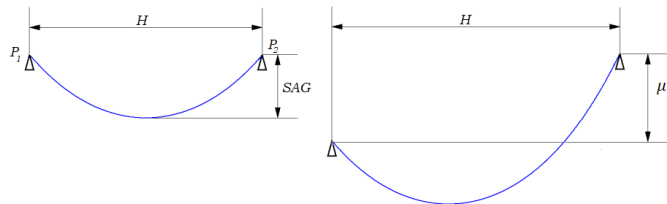


Figure 4.10: Mathematical approach catenary shape



An equation for the uniform scaling parameter 'a' can be derived with these additional parameters. This value is extracted by equating the left and the right-hand side of equation 4.8, plotting and calculating the intersection.

$$\begin{aligned}
 s^2 - v^2 &= 2a^2 \left( \cosh\left(\frac{x_2 - x_1}{a}\right) - 1 \right) \\
 &= 4a^2 \sinh^2\left(\frac{H}{2a}\right) \\
 \sqrt{s^2 - v^2} &= 2a * \sinh\left(\frac{H}{2a}\right)
 \end{aligned}
 \tag{4.8}$$

4.2.2.3. Catenary cable

In figure 4.11 the first overview shows a cable under tension. The sag is neglectable small. The weight of the cable is the weight per unit length times the length over the cable. Overview 2 represents a cable with a small sag. The cable length is approximately the same as the horizontal length, and the force is still equally distributed in the horizontal direction. The total weight is the weight per unit length times the length over the cable. With much sag, the weight per unit length gets closer in the x-direction. This shift happens because the distance of the arc is not the same as the distance in x. The total weight is not at the cable's centre but somewhere to the right. The line of action is unknown.

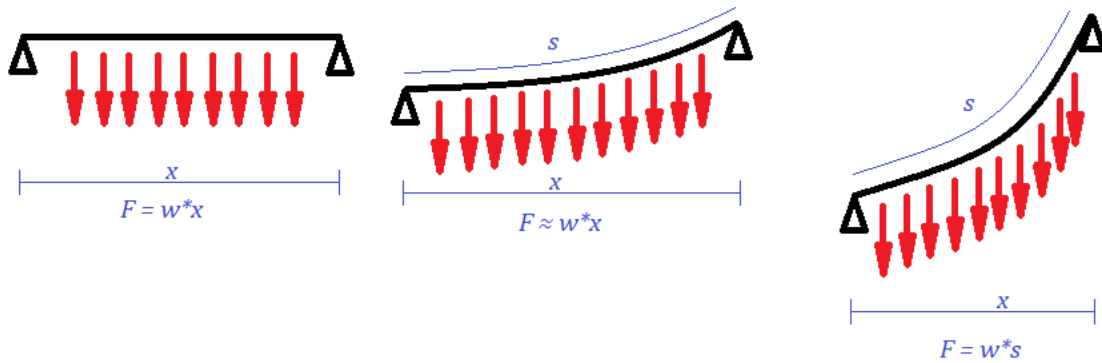


Figure 4.11: Cable evaluations

The global coordinate system is shifted to match the lowest point of the cable (figure 4.12 and 4.13). The vertical distance between the coordinate system and the lowest point is called 'c'.

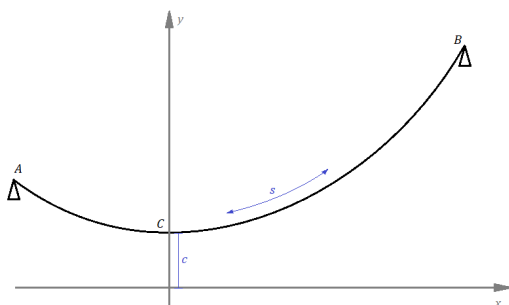


Figure 4.12: Overview example catenary

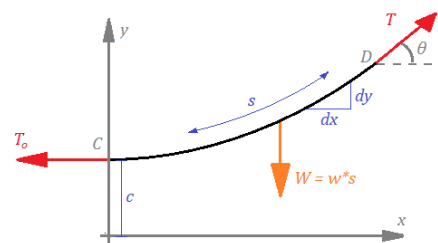


Figure 4.13: Section of the overview example catenary

Analysing a segment of the cable results in the following relations (figure 4.14 with equation 4.9 and figure 4.15 with equation 4.10):

$$\begin{aligned}
 T &= \sqrt{T_0^2 + (w * s)^2} \\
 T &= w \sqrt{\frac{T_0^2}{w^2} + s^2} \leftarrow \frac{T_0}{w} = c \\
 T &= w \sqrt{c^2 + s^2} \\
 T &= w * c \sqrt{1 + \frac{s^2}{c^2}} \\
 T &= T_0 \sqrt{1 + \frac{s^2}{c^2}}
 \end{aligned}
 \tag{4.9}$$

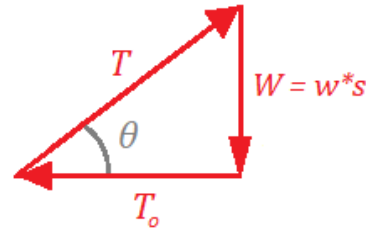


Figure 4.14: Force vectors segment of catenary

$$\begin{aligned}
 dx &= ds * \cos\theta \\
 dx &= ds * \frac{T_0}{T} \\
 dx &= ds * \frac{c * w}{T_0 \sqrt{1 + \frac{s^2}{c^2}}} \\
 dx &= ds \frac{T_0}{T_0 \sqrt{1 + \frac{s^2}{c^2}}} \\
 dx &= \frac{ds}{\sqrt{1 + \frac{s^2}{c^2}}}
 \end{aligned}
 \tag{4.10}$$

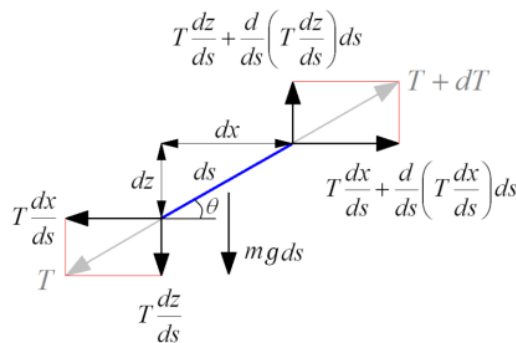


Figure 4.15: Equilibrium of an infinitesimal element of the inextensible cable under gravitational load [27]

'In general, the static equilibrium shape of a loaded cable differs significantly from that of a straight line, and the problem of determining the shape of a loaded cable is a geometrically non-linear problem' [27].

### 4.2.3. Weight of the Jumper hose

In physics, a catenary is the shape of a chain in an equilibrium formed due to the least potential energy. The first model evaluates the shape of a chain under its weight. The structure is a catenary chain with two pinned connections located at the same height (y-direction). This model is used to calculate the weight of the jumper hose and the reaction force on the connection points.

The weight of the jumper hose is calculated with the following properties of the hose (table 4.2):

- Inner diameter (ID)
- Outer diameter (OD)
- Total length of the jumper hose
- Mass of the hose substance (slurry)

The weight of the jumper hose is modelled as the gravitational force per meter hose. If there is no vertical difference between the connection points, the reaction forces are equal and half of the total weight of the hose. The following situations are evaluated. First, the catenary hose is situated in air, with water and slurry as its substance (figure 4.16 and table 4.3) . Next, the catenary hose is situated in water, with water and slurry as its substance (figure 4.17 and table 4.4).

Item	Value	Unit
Length hose		[m]
Mass hose material		[kg/m]
Mass slurry		[kg/m <sup>3</sup> ]
Mass water		[kg/m <sup>3</sup> ]
ID		[m]
OD		[m]

Table 4.2: Parameters

Force, material hose = mass hose \* g \* L  
= kN

Force, slurry = mass slurry \* area ID \* g \* L  
= kN

Force, water = mass water \* area ID \* g \* L  
= kN

Result	Value	Unit
Total weight hose material in air		[kN]
Total weight hose filled with water in air		[kN]
Total weight hose filled with slurry in air		[kN]

Table 4.3: Result own weight jumper hose in air

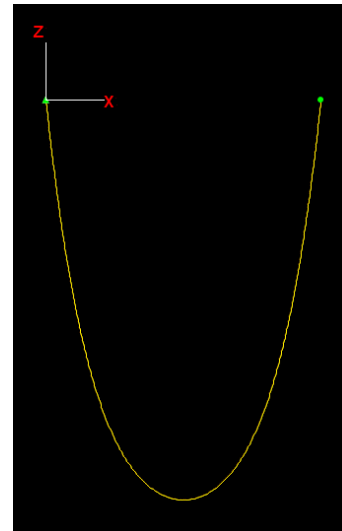


Figure 4.16: Catenary hose in air, filled with slurry or water

The physical law of buoyancy states that anybody, completely or partially submerged in fluid at rest, is acted upon by an upward or buoyant force, [6]. Putting the jumper hose in water results in a buoyancy force due to water displacement, the Archimedes principle. The buoyancy force is calculated below.

Displacement force = area OD \* density water \* g \* L  
= kN

Weight hose (water) in air - displacement force  
= kN - kN

Weight hose (slurry) in air - displacement force  
= kN - kN

Result	Value	Unit
Buoyancy force / Displacement force		[kN]
Total weight hose filled with water in water		[kN]
Total weight hose filled with slurry in water		[kN]

Table 4.4: Result weight jumper hose in water

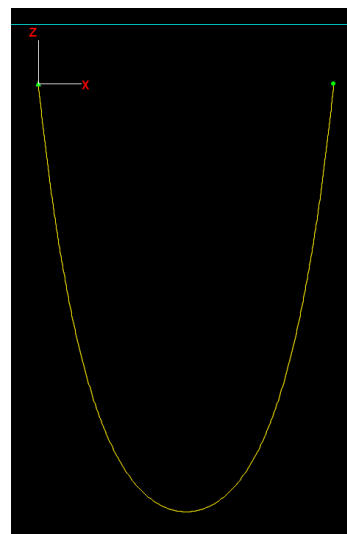


Figure 4.17: Catenary hose in water, filled with slurry or water

If the connection points are not at the same height, the force distribution is not equal but depends on the lowest point of the chain. The difference in reaction force can be calculated using the formulas of the mathematical catenary, equation 4.5 up to 4.15.

#### 4.2.4. Adding buoyancy to the model

Buoyancy modules are clamped on the jumper hose to get the desired S-shape. The hose is connected to the VTS and hangs freely, so the VTS supports a part of the weight. One buoyancy nodule can lift 429 kg of weight, so to lift the total weight of the hose in the water, there are at least 32 buoyancy modules needed. The number of buoyancy modules differs from the number presented in the preliminary design. In consultation with IHC and calculating the buoyancy force designed in the preliminary design, it is decided that the parameter 'number of buoyancy modules' is detained to **32 buoyancy modules**. This adjustment influences other parameters and the result of the preliminary design.

### 4.3. Geometrically nonlinear system

As stated in subsection 4.2.2.3, determining the shape of a loaded cable is a geometrically non-linear problem. Modelling the jumper hose by one limp cord fails to capture specific effects specified on the local mechanics. The first schematization is to simulate the jumper hose as a catenary chain. This obtains the global parameters and include the axial stiffness. The node's local internal forces and location are calculated by implementing a Finite Element Method to determine the structure's shape.

#### 4.3.1. Finite element method and geometrically nonlinear elements

The finite element method (FE) is a numerical method to solve differential equations. The basic idea is that the structure is divided into smaller and simpler elements with boundary conditions at the end. This method allows for obtaining a set of algebraic equations to be solved for specific quantities per element, which can be combined/connected to get the solution for the whole system. The basic steps for a FE method are displaced below <sup>1</sup>.

1. Discretize the system into smaller elements with boundary conditions
2. Assume a deformed shape for the total system
3. Approximate the solution by a piece-wise polynomial, defined in each element, the shape function
4. Define the elemental weak form at each element to satisfy the problem in a weak sense
5. Assemble the global system and construct the global algebraic equation
6. Apply the boundary conditions on the system

#### 4.3.2. Large deformations, static solution for a catenary model, axial extensible

For the calculation of the large deformations of the axial-extensible catenary model, a FEM approach is used. The model is divided into smaller elements. These elements have an axial stiffness, unit mass and a tensionless length. The deformation happens only axial so that the element can rotate and extend, but no shear or bending is included. These conditions for deformation correspond to a string element. A string is an idealization of a long flexible structure, which is tensioned by external forces so that the restoring force associated with this tension is much greater than that associated with the bending stiffness [24]. The string elements are connected and can rotate freely. Figure 4.18 shows an overview of a discretised string.

<sup>1</sup>Lecture 5, Geometrically non-linear systems (axially deforming strings), Introductions to computational dynamics, OE44090, 2019

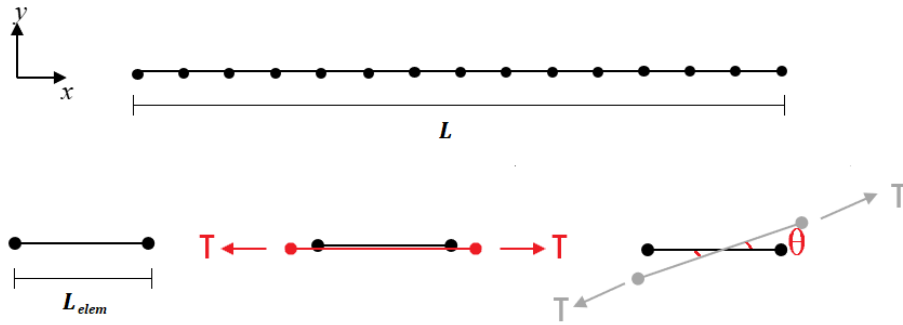


Figure 4.18: Discretizing of the system by dividing into smaller elements [14]

Assume a deformed reasonable shape that satisfies the boundary conditions, as shown in figure 4.19, the red line does not represent an axial-extensible catenary shape very well. Based on the deformed shape the extension, tension, orientation angle and internal forces can be calculated per element, using equation 4.11 to 4.15. Where  $\epsilon$  is the deformation in terms of the global coordinates,  $\theta$  is the orientation angle in terms of the global coordinates,  $T$  is the tension and  $F_{int}$  are the internal forces at the nodes.

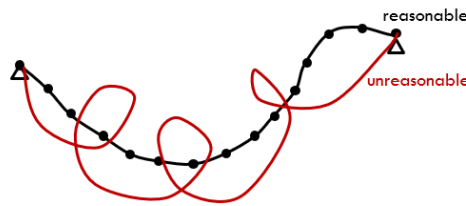


Figure 4.19: A reasonable shape [14]

$$\epsilon = \frac{\sqrt{(x_{0,r} + u_{x,r} - x_{0,l} - u_{x,l})^2 + (y_{0,r} + u_{y,r} - y_{0,l} - u_{y,l})^2} - L}{L} \tag{4.11}$$

$$\theta = \tan^{-1} \left[ \frac{y_{0,r} + u_{y,r} - y_{0,l} - u_{y,l}}{x_{0,r} + u_{x,r} - x_{0,l} - u_{x,l}} \right] \tag{4.12}$$

$$T = EA\epsilon \tag{4.13}$$

$$F_{x,l} = -T \cos \theta \qquad F_{x,r} = T \cos \theta \tag{4.14}$$

$$F_{y,l} = -T \sin \theta \qquad F_{y,r} = T \sin \theta \tag{4.15}$$

After integration, the contribution of the internal and external forces needs to balance at the nodes. For each node, a force balance is constructed. In equation 4.16 a force balance is composed of the node that connects element 1 to element 2 and also the node that connects element 2 to element 3 (figure 4.20). External force  $P$  needs to be equal to the sum of the tensions of both elements.

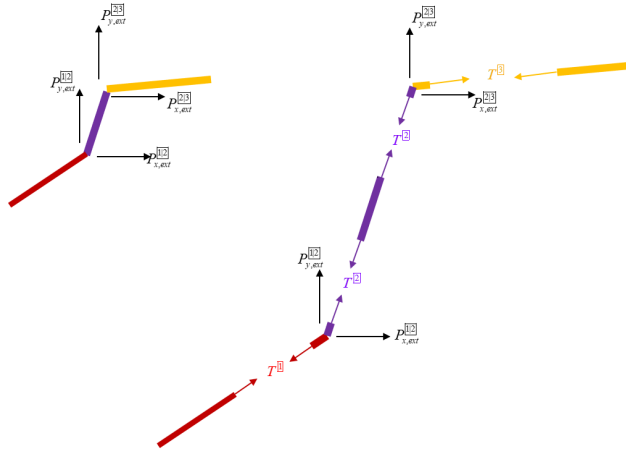


Figure 4.20: Balancing of forces in 2 nodes [14]

$$\begin{aligned}
 P_{x,ext}^{[1|2]} &= T^{[1]}\cos\theta^{[1]} - T^{[2]}\cos\theta^{[2]} \\
 P_{y,ext}^{[1|2]} &= T^{[1]}\sin\theta^{[1]} - T^{[2]}\sin\theta^{[2]} \\
 P_{x,ext}^{[2|3]} &= T^{[2]}\cos\theta^{[2]} - T^{[3]}\cos\theta^{[3]} \\
 P_{y,ext}^{[2|3]} &= T^{[2]}\sin\theta^{[2]} - T^{[3]}\sin\theta^{[3]}
 \end{aligned} \quad (4.16)$$

The force balances from equation 4.16 are written in vectorial form (equation 4.17) where  $\mathbf{P}$  is the external force,  $T$  the tension and  $\mathbf{t}$  the rotation vector. This results in the system of equations that need to satisfy the initial conditions. And results in the formulation of the weak form of the system. The weak form is presented in equation 4.18, and this is a non-linear dependency.

$$\begin{bmatrix} \vdots \\ \mathbf{P}_{ext}^{[i|j]} \\ \vdots \end{bmatrix} = \begin{bmatrix} \vdots \\ T^{[i]}\mathbf{t}^{[i]} \\ \vdots \end{bmatrix} - \begin{bmatrix} \vdots \\ T^{[i]}\mathbf{t}^{[i]} \\ \vdots \end{bmatrix} \quad \mathbf{P}_{ext}^{[i|j]} = \begin{bmatrix} P_{x,ext}^{[i|j]} \\ P_{y,ext}^{[i|j]} \end{bmatrix} \quad \mathbf{t}^{[i]} = \begin{bmatrix} \cos\theta^{[i]} \\ \sin\theta^{[i]} \end{bmatrix} \quad (4.17)$$

$$\begin{bmatrix} \vdots \\ \mathbf{P}_{ext}^{[i|j]} \\ \vdots \end{bmatrix} = \begin{bmatrix} \vdots \\ T^{[i]}\mathbf{t}^{[i]} \\ \vdots \end{bmatrix} - \begin{bmatrix} \vdots \\ T^{[i]}\mathbf{t}^{[i]} \\ \vdots \end{bmatrix} \rightarrow \begin{bmatrix} \vdots \\ \mathbf{P}_{ext}^{[i|j]} \\ \vdots \end{bmatrix} - \begin{bmatrix} \vdots \\ T^{[i]}\mathbf{t}^{[i]} \\ \vdots \end{bmatrix} + \begin{bmatrix} \vdots \\ T^{[i]}\mathbf{t}^{[i]} \\ \vdots \end{bmatrix} = 0 \rightarrow \mathbf{P}_{ext} - \mathbf{f}_{int}(\mathbf{x}_0 + u) = 0 \quad (4.18)$$

The residual of the non-linear system is linearized with a Taylor Series Expansion (TSE) to solve the system (equation 4.19).

$$\mathbf{P}_{ext} - \mathbf{f}_{int}(\mathbf{x}_0 + u) = \mathbf{R} \quad (4.19)$$

Apply TSE so the residual can be approximated in a given linearized point  $u^*$  plus the derivative of the residual time  $u - u^*$  (equation 4.20).

$$\begin{aligned}
 \mathbf{R}(u) &\approx \mathbf{R}(u^*) + \frac{\partial \mathbf{R}(u^*)}{\partial u} (u - u^*) = \mathbf{F}_{ext} - \mathbf{F}_{int}(\mathbf{x}_0 + u^*) - \mathbf{K}(u^*)\delta u \\
 &= \mathbf{R}(u^*) - \mathbf{K}(u^*)\delta u
 \end{aligned} \quad (4.20)$$

Where:

- $\mathbf{K}u^*$  is derivative of  $\mathbf{F}_{int}(\mathbf{x}_0 + u)$  concerning the FEM coefficients  $u$
- $u$  is the vector of coefficients vector of DOFS
- $\mathbf{K}u^*$  is the linearized stiffness matrix

To make the residual equal to 0, the Newton-Raphson method is employed to take an initial guess  $u^0$  that satisfies the boundary conditions and iterate it until the residuals are below a certain tolerance.

The Newton Raphson method is implemented, a numerical iteration method that determines the roots or zeros (the residual to be 0). The mathematical implementation of the Newton Raphson method is presented below from equation 4.21 to 4.27. The iteration exits of the following two steps:

- Starting by selecting an initial guess that satisfies the boundary conditions
- Iterate until the residual force is less than a certain tolerance

$$\text{Residual approximation} \quad \mathbf{R}(\mathbf{u}) \approx \mathbf{R}(u^*) - \mathbf{K}(u^*)\delta\mathbf{u} \quad (4.21)$$

$$\mathbf{K}(u^{i-1})\delta\mathbf{u}^i = \mathbf{R}(u^{i-1}) \quad (4.22)$$

$$\delta\mathbf{u}^i = \mathbf{K}^{-1}\mathbf{R} \quad (4.23)$$

$$\text{Compute the increment :} \quad \delta\mathbf{u}^i = \mathbf{K}(u^{i-1})^{-1}\mathbf{R}(u^{i-1}) \quad (4.24)$$

$$\text{Update nodal position} \quad \mathbf{u}^i = \mathbf{u}^{i-1} + \delta\mathbf{u}^i \quad (4.25)$$

$$\text{Compute new residual :} \quad \mathbf{R}(\mathbf{u}^i) = \mathbf{F}_{ext} - \mathbf{F}_{int}(\mathbf{x}_0 + \mathbf{u}^i) \quad (4.26)$$

$$\text{Check tolerance :} \quad \mathbf{R}(\mathbf{u}^i) > \text{toll} \quad (4.27)$$

#### 4.3.2.1. Linearized stiffness matrix

As stated in subsection 4.3.2, it's assumed that each string can only extend and rotate, the shear and bending forces disregarded. Thus the local stiffness matrix of each string element corresponds to the coupled matrix of a rod plus a string<sup>2</sup>. The linearized local stiffness matrix of an element is displayed in equation 4.29 where  $\mathbf{K}$  is the local stiffness matrix of a rod plus string element and  $\mathbf{T}$  is the rotation matrix (equation 4.28).

$$\mathbf{K} = \frac{1}{L} \begin{bmatrix} EA & 0 & -EA & 0 \\ 0 & T & 0 & -T \\ -EA & 0 & EA & 0 \\ 0 & -T & 0 & T \end{bmatrix} \quad \mathbf{T} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & \cos\theta & -\sin\theta \\ 0 & 0 & \sin\theta & \cos\theta \end{bmatrix} \quad (4.28)$$

$$\tilde{\mathbf{K}} = \mathbf{T} \mathbf{K} \mathbf{T}^T \quad (4.29)$$

The global linearized stiffness matrix of the total system can be composed of the element's local linearised stiffens matrices (equation 4.30). By respecting the boundary conditions the global linearized stiffness matrix is assembled as shown in figure 4.21 and 4.22.

$$\tilde{\mathbf{K}}_{ij} \mathbf{u}_j = \begin{bmatrix} \tilde{K}_{ii} & \tilde{K}_{ij} \\ \tilde{K}_{ji} & \tilde{K}_{jj} \end{bmatrix} \begin{bmatrix} u_i \\ u_j \end{bmatrix} \quad (4.30)$$

<sup>2</sup>Lecture 6.1, Geometrically non-linear systems (axial deforming strings), Introduction to computational dynamics, OE44090, 2020

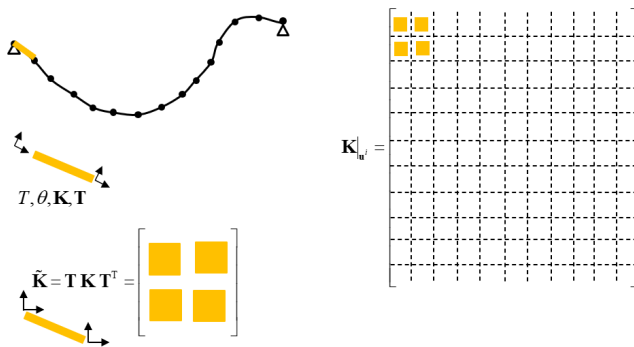


Figure 4.21: Implementing local stiffness element in global stiffness system [14]

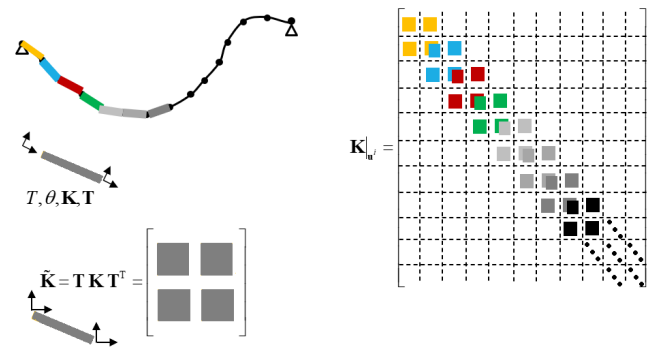


Figure 4.22: Implementing local stiffness element in global stiffness system for all elements [14]

## 4.4. Research question 2 and the related sub-question

This chapter clarifies the second research question: *How is the static model constructed, and which methodology is used?*. The base static model is a hose model subjected to its weight and composed of geometrically non linear elements. The construction of the base model is developed in chapter 6.

The sub-question, *Which simplifications and adaptations are necessary for the contemplation of the static model?* is partially answered in this chapter. The following parameters are disregarded:

- Bending stiffness
- Bending deformation
- Shear deformation
- A continuous hose
- Resistance against rotation at the connection points

The static model is simplified by:

- The structure is divided into smaller and simpler elements with boundary conditions
- The elements have an axial stiffness, unit mass and tensionless length
- Deformation of the elements only happens axial (rotation and extension)
- The elements behave as string elements
- The string elements are connected and can rotate freely around the connection points

In chapter 6, a full answer is given to this sub-question.



# Mathematical software model - OrcaFlex

The aim of this thesis is a dynamic analysis of the jumper hose. The Numerical Matlab model is static. The current chapter links the numerical Matlab model and the upcoming OrcaFlex model. It also provides the OrcaFlex software's theoretical background, specifying the objects used, environmental data, static analysis and dynamic analysis. Finally, an answer is given to research question 4, *What is the theory behind the dynamic software used and does it connect with static analysis?*.

## 5.1. The software to enhance the static MatLab model

In subchapter 7.5 the parameters for the concept design are selected by simulating the jumper hose design in MatLab. This analytical hose design is created with geometrically non-linear elements and iterates to a static equilibrium for given parameters. This model is a simplified representation of reality and it disregards some of the initial design parameters like bending stiffness. These parameters contribute to the overall performance of the hose. The aim of this thesis is a dynamic analysis of the jumper hose. There is a trade-off made between extending the analytical MatLab model and building a new model in a software program called OrcaFlex.

*OrcaFlex is a software program for static and dynamic analysis for offshore systems like risers, vessels, buoys and other structures. It is a fully 3D non-linear time or frequency domain software package that uses finite element and lumped mass element methods to analyze implemented systems' performance and physical behaviour. OrcaFlex, developed by Orcina, displays its result in a user-friendly graphic environment that meets industry standards. It models desired objects like vessels, hoses, mooring lines, risers, wind turbines, etc. It can also perform given scenarios or simulations like system towing, installation operations, disconnections, etc. The software can also implement different loading cases like environmental conditions and external forces.*

The decision was made to model the next phase of the jumper hose in OrcaFlex because it has a wide range of static and dynamic analysis implementations. It is continuously verified and tested by a team of specialists. Using this software, an extension of the static model is built and used as a starting configuration for dynamic analysis. The numerical MatLab model lacks a lot of design and environmental parameters and simplifies the design parameters. The analytical Matlab model is used for the concept parameter design and the OrcaFlex model is used to elaborate and evaluate the conceptual parameters of the design. This twofold is that the jumper hose is a complicated design, making the exact calculation complex. With the analytical Matlab model, a range of design parameters is tested and graded, resulting in graded parameter combinations, where the best combination can be selected. The simplified calculations are a lot faster than the complex ones, which results in a faster selection process.

### 5.1.1. Orcaflex program approach

OrcaFlex builds a mathematical model built up from a series of interconnected objects. It can't exactly represent every aspect of a real-world system because that would require infinite data and computational time. By choosing which important system features, the system is simplified and still gives insight into the mathematical model's static and dynamic behaviour. "OrcaFlex offers a variety of analysis" [18]:

- Static analysis, in which OrcaFlex calculates the static equilibrium position of the model; current and wind loads are included, but not wave loads.
- Modal analysis, in which OrcaFlex calculates and reports the undamped natural modes of the model, or of an individual line in the model.
- Time-domain dynamic analysis, in which OrcaFlex carries out a time simulation of the response of the system to waves, current and a range of user-defined inputs. A choice of implicit and explicit integration scheme is offered.
- Frequency-domain dynamic analysis, in which OrcaFlex carries out linear frequency domain analysis of the response of the system to waves, current and a range of user-defined inputs..
- Restart analysis, in which OrcaFlex restarts an from an existing analysis after modifying its data.
- Fatigue analysis, in which OrcaFlex calculates and collates fatigue damage.

## 5.2. Orcaflex theoretical background, objects and data

*The theoretical background, object information and calculation methods are obtained on OrcaFlex manual, [18].*

An Orcaflex system consists of two main components. The first components are the objects which represent the mathematical model of the desired system. These are built with objects like lines, shapes, winches, etc. and the second component is the environmental data that the system is subjected to, like waves, currents, seabed etc.

The objects of OrcaFlex are:

- Vessel
- Line
- Buoy
- Winch
- Link
- Shape
- Constrained

The following subsections discuss the governing model objects.

### 5.2.1. Line object and theory

“Lines are catenary elements used to represent pipes, flexible hoses, cables, mooring lines, etc. Lines are represented in OrcaFlex by a lumped mass model. That is, the line is modelled as a series of ‘lumps’ of mass joined together by massless springs, rather like beads on a necklace. The lumps of mass are called nodes and the springs joining them are called segments. Each segment represents a short piece of the line, whose properties (mass, buoyancy, drag etc.) have been lumped, for modelling purposes, at the nodes at its ends” [21].

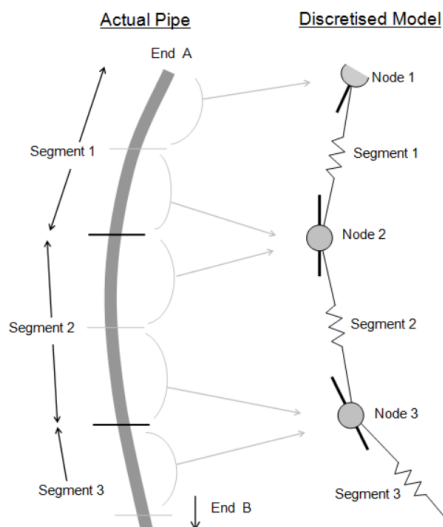


Figure 5.1: Line object model OrcaFlex, discretised model [21]

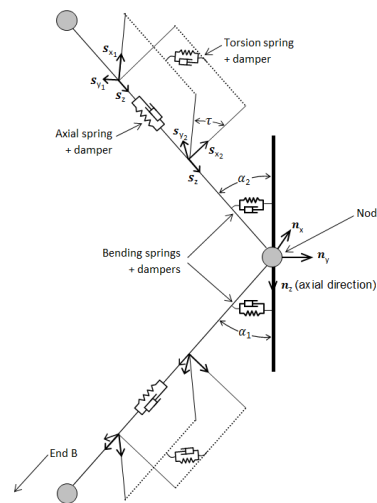


Figure 5.2: Line object model OrcaFlex, structural model [21]

### 5.2.1.1. Line structural model

Figure 5.1 gives a global overview of the nodes and segments. It shows which part of a segment is lumped into a node to get the discretized model. Figure 5.2 gives a more detailed overview of the modelling of the segments. “These include various spring-dampers to model the structural properties of the line” [21].

- The axial spring-damper models the axial stiffness and damping of the line. This spring-damper results in an equal and opposite effective tension force to the nodes. These forces are located at the end of the segment.
- The rotational spring-damper models the bending properties of the line.
- The torsional spring-damper models the torsional stiffness and damping. This spring-damper results in an equal and opposite torque moment to the nodes. These moments are located at the end of the segment. Torsion is optional to include in the line properties.

”For the forces and moments on the mid-node, OrcaFlex calculates in five stages” [21].

- Tension forces
- Bend moment
- Shear forces
- Torsion moment

The in-depth explanation of these forces and moments is described in Appendix D.1. “The total force and moment on each node combine the calculation stages and other non-structural loads like weight, drag, added mass etc.” [21].

### 5.2.1.2. Line content flow

It’s possible to specify the content flow for pipes. “If contents flow effects are not significant and may be neglected, the flow rate is zero. Flow effects can be significant for pipes carrying high-density contents at rapid flow rates and can’t be ignored” [21]. Three forces contribute to contents flow:

- Centrifugal
- Coriolis
- Flow friction

The in-depth explanation of these forces are described in Appendix D.2

The content can be modelled as:

- Uniform content, the entire line is filled with contents of uniform density
- Free-flooding, line filled with seawater
- Slug flow, spatial and temporal variation of content

The in-depth explanation of modelling of the content is described in Appendix D.2.

### 5.2.1.3. Hydrodynamic and aerodynamic loads, Drag

"Hydrodynamic and aerodynamic drag forces are applied to the line, represented by the drag term in Morison's equation. The same drag formulation is used for hydrodynamic and aerodynamic drag forces" [21]. The Morison equation and the extended form used by OrcaFlex is described in Appendix D.3.

Drag force formula:

$$\begin{aligned} f_{Dx} &= \frac{1}{2} p \rho d_n l C_{Dx} v_x |\mathbf{v}_n| \\ f_{Dy} &= \frac{1}{2} p \rho d_n l C_{Dy} v_y |\mathbf{v}_n| \\ f_{Dz} &= \frac{1}{2} p \rho \pi d_a l C_{Dz} v_z |v_z| \end{aligned} \quad (5.1)$$

Where,

- $\rho$  = fluid density
- $p$  = proportion wet or dry
- $D_n$  = diameter normal to line
- $D_a$  = diameter line
- $C_{Dx}, C_{Dy}, C_{Dz}$  = drag coefficients
- $\mathbf{v}$  = velocity in direction

The drag forces applied to a line are calculated using the cross-flow principle. Hoerner refers to the cross-flow principle, "At an angle of attack,  $\alpha$  flow pattern and fluid-dynamic pressure forces of such bodies only correspond to the velocity component (and the dynamic pressure) in the direction normal to their axis" [9]. "The fluid velocity is relative to the line  $\mathbf{v}_r$  is split into its components  $\mathbf{v}_n$  and  $\mathbf{v}_z$  normal and parallel to the line axis. The drag force normal to the line axis is then determined by  $\mathbf{v}_n$  and its x- and y-components  $\mathbf{v}_x, \mathbf{v}_y$ ; the drag force parallel to the line axis is determined by  $\mathbf{v}_z$ .

OrcaFlex offers a choice of formulations for the drag force  $\mathbf{f}_D$ . These differ in how the drag force components vary with the incidence angle  $\phi$  between the flow and the line axial direction. Some methods are more suited to particular configurations than others: they are reviewed in Casarella and Parsons. That the drag force can vary with the incident angle is also taken into adopted in OrcaFlex" [21].

### 5.2.1.4. Hydrodynamic and aerodynamic loads, Lift

Hydrodynamic lift is formulated as:

$$f_l = p |\mathbf{u}_n \times \mathbf{u}_z| \frac{1}{2} \rho d_n l C_L |v_t|^2 \mathbf{u}_L \quad (5.2)$$

Where,

$\mathbf{u}_n$  = unit vector in seabed outward normal direction

$\mathbf{u}_z$  = unit vector in z-direction of node

$\mathbf{u}_L$  = unit vector in lift force direction

$\mathbf{u}_t$  = unit vector in flow direction of lift purposes

$C_L$  = lift coefficient

$v_t$  = component of  $v_r$  in transverse direction  $\mathbf{u}_t$

### 5.2.2. Line Attachment buoyancy modules

In the line object, it's possible to model attachments connected to a line—examples like a clump weight or a buoyancy module. Orcaflex has the following attachment type: clump types, drag chain types, flex joint types and stiffener types.

#### 5.2.2.1. Attachment, Clump types

“Clumps are concentrated attachments connected to a node on a line object. It represents a small body that experiences forces like weight, buoyancy, drag, etc., as a 3D buoy. A 3D buoy object is a simplified point element with 3 DOF (degrees of freedom), **X**, **Y** and **Z**. The clump is constrained to move with the attached node and applies a buoyancy and hydrodynamic force on the connected node. The clump applies no moment to the node” [21].

The drag force of the clump is calculated in the three translations:

$$\begin{aligned} f_{Dx} &= -p_w \frac{1}{2} \rho C_{Dx} A_x v_x |\mathbf{v}| \\ f_{Dy} &= -p_w \frac{1}{2} \rho C_{Dy} A_y v_y |\mathbf{v}| \\ f_{Dz} &= -p_w \frac{1}{2} \rho C_{Dz} A_z v_z |\mathbf{v}| \end{aligned} \quad (5.3)$$

Where,

$p_w$  = clumps proportion wet

$\rho$  = water density

$C_a$  = drag coefficient component

$A$  = drag area component

$\mathbf{v}$  = clump velocity relative to fluid velocity

The added mass force of each clump is calculated by:

$$f_a = p_w \rho C_a V \quad (5.4)$$

Where,

$C_a$  = corresponding component of the given added mass coefficient

$V$  = Volume of clump

#### 5.2.2.2. Attachment, drag chain

“A drag chain is a straight-chain that hangs down from the line. These chains apply a weight, buoyancy and drag force to the node to which they are attached, but no does not include added mass effects” [21].

#### 5.2.2.3. Attachment, flex joint

“A flex joint attachment allows for additional bending stiffness to a particular node. It can add stiffness to the node. But it can also remove the stiffness from a node, creating a pin-joint” [21].

#### 5.2.2.4. Attachment, stiffeners

"A stiffener is an attachment that could be spread over more than one node. It usually models a bend stiffener or restrictor. The stiffener matches the line properties, unlike a flex joint" [21].

#### 5.2.3. Constraint object

A constraint object provides an enhanced, versatile means of connecting objects. It has no physical properties. They can fix individual DOFs, introduce individual DOFs or impose displacements on individual DOFs. It has an in-frame reference and an out-frame reference. The in-frame is rigidly connected to a parent object and translates and rotates with this object. The out frame represents the child object, so the child object translates and rotates with the constraint.

#### 5.2.4. Environmental data

The second component of the OrcaFlex model is the environmental data of the system. OrcaFlex incorporates waves, currents, seabed interaction etc., in its dynamic calculations. Environmental data is implemented under the Environment tab of the OrcaFlex model. For example, the sea tab sets the surface height, kinematic viscosity, temperature and the type of Reynolds number calculation. Figure 5.3 shows the schematic overview of the environment.

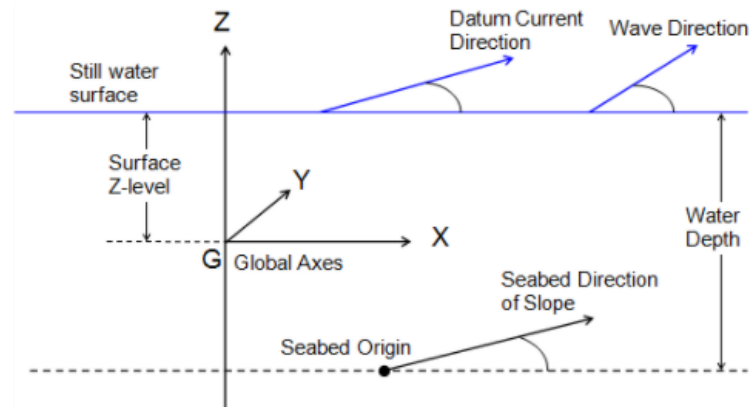


Figure 5.3: Overview environment in OrcaFlex [20]

#### 5.2.5. Sea data

##### Reynolds number

The sea data is specified under the environmental data of OrcaFlex. There are three options for the Reynolds number calculations: nominal, cross flow and flow direction. These options are based on the relative flow velocity and the characteristic length (table 5.1).

The Reynolds number is defined as,

$$Re = \frac{vl}{\nu} \quad (5.5)$$

Where,

$v$  = velocity characteristic

$l$  = length characteristic

$\nu$  = the kinematic viscosity

	Nominal	Cross-flow	Flow direction
Velocity characteristic	$v =  v_r $	$v =  v_n $	$v =  v_r $
Length characteristic	$l = d$	$l = d$	$l = d/\cos\alpha$
Reynolds number	$Re_{nom} = \frac{ v_r d}{\nu}$	$Re_{cross} = \frac{ v_r d\cos(\alpha)}{\nu}$	$Re_{flow} = \frac{ v_r d}{\nu\cos(\alpha)}$

Table 5.1: Velocity and length characteristics for Reynolds number

### Sea density

It's possible to vary the sea density with the depth. There are three options in OrcaFlex for this calculation. The first option is a constant sea density. The second option is to vary the sea density with the bulk modulus due to the compressibility of water. The last option is to linearly interpolate the sea density over the depth.

#### 5.2.5.1. Seabed

The seabed is a boundary condition of the model. It must withstand an object from passing through or act as an anchorage. This boundary condition is modelled as a "reaction force composed of the resistance force of the seabed in the normal direction and a friction force direction tangential to the seabed plane and towards the friction target position" [20].

There are two seabed models available in OrcaFlex, elastic and non-linear soil models. "The elastic model behaves as a simple elastic spring in the direction of the normal and the tangential to the seabed plane. The non-linear soil model includes the non-linear hysteric behaviour of the seabed soil in the normal direction" [20].

#### 5.2.5.2. Current

The current can vary vertically and is specified by one of two methods.

##### Interpolated method

The horizontal current for a discrete depth is set in the environmental data part of the OrcaFlex model. "The values for intermediate depths are obtained by linear interpolation. The profile should be specified from the still water surface to the seabed; if the data do not cover the full depth, then extrapolation is applied" [20].

Vertical stretching is applied, which calculates the current speed and direction at the origin,

$$Z_s = Z_o + \lambda(Z - Z_b) \quad (5.6)$$

Where,

$Z_o$  = z coordinate seabed origin

$$\lambda = \frac{D_o}{D_b}$$

$D_o$  = water depth at seabed origin, still water level

$D_b$  = water depth at (X,Y), measured from the still water level

$Z_b$  = z-coordinate of seabed below (X,Y)

The current speed and direction are evaluated from the interpolation data.

##### Power law method

The current direction is fixed and does not vary with depth.

$$S = S_b + (S_f - S_b) \left[ \frac{Z - Z_b}{Z_f - Z_b} \right]^{\frac{1}{p}} \quad (5.7)$$

Where,

$S_f, S_b$  = current speed at surface and seabed

$p$  = the power law exponent

$Z_f$  = z coordinate still water level

$Z_b$  = z coordinate of seabed

### 5.2.5.3. Buoyancy variation with depth

"The buoyancy of an object is normally assumed to be constant and not vary significantly with position" [20].

$$F_{buoy} = \rho V g \quad (5.8)$$

Where,

$\rho$  = water density

$V$  = Volume of the object

$g$  = acceleration due to gravity

"In reality, buoyancy does vary due to the following effects:" [20]

- If the object is compressible, its volume  $V$  reduces with depth due to the increasing pressure.
- The water density  $\rho$  can vary with position, due to the compressibility of the water or from temperature or salinity variations. Normally the density increases with depth since otherwise, the water column would be unstable (the lower density water below would rise through the higher density water above).

This buoyancy can be implemented in OrcaFlex by setting the bulk modules on for object or attachment (option in OrcaFlex). The bulk modules govern how the object's volume changes with pressure.

$$V = V_0 \left(1 - \frac{P}{B}\right) \quad (5.9)$$

Where,

$P$  = pressure excess over atmospheric pressure

$V_0$  = Uncompressed volume atmospheric pressure

$B$  = Bulk modulus

## 5.3. Orcaflex theoretical background, Statics

Subsection 5.1.1 says that once the mathematical model is built, a variety of analyses can be performed. The solver moves through an order of states starting at calculating statics, statics complete, calculating dynamics, dynamics complete as shown in figure 5.4.

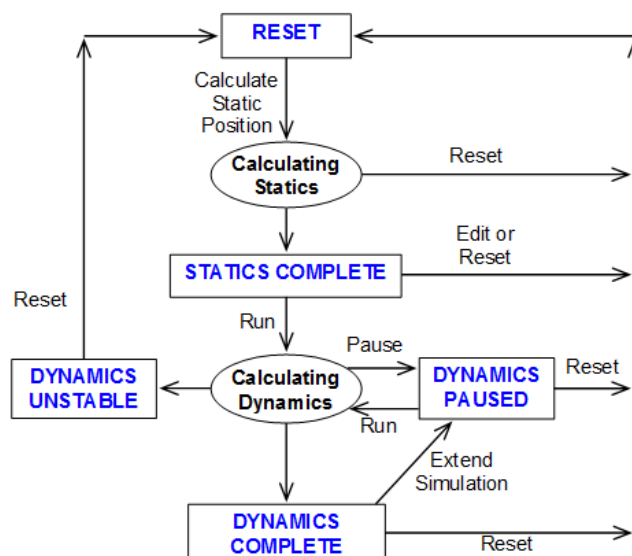


Figure 5.4: Overview calculation stages OrcaFlex [22]



The goal of the first state is to perform the static calculation of the model and is to find positions and orientations for each element in the model such that all forces and moments are in equilibrium [22]. The static calculation is used as a starting configuration for the dynamic simulation. “If the system were linear, then the equilibrium configuration could be calculated directly with a single matrix solve; in practice, however, OrcaFlex models are invariably non-linear and so calculating statics requires an iterative approach using the multi-dimensional form of Newton’s method. The iterative stages OrcaFlex goes through are” [22].

1. Fix the degrees of freedom (DOFs) of all objects (e.g. buoys, vessels, constraints etc.) other than lines
2. Calculate the line statics to determine the equilibrium configuration for all the lines (itself a two-step iterative process)
3. Release all DOFs and perform a whole system statics analysis for the entire system using Newton’s method, with the initial guess for the iteration coming from the previous two stages

### 5.3.1. Line Statics

The line statics is the second stage of the overall static calculation, after fixing all the DOF of the objects. This stage aims to provide a good starting position for the whole system statics calculation. It provides an equilibrium configuration for all lines while other objects are fixed in space and time. “Line statics is itself a two-step process, for much the same reasons: step 1 of line statics is aimed largely at providing a good starting position for the iterative step 2 calculation” [22].

#### 5.3.1.1. Line statics, step 1 methods

The first step of the line statics aims to provide a good starting position for the iterative calculation step. It’s possible to select between 5 methods,

- quick
- analytic
- catenary
- spline
- prescribed

The selection of the method depends on the system’s requirements and which effects are important. The methods are described below from simplest to more complex. “The quick method provides a simple approximation that leaves the line in an approximate catenary shape but neglects many effects that may be important, such as seabed contact and buoyancy” [22]. It’s a fast and robust method and results in a crude catenary shape.

“The analytic catenary method representation is primarily designed to facilitate quasi-dynamic mooring analysis in OrcaFlex, in which the mooring line loads are calculated from analytic catenary equations” [22]. These equations take the weight, buoyancy, contents and axial stiffness into account, but the inertia, drag, added mass, torsional stiffness and bend stiffness of the line is neglected. It also has limitations on the type of attachments of the line, attachments other than clumps are not available

“The analytic catenary representation works by solving a set of simple equations to calculate the force applied by the top end of the line to whatever object it is connected to. These equations consider the line a single continuous object; no discretization into constituent nodes is required. The analytic catenary representation does not account for dynamic effects: for a given set of input parameters, the equations predict a purely static configuration of the line” [22].

The catenary method finds the line’s equilibrium catenary position, including weight, buoyancy, axial elasticity and drag. It does not include bend stiffness or interaction with shapes. “The catenary algorithm is robust and efficient for most realistic cases, but it cannot handle cases where the line is in compression” [22].

The spline method is used when the catenary does not find an equilibrium position and this is mostly for lines with slack or neutrally buoyant. “The spline method gives the line an initial shape that is based on a user-defined smooth Bezier spline curve” [22]. The true equilibrium is then calculated in the second step of the line statics.

“The prescribed method is most suitable for pull-in analysis: you define the starting position of the line as a sequence of straight line or curved sections on the seabed” [22].

#### 5.3.1.2. Line statics, step 2

The second step of the line statics takes the first step’s results as starting position for the line. “The full statics calculation is a line statics calculation that includes all forces modelled in OrcaFlex. In particular, it includes the effects of bend stiffness and interaction with shapes” [22].

#### 5.3.2. Whole system statics

The last step of the static calculation releases all the DOFs of the remaining objects and performs a static analysis for the entire system using Newton’s method.

## 5.4. OrcaFlex theoretical background, dynamic analysis

When the static analysis is completed, the dynamic analysis is executed. The dynamic analysis consists of implementing and executing the environmental and operational loads over a specified period and analyzing the system’s dynamic response. These loads influence the motions of the objects. “Orcaflex offers two approaches for the dynamic analysis: frequency domain and time domain analysis” [19].

### 5.4.1. Dynamic approaches

#### 5.4.1.1. Frequency-domain analysis

The frequency-domain analysis is a linear analysis that aims to solve the dynamic response of a linearized system at either wave frequency, system subjected to first-order dynamic loads, or low frequency, system subjected to both second-order wave drift dynamic loading. “Because of the stochastic nature of frequency domain analysis, the primary results outputs are statistics and spectral density, which makes it not appropriate for analysing time-varying operations” [19].

“The response process is calculated in OrcaFlex by deriving, in stages, a set of linear transfer functions that map the wave elevation process through to the response process” [19].

$$\mathbf{x}(f_n) = \mathbf{X}(f_n)\boldsymbol{\lambda}(f_n)\eta(f_n) \quad (5.10)$$

Where,

$\mathbf{X}(f_n)$  = complex matrix-valued linear transfer function

$\boldsymbol{\lambda}(f_n)$  = complex vector-valued linear transfer function

$\eta(f_n)$  = discretised scalar-valued wave elevation process, calculated from the wave elevation spectrum

With,

$$\mathbf{X}(f_n) = (-(2\pi f_n)^2\mathbf{M} + i2\pi f_n\mathbf{C} + \mathbf{K})^{-1} \quad (5.11)$$

Where,

$\mathbf{M}$  is the system inertia matrix

$\mathbf{C}$  is the system damping matrix

$\mathbf{K}$  is the system stiffness matrix.

#### 5.4.1.2. Time-domain analysis

The time-domain analysis is a fully non-linear analysis that aims to solve the system’s dynamic response at each defined time step. An implicit or explicit integration scheme can solve the dynamic response.

“Both integration schemes use numerical time-stepping algorithms to solve the equations of motion in the time domain. For time-domain analysis, the primary results output is time histories of response variables” [19]. The time-domain analysis is split into two stages: the build-up and the main stages, figure 5.5. The build up stage is used to ramp the motions of the objects smoothly from zero to the defined value, giving a gentle start to the simulation.

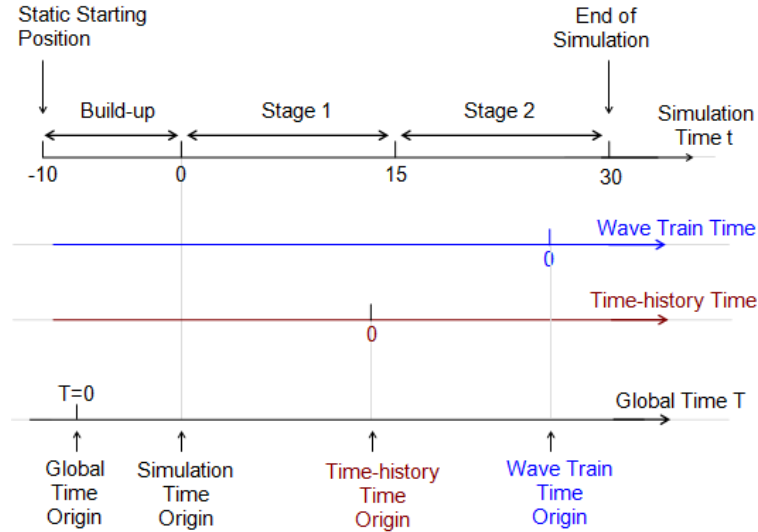


Figure 5.5: Overview time-domain analysis [20]

“The equation of motion which OrcaFlex solves for the time domain is” [19].

$$M(p, a) + C(p, v) + K(p) = F(p, v, t) \quad (5.12)$$

Where,

$M(p, a)$  is the system inertia load

$C(p, v)$  is the system damping load

$K(p)$  is the system stiffness load

$F(p, v, t)$  is the external load

$p$ ,  $v$  and  $a$  are the position, velocity and acceleration vectors, respectively

$t$  is the simulation time.

#### 5.4.1.3. Time-domain Explicit solver

“The explicit scheme is semi-implicit Euler with a constant time step. At the start of the time simulation, the initial positions and orientations of all objects in the model, including all nodes in all lines, are known from the static analysis.” [19]. The condition for a stable system is that the time step should be smaller than the shortest natural nodal period. The equation of motion (Newton’s law) for each free body (equation 5.13) is solved for the acceleration vector at the beginning of each time-step, resulting in equation 5.14 and then integrated.

$$M(p)a = F(p, v, t) - C(p, v) - K(p) \quad (5.13)$$

$$\begin{aligned} v_{t+dt} &= v_t + dt * a_t \\ p_{t+dt} &= p_t + dt * v_{t+dt} \end{aligned} \quad (5.14)$$

“At the end of each time step, the positions and orientations of all nodes and free bodies are again known and the process is repeated” [19].

#### 5.4.1.4. Time-domain Implicit solver

“For implicit integration, OrcaFlex uses the generalised- $\alpha$  integration scheme as described by Chung and Hulbert. The forces, moments, damping, mass etc., are calculated in the same way as for the explicit scheme. Then the system equation of motion is solved at the end of the time step” [19]. This scheme is unconditional stable for a linear system (backward Euler method). Still, the position, velocity and acceleration are unknown, so an iterative solution method requires more calculation time. On the other hand, it is possible to take much longer time steps than the explicit solver.

#### 5.4.2. Selected approach

The time-domain solver is chosen because the implementation of the motion of the crawler is time domain-based. There are 2 options for the time-domain solver and the implicit solver is chosen. This choice is made due to the number of segments that evaluate the jumper hose, which results in a lot of EOM for each free body. Due to the many segments, an explicit method would require small time steps.

## 5.5. Research question 4

This chapter clarifies the fourth research question: *What is the theory behind the dynamic software used and does it connect with static analysis?*. The theory behind the OrcaFlex model is divided into three sections. Subchapter 5.2 describes the theory behind the objects and environmental data  $d$ , which are used to construct the model. Subchapter 5.3 elaborates the statics and subchapter 5.4 elaborates the dynamics and explains the choice for an implicit time-domain solver.

The link between the static numerical Matlab model and static OrcaFlex model is elaborated in subchapter 5.1.



# Results and Analysis



# Static analysis - MatLab model

This chapter focuses on the static analysis and the corresponding MatLab model to analyse the varying parameters. Subchapter 6.1 describes the construction of the static jumper hose model. Starting with an axial-extensible catenary model, which is then enhanced with one buoyant node, then implementing a schematic overview of the buoyancy. Also, the segment size of the hoses is evaluated. Subchapter 6.2 takes a closer look into the implementation of the buoyancy and the trade-off between a realistic implementation and the suitability or variability of the parameter. Subchapter 6.3 displays the final static model that is used to vary the parameters for the parameter study, which is a follow up of the model of chapter 4. Also, the full answer is given to the sub research question 2, *"Which simplifications and adaptations are necessary for the contemplation of the static model?"*

## 6.0.0.1. Calculation structure for static analysis

This research aims to predict the dynamic behaviour of the jumper hose. Before this dynamical analysis is implemented, a static analysis is conducted. The construction of the first static model (cable only subjected to its own weight) consists of the following steps:

- The schematisation of the hose design and the limitations and boundaries of the model
- A basic catenary system, a string connected to two rigid points
- Analytical hand calculations to gain an understanding of the parameters and the system
- An expansion of the axial extensible catenary system, a rod connected to rigid points with simulated buoyancy
- The final axial extensible catenary system that is used as the first approximation of the jumper hose
- The extension and continuation of the axial extensible catenary system with geometrically non-linear elements and
- The resulting static model

After the first static model is completed, the next step is the implementation of the buoyancy. These steps are described in subsection 6.1.2.

## 6.1. The shape simulation of the jumper hose model

With the method explained and clarified in subchapter 4.3 a model to approximate the shape of the jumper hose is created. The jumper hose is evaluated with a FEM with non-linear elements built up in a sequence. The first step is to create the catenary cable and evaluate it with global calculations. Next, buoyancy is implemented. This buoyancy implementation is done in two steps. First, one node of the catenary cable is buoyant, the second step is to spread the buoyancy out over the jumper hose as designed in the preliminary design.

For the shape approximation model, the jumper hose is constructed with the following parameters defined in subchapter 3.2 and displayed below in table 6.1 and 6.2. The shape approximation model is built with Matlab. It is based on an existing script from the course 'Introduction to Computational Dynamics for offshore structures' (OE044090) of the Tu Delft, written by Chris Keijdenner and Joao Barbosa. The script approximates the shape of a mooring line with geometrically non-linear elements with

the FEMish approach as described above.

Parameters	Value	Unit
Length approximated		[m]
Length hose section		[m]
Total length		[m]
Number of sections		[sections]
P1(x1,y1)	;	[m;m]
P2(x2,y2)	;	[m; m]
Density slurry		[kg/m <sup>3</sup> ]
Density water		[kg/m <sup>3</sup> ]
Dry weight of hose material		[kg/m]

Table 6.1: Preliminary design parameters

Design parameters:

Parameters	Value	Unit
Length hose		[m]
Number of buoyancy modules		[modules]
Modules per section		[modules per section]
Buoyant hoses		[hoses]
Buoyancy starts at		[section]
Height VTS system from the seafloor		[m]
Width between crawler and VTS		[m]
Hose substance		[-]

Table 6.2: Preliminary design parameters

#### 6.1.0.1. Assumptions for the shape simulation model

The script generates a model that consists of many elements connected by nodes and the end nodes fixed on a coordinate. The elements are modelled by a rod that can only rotate and extend. These rods are connected and the angle between these elements is not restricted. The bending stiffness is not modelled in this design. It is assumed that the bending stiffness is not a critical parameter for this particular model for the shape simulation model. In the following chapter on validating the static shape model, the influence of the bending stiffness is evaluated.

The simulated elements are evaluated with geometrical non-linear elements throughout building of the shape simulation model. These elements can not be evaluated with small deformations and the associated assumptions because the elements show large deformations.

The shape simulation model is a static model where environmental forces are neglected for this particular model. In the following chapter on the validation of the static shape model, the influence of the environmental forces is evaluated.

#### 6.1.1. Catenary model, axial extensible

The first step in constructing the shape simulation model is to calculate and plot the shape of the hose due to its weight, supported by two connection points. This shape is a catenary shape and the first step of the shape simulation is the axial-extensible catenary model. The calculation of the axial extensible catenary model follows the same approach as described in subsection 4.3.2. The jumper hose is constructed of 14 hose sections of 22.86 m. These sections are divided into smaller segments representing the non-linear elements of subsection 4.3. The more segments used to evaluate a section, the more accurate the results are, but the number of iterations to reach the equilibrium positions increase



the calculation time. For the first estimation with an axial-extensible catenary, the segment length is equated to the section length. In subsection 6.1.4, the optimal segment length is engineered and the calculations are computed again.

The sections are evaluated by one segment and connected by nodes, resulting in a model of 14 segments and 15 nodes. The nodes can only move in x and z-direction, resulting in 2 degrees of freedom per node, a total of 30 DOFs. Figure 6.1 shows an example overview of the model evaluated with 14 sections and each section approximated with 3 segments. This section deviation results in a less angular design and the section can create a radius. The circles represent the nodes of the model and these nodes are hinges. The red nodes represent the end or begin node of a section and the section numbers are coloured red. The black numbers represent the node numbers of the model evaluated by three segments per section. The connection points of the hose are the end nodes and represent the connection with the crawler and the VTS. For this model, the global coordinate system is placed on the connection of the crawler.

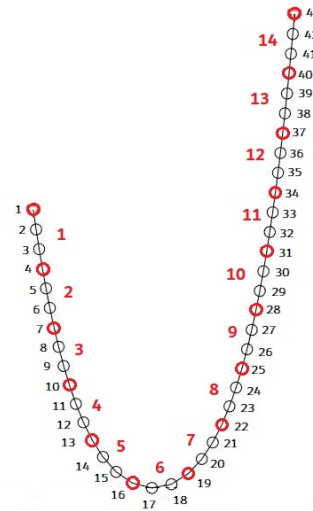


Figure 6.1: Sections and segments

6.1.1.1. Formulas to approach catenary shapes

A simplified parabolic function approximates the initial guess for the deformed shape because a catenary shape is close to the shape of a parabola (equation 6.1). The parabola went through the connection points (A and C) and guessed the third point between the connection points(B). For each connection point a algebraic equation is written (equation 6.2) and this results in a system of equations (6.3). By taking the inverse of the components matrix, the constants can be calculated (equation 6.4). The results are displayed in table 6.3.

$$y = a * x^2 + b * x + c \tag{6.1}$$

$$\begin{aligned} y_{A(2)} &= a * x_{A(1)}^2 + b * x_{A(1)} + c \\ y_{B(2)} &= a * x_{B(1)}^2 + b * x_{B(1)} + c \\ y_{C(2)} &= a * x_{C(1)}^2 + b * x_{C(1)} + c \end{aligned} \tag{6.2}$$

$$\begin{bmatrix} a_1 & b_1 & c_1 \\ a_2 & b_2 & c_2 \\ a_3 & b_3 & c_3 \end{bmatrix} * \begin{bmatrix} a \\ b \\ c \end{bmatrix} = \begin{bmatrix} y_{A(2)} \\ y_{B(2)} \\ y_{C(2)} \end{bmatrix} \tag{6.3}$$

**D\*[abc] = Y**

$$D^{-1} * Y = [abc] \tag{6.4}$$

Connection points	A = [0;0] B = [30; -50] C = [100; 75]
Constants	a = 0,062 b = -5,45 c = 0

Table 6.3: Results constants initial guess parabola

The external forcing arises from the weight of the hose and is uniformly divided over the elements. Then the force balances at the nodes are calculated, implemented in a system of equations and the residual is calculated. The tolerance of the system is set on  $1e^{-4}$ . Three iteration steps are displayed in figure 6.2 to 6.4. The whole iteration set is displayed in Appendix A.1. In figure 6.5 the initial parabola is plotted vs the equilibrium of the axial-extensible catenary.

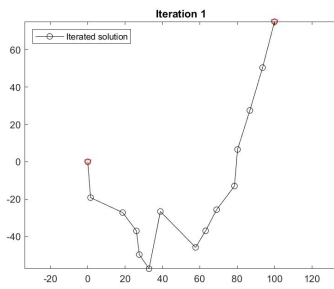


Figure 6.2: 1st iteration catenary system

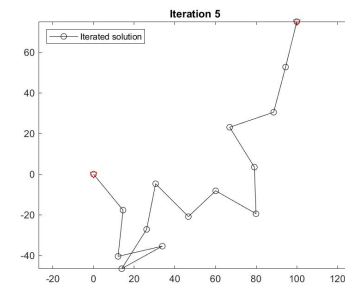


Figure 6.3: 5th iteration catenary system

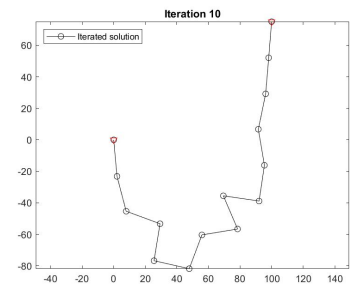


Figure 6.4: 10th iteration catenary system

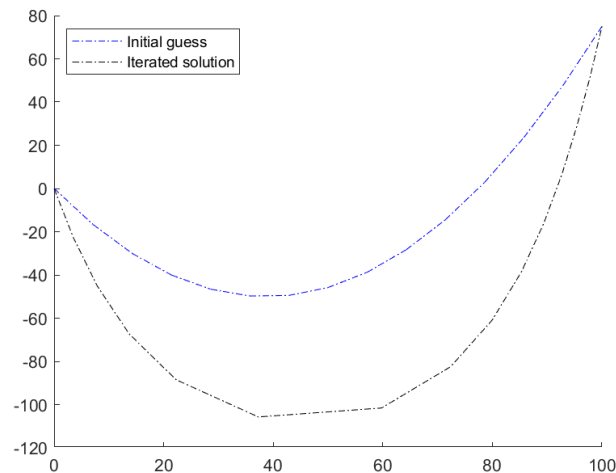


Figure 6.5: Initial parabola guess vs iterated catenary shape

The external force is modelled by taking half of the weight on the left segment and half of the weight of the right segment. Table 6.4 shows the external forcing on the nodes. The sum of the external load vector is 130,20 kN. This value corresponds to the weight calculations of subsection 4.2.3, table 4.4. In chapter 5, these values are compared to software calculations.

Node number	$P_{ext}$	Unit
1		[N]
2		[N]
3		[N]
⋮	⋮	⋮
13		[N]
14		[N]
15		[N]

Table 6.4:  $P_{ext}$  due to weigh jumper hose filled with slurry

### 6.1.2. Adding buoyancy to the static model concentrated on 1 node

The axial-extensible catenary model is the base for the jumper hose model. The next step is analysing the forces of the buoyancy modules and the shape of the jumper hose by making one node buoyant. The total buoyancy force is 139,99 kN due to 32 modules. This force is more than the total weight of the jumper hose filled with slurry, so it does not hit the ground. The buoyancy is placed on one specific

node. The buoyancy modules start at the fourth section, with four buoyant sections in the preliminary design. The nodes corresponding to the buoyant section are node 4 up to node 8. For the static model with the buoyancy force concentrated on one node, node 6 is selected as the buoyant node. This location is in the middle of the buoyant sections.

The new assembled external load vector is created, see table 6.5, which is a combination of the weight of the hose per node and the external force. The same procedure is executed. For each element, an internal force, stiffness matrix of the element and the tension are calculated, which result in a new node position. The residual forces are calculated and evaluated with the tolerance of the system. Three iteration steps are displayed in figure 6.2 to 6.4. The whole iteration set is displayed in Appendix A.2. In figure 6.9 the equilibrium of the one node buoyant model is shown.

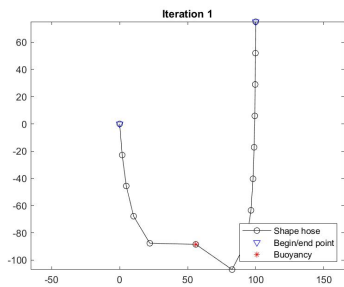


Figure 6.6: 1st iteration 1 node buoyant

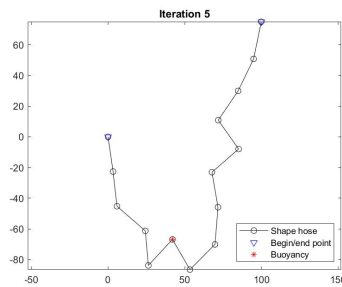


Figure 6.7: 5th iteration 1 node buoyant

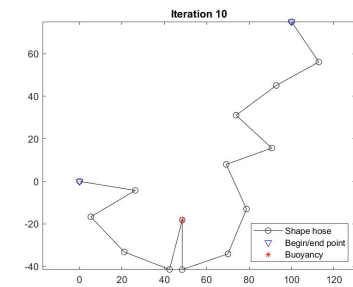


Figure 6.8: 10th iteration 1 node buoyant

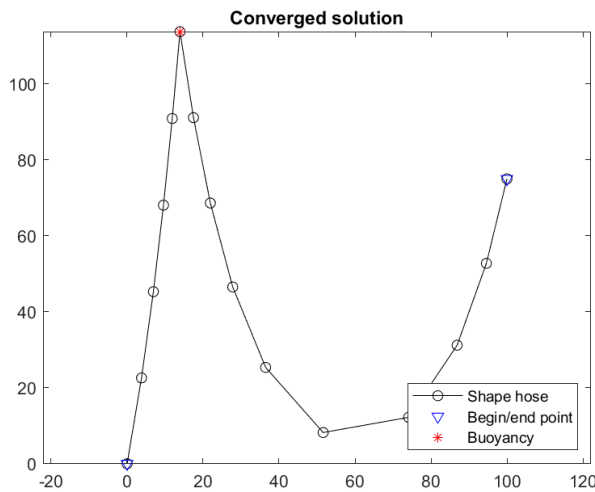


Figure 6.9: Equilibrium 1 node buoyant

Node number	$P_{ext}$	Unit
1		[N]
2		[N]
3		[N]
⋮	⋮	⋮
6		[N]
⋮	⋮	⋮
13		[N]
14		[N]
15		[N]

Table 6.5:  $P_{ext}$  due to weigh jumper hose filled with slurry and buoyant node 6

Parameters	Value	Unit
Number of nodes		[-]
Number of DOFs		[-]
Number of sections		[-]
Number of segments		[-]
Total weight of the hoses with slurry		[kN]
Total buoyancy force		[kN]
Location of buoyancy		[node]

Table 6.6: Number of Nodes, DOFs and other data

### 6.1.3. Spreading out the buoyancy in the static model

With the one node buoyant, the model shows it can go from an initial shape to an abstract desired s-shape due to the addition of buoyancy. To approach the preliminary design better, the buoyancy is modelled as described in subsection 6.2.1. After the buoyancy calculation, it is stated that there are 32 buoyancy modules needed and they are clamped with 8 modules per hose section, starting after the third section.

The nodes that represent the buoyant elements are nodes 4 to 8. The buoyancy modules are distributed evenly along the section. The dimensions are displayed in figure 6.10. For this model, the exact location of the buoyancy modules is not important because it is the first approximation of the s-shape. In subsection 6.2.1, considerations of the different possibilities to implement the buoyancy force are traded off. The total buoyancy force is distributed evenly along with the buoyant segments and the corresponding nodes, resulting in the following external load vector.

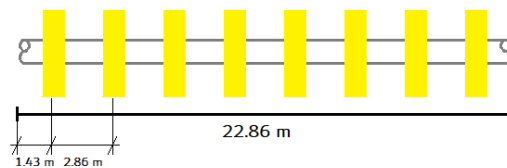


Figure 6.10: Buoyancy distribution over a section

The same procedure, as explained in subchapter 4.3, is applied to calculate the new node positions and the corresponding residual forces. The iteration steps are displayed in figure 6.11 to 6.13. Appendix A.3 shows the whole iteration set.

The new assembled external load vector is displayed in table 6.7 and figure 6.14 shows the equilibrium of the spread buoyancy model.

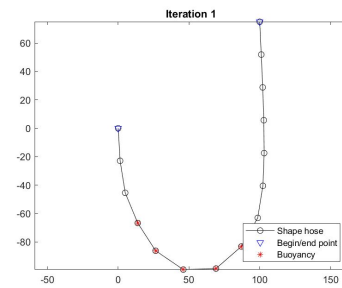


Figure 6.11: 1st iteration

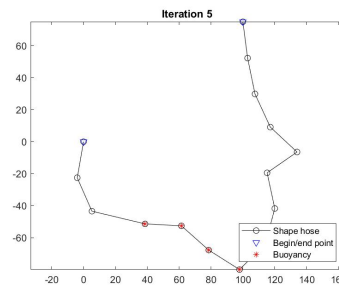


Figure 6.12: 5th iteration

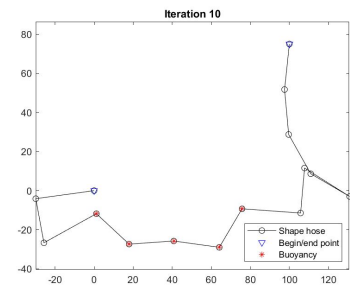


Figure 6.13: 10th iteration

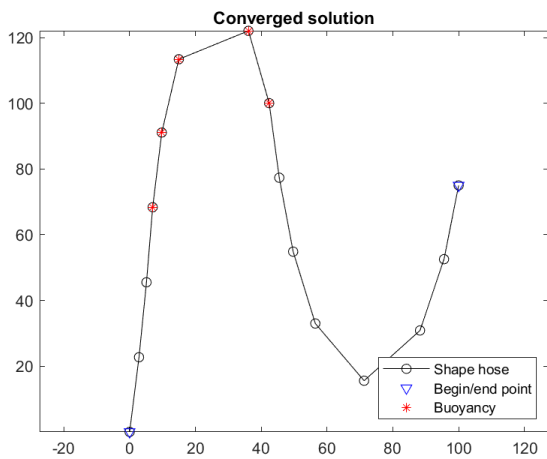


Figure 6.14: Equilibrium buoyancy spread out over 4 segments

Node number	$P_{ext}$	Unit
1		[N]
2		[N]
3		[N]
4		[N]
⋮	⋮	⋮
8		[N]
10		[N]
⋮	⋮	⋮
13		[N]
14		[N]
15		[N]

Table 6.7:  $P_{ext}$  due to weigh jumper hose filled with slurry and buoyant nodes 4 to 8

### 6.1.4. Segment size of the static model

In subchapter 4.3, the size of the structure element is an important aspect to capture certain local abnormalities. The smaller the size of the elements, the better the representation of the shape is. A downside of increasing the number of elements is the high amount of processed data and computational time.

The convergence study increases the number of segments per hose section and keeps the other parameters constant. For each number of segments per section, the highest point of the s-shape is calculated and plotted in graph 6.15. The graph shows the course of the heights y coordinate. In appendix A.4, the equilibrium situation of the weight vs the buoyancy of the number of segments per section is plotted, and a table with corresponding information. The convergence limit is reached when a maximum variation of 1% is relative to the length of the sections. The convergence limit is reached when the difference in the y coordinate of two segments is lower than 0.2286 m. In figure 6.16 the difference between two consecutive numbers is plotted. The limit is reached when the segments are updated from five segments per hose to six segments per hose. Five segments per hose section are selected.

In the convergence graph, a trap like behaviour is witnessed. This is because the highest node of the s-shape does not occur on the same arc length of the hose per increment of segment per section. The arclength describes the path over the jumper hose. Table 6.8 shows the heights node number and the corresponding arc length.

Number of segments per section	Highest node	Unit	Corresponding arclength	Unit
1	122,03	[m]	137,16	[m]
2	122,60	[m]	125,73	[m]
3	121,67	[m]	129,54	[m]
4	121,81	[m]	125,73	[m]
5	121,54	[m]	128,02	[m]
6	121,56	[m]	125,73	[m]
7	121,46	[m]	127,36	[m]
8	121,45	[m]	125,73	[m]
9	121,41	[m]	127,00	[m]
10	121,38	[m]	125,73	[m]

Table 6.8: Highest node vs corresponding arclength

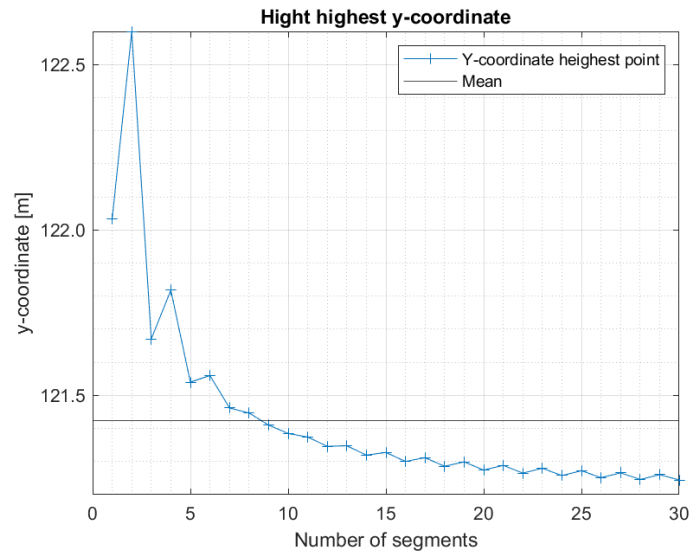


Figure 6.15: Highest point of the s-shape vs number of segments per section

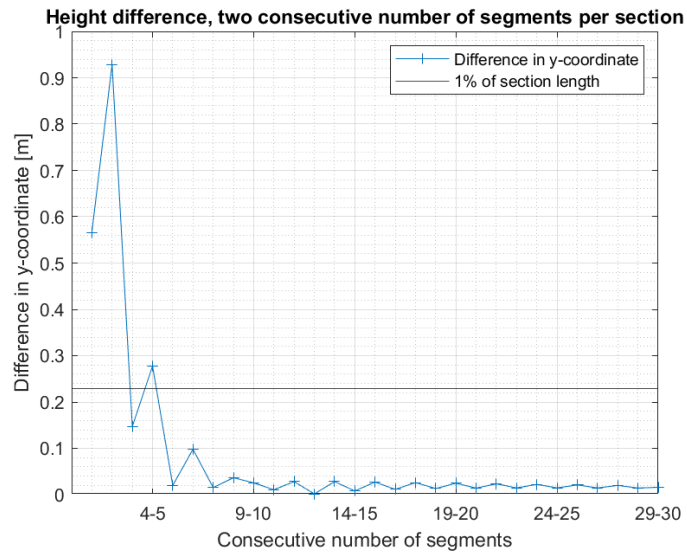


Figure 6.16: Convergence spreaded buoyancy

## 6.2. Model design for parameter variation

With the parameters of table 6.15, the static model is calculated, following the literature of subchapter 4.2, 4.3 and 6.1. The code is written in Appendix C and constructed in the following sequence.

1. The definition of all the parameters applied to the model.
2. Connecting the elements and their properties to get a straight line.
3. An initial guess of the deformation is made.
4. With that basic deformation, the resulting shape is calculated with the catenary theory of subchapter 4.2.
5. The load vector is assembled, in subsection 6.2.1 the implementation and construction of the load vector are appointed.
6. The new deformation is calculated starting with the outcome of the catenary solution.

### 6.2.1. Implementing buoyancy

A section is divided into 5 segments, which are 4.572 meters long each. Figure 6.17 shows an overview of the dimensions of the clamping system for the specific parameters of the initial design of the static model.

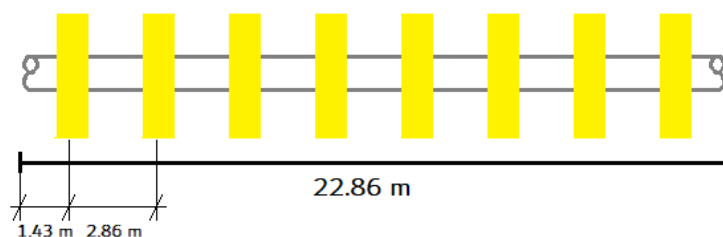


Figure 6.17: Buoyancy distribution

The implementation of the buoyancy is done with different levels of accuracy. There are several options to model the buoyancy, enumerated from modelling the real situation to making simplifications.

1. Trying to implement the design parameters to the model
  - (a) Implement the buoyancy force of a buoyancy module exactly on the module location as displayed in figure 6.17.
  - (b) Rearranging and scaling the buoyancy force to the number of segments. Implementing a calculated buoyancy force to the nodes as shown in figure 6.14.
  - (c) Rearranging the number of segments to have a length corresponding to spacing between the buoyancy modules.
2. Simplifying the buoyancy by taking an average force per segment due to the number of modules per section
  - (a) Implement the average buoyancy force per segment due to the total number of buoyancy modules and modules per section. This force is translated to the corresponding nodes over the exact length of the buoyancy sections.
  - (b) Implement the force due to the number of modules per section. Full sections have an average force per segment, and the end section has a different force per segment depending on the number of modules over that section.
  - (c) Implementing the average buoyancy force over a rounded number of buoyant segments.
3. Simplifying the buoyancy by distributing the buoyancy force over the sections

- (a) Implementing the buoyancy force by dividing the total force by the number of buoyant segments. Distributing the calculated force per segment over the corresponding segment nodes.
- (b) Implementing the buoyancy force by dividing the total force by the number of buoyant sections. Distributing the calculated force over the corresponding sections and all the nodes representing the section. The number of sections is rounded upward.

Note that the initial design parameters of the static model are based on 32 modules with 8 modules clamped per section. The parameters are varied and combined for the parameter study, resulting in different outcomes on calculated values.

*Option 1 a to c* are very hard to implement in the model due to the buoyancy's exact location, which does not always match the nodes and element length. The Matlab model used for the static analysis has the limitation that the buoyancy force can only be implemented on the nodes of the model.

Another downside is that multiple variations of different parameters like the number of modules per section affect the spacing and the hose's implementation location. The design and environmental parameters are already simplified and the goal of this Matlab is a parameter variation study. Reducing the segment length results in an exponentially increasing calculation time and engages an overkill. The segment length is a fixed number in the parameter variation study, so it can't be adapted per variation of modules per section.

*Option 2 a to c* is simplifying the design, ranked from most realistic to abstract (a to c). These options are implementable in the Matlab model. *Option 2c* is the biggest simplification of *option 2* and is implemented to review the preliminary parameters in subchapter 4.2. The results show that it doesn't represent the desired shape.

The choice has been made to implement *Option 2a*, implementing an average buoyancy force per node over the length of buoyancy modules. This option is selected because *Option 1* is unrealistic to implement. The only option for implementing the force is on the nodes, depending on the chosen segment length. Also, varying the parameters result in switching between buoyant nodes all the time, which leads to a different difference between the designed location and the implemented location. The second reason is that there is no bending stiffness in the model, so if the buoyant nodes are discontinuous, the node that does not have a buoyancy force on them moves into an axial-extensible catenary shape until a node with buoyancy implemented.

*Option 3 a to c* are a bigger simplification than *option 2*. So these are not chosen because they deviate further from reality.

### 6.2.2. Buoyancy implementation of the initial design

The total number of buoyant nodes depends directly on the total number of modules, the number of modules clamped per section, the distribution of the forces and the segment length. Table 6.9 and 6.10 show that the number of buoyant sections varies when the total number of modules is varied or when the number of clamped modules per section varies.

Fixed, number of modules	Modules per section	Number buoyant sections	Number of modules	Fixed, modules per section	Number buoyant section
32	8	4	25	8	3,125
32	7	4,5714	26	8	3,25
32	6	5,3333	27	8	3,375
32	5	6,4	28	8	3,5
32	4	8	29	8	3,625
32	3	10,6667	30	8	3,75
32	2	16	31	8	3,875
32	1	32	32	8	4

Table 6.9: Variation parameters, influence buoyant sections

Table 6.10: Variation parameters, influence buoyant sections



The buoyancy modules start at the beginning of a section and spacing between track centres is kept constant. This design choice of the initial design results in the following parameters and measurements when the number of clamped modules is varied, see table 6.11 and figure 6.18. 6.18.

Modules per section	Spacing between centres of modules	1/2 spacing between track centres of modules	Distance begin section to centre module
8	2,8575	1,4284	1,4284
7	3,2657	1,6329	1,6329
6	3,8100	1,9050	1,9050
5	4,5720	2,2860	2,2860
4	5,7150	2,8575	2,8575
3	7,6200	3,8100	3,8100

Table 6.11: Dimensions spacing between buoyancy modules

The slurry's weight distribution and the hose material are evenly distributed over the hose. The hose model is built up of sections. Sections are divided into segments that are connected by nodes. The force vector is applied to the nodes and constructed by force per meter multiplied by the segment length. The modelling of the force vector on a node exists of half the weight of the left segment and half of the weight of the right segment. The buoyancy is also equally distributed over the buoyant hose length and the number of buoyant nodes depends on the relative occupation of the last hose section. For the design model of eight modules per section, one buoyancy module occupies 1/8 of the section length, which includes 1/2 of the left and 1/2 of the right area of the buoyancy module.

The last factor of influence on the number of buoyant nodes is the segment length of the hose. The sections are divided into five segments, each 4,572 m wide for the initial static design. The number of segments is evaluated at subsection 6.1.4. The node numbers segment length does not correspond to the buoyancy modules centre locations and force load carrying range.

Node	Location [m]	Node	Location [m]
Node 1	4,57	Node 9	41,14
Node 2	9,14	Node 10	45,72
Node 3	13,71	Node 11	50,29
Node 4	18,28	Node 12	54,86
Node 5	22,86	Node 13	59,43
Node 6	27,43	Node 14	64,00
Node 7	32,00	Node 15	68,58
Node 8	36,58		

Table 6.12: Node locations of the segment

Centre	Location [m]
Buoyancy module 1	1,4284
Middle module 1 - 2	2,8575
Buoyancy module 2	4,2862
Middle module 2 - 3	5,7150
Buoyancy module 3	7,1437
Middle module 3 - 4	8,5725
Buoyancy module 4	10,0012
Middle module 4 - 5	11,4300
Buoyancy module 5	12,8587
Middle module 5 - 6	14,2875
Buoyancy module 6	15,7162
Middle module 6 - 7	17,1450
Buoyancy module 7	18,5737
Middle module 7 - 8	20,0025
Buoyancy module 8	21,4312

Table 6.13: Locations centre buoyancy modules

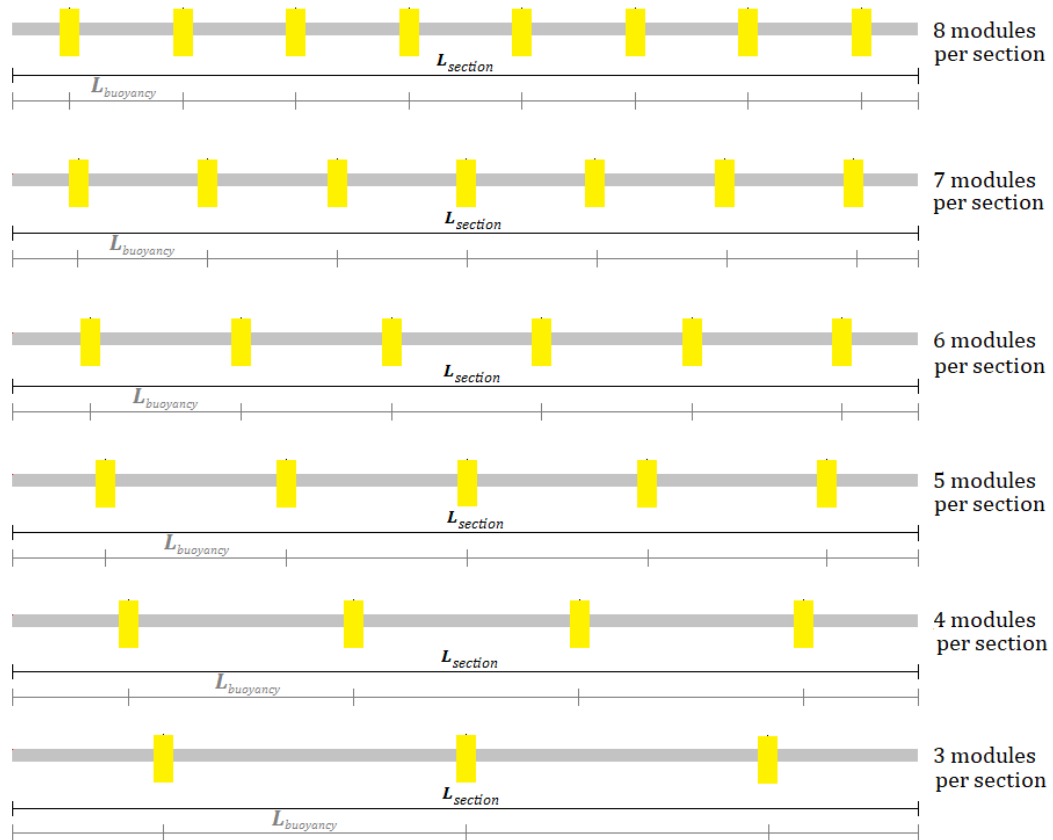


Figure 6.18: Segment length with varying buoyancy per section

As an approximation, the last buoyant segment is rounded to a whole segment and then the corresponding node is calculated. This approximation is done to model the total implemented buoyancy length close to the exact (start module to module) buoyancy module length. An example calculation is shown in table 6.14 where the total number of buoyancy modules is varied and this shows a varying number of buoyant sections. Figure 6.19 is a visual overview of table 6.14.

Total number of modules	Modules per section	Number of buoyant sections	Number of segments	Rounding off number of segments	Number of buoyant nodes
24	8	3	15	15	16
25	8	3,125	15,625	16	17
26	8	3,25	16,25	16	17
27	8	3,375	16,875	17	18
28	8	3,5	17,5	17	18
29	8	3,625	18,125	18	19
30	8	3,75	18,75	19	20
31	8	3,875	19,375	19	20
32	8	4	20	20	21

Table 6.14: Results calculation of buoyant nodes

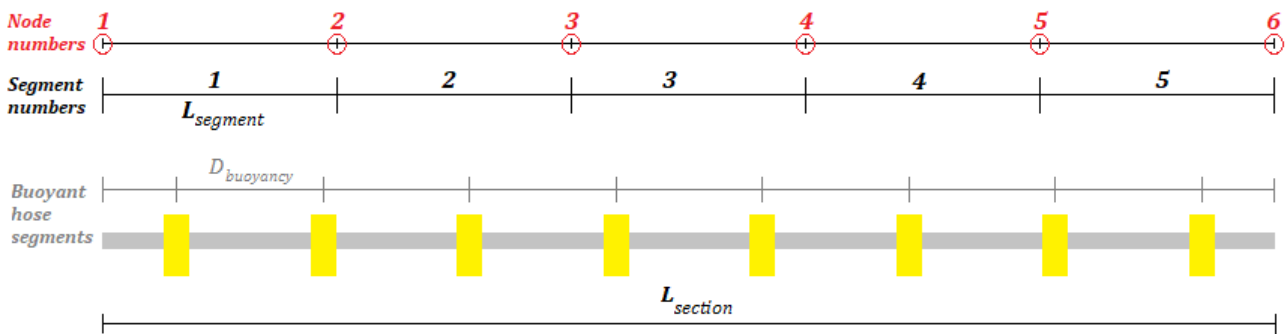


Figure 6.19: Schematization 8 modules per section and 5 segments per section

The corresponding number of segments is calculated by calculating the relative number of buoyant sections. This number is rounded to the nearest integer.

$$\text{Number of buoyant sections} = \text{Total number of modules} / \text{Modules per section} \tag{6.5}$$

$$\text{Number of buoyant segments} = \text{Number segments per section} * \text{Number buoyant sections} \tag{6.6}$$

$$\text{Relative number of buoyant segments} = \text{round}(\text{Number of buoyant segments}) \tag{6.7}$$

With the updated number of segments, the number of nodes is calculated. Each segment has two connecting nodes. One segment results in two nodes, two segments result in three nodes, and so on. The number of buoyant nodes is the updated number of segments plus one.

$$\text{Number of buoyant nodes} = \text{Relative number of buoyant segments} * 2 + 1 \tag{6.8}$$

There is one special case for the number of segments. Suppose the number of segments is exactly a half, then the number of segments is rounded downward. This results from the force distribution of the nodes, 1/2 segment left and 1/2 segment right. Figure 6.20 shows a section of the last buoyant hose section with parameters, 25 modules and eight modules clamped per section. Here 3.5 sections of hoses are buoyant and 17.5 segments (equation 6.9 to 6.13 ).

$$28 \text{ modules} , 8 \text{ per hose}, \tag{6.9}$$

$$\text{Number of buoyant sections} = 3,5 \tag{6.10}$$

$$\text{Number of buoyant segments} = 17,5 \tag{6.11}$$

$$\text{Relative number of buoyant segments} = 17 \tag{6.12}$$

$$\text{Number of buoyant nodes} = 18 \tag{6.13}$$

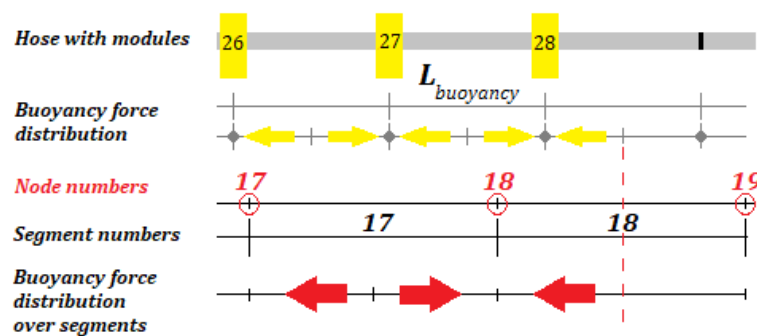


Figure 6.20: Zoom of schematization 8 modules per section and 5 segments per section

## 6.3. Continuation on sub research question 2 and the final static MatLab model

The subquestion, "Which simplifications and adaptations are necessary for the contemplation of the static model?" is answered in this chapter. As stated in subchapter 4.4, the following parameters are disregarded:

- Bending stiffness
- Bending deformation
- Shear deformation
- A continuous hose
- Resistance against rotation at the connection points

The static model is simplified by:

- Dividing the structure into smaller and simpler elements with boundary conditions
- The elements have an axial stiffness, unit mass and tensionless length
- Deformation of the elements only happens axial (rotation and extension)
- The elements behave as rod elements
- The rod elements are connected and can freely rotate around the connection point

The simplification of the buoyancy is elaborated in subchapter 6.2.1. It has been chosen to spread the buoyancy force over the buoyant nodes. The number of buoyant nodes depends on the relative location of the last buoyancy model.

The following parameters are used to construct the Static MatLab model. This model is used in chapter 7 to vary the parameters and find a new set of parameters. Figure 6.21 shows the axial-extensible catenary shape of the hose, and figure 6.22 shows the same model with buoyant nodes. In figure 6.23 the tension of both models is plotted, and it's shown that three locations are important. The starting point, the node before the buoyant nodes and the end node. These locations are explained in subsection 9.2.4.

Preliminary parameters are implemented and updated, and the base model is constructed. The summarised parameters of the base model are displayed in table 6.15.

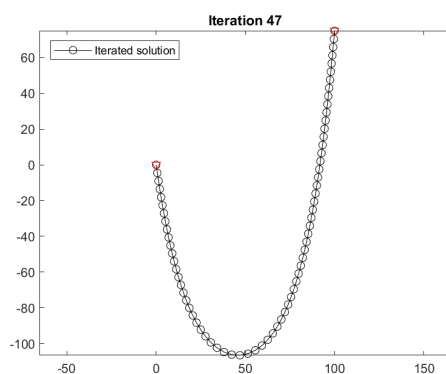


Figure 6.21: Catenary model of weight Jumper hose

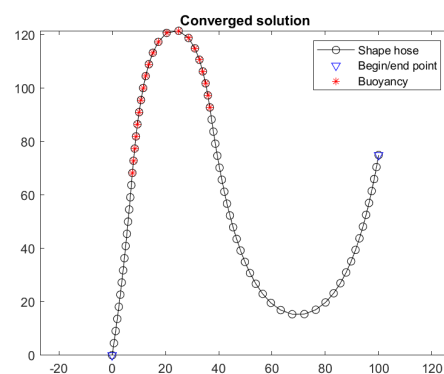


Figure 6.22: S-shape Jumper hose

Parameters	Value	Unit
EA hose		[N]
ID		[m]
OD		[m]
Area ID		[m <sup>2</sup> ]
Area OD		[m <sup>2</sup> ]
Length hose		[m]
Section length hose		[m]
Segment length hose		[m]
Height VTS		[m]
Width crawler		[m]
Number of sections		[-]
Number of segments		[-]
Number of nodes		[-]
Number of DOFs		[-]
Number of buoyancy modules		[modules]
Number of buoyancy modules per section		[modules per section]
Starting section buoyancy	4th	[section]
Number of buoyant nodes	21	[-]
Start node buoyancy	36	[-]
End node buoyancy	74	[-]
Total weight of the hose with slurry	130,20	[kN]
Weight per node	1,860	[kN]
Total buoyancy force	133,99	[kN]
Buoyancy per node	6,38	[kN]

Table 6.15: Results static model summarized

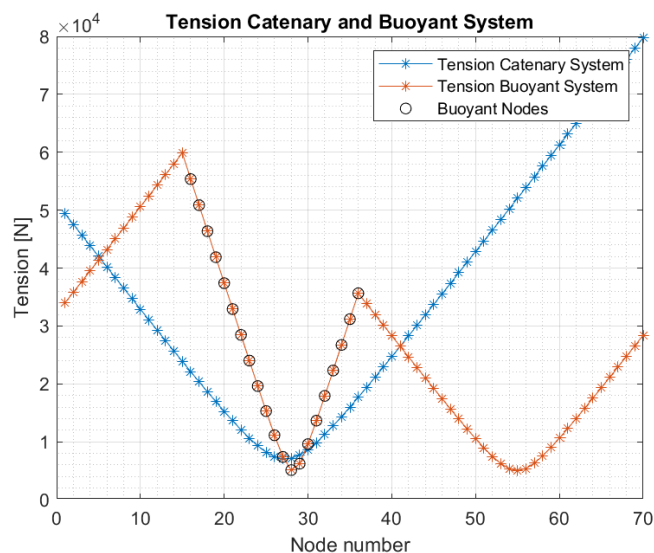


Figure 6.23: Tension per node catenary shape and s-shape



# Parameter study

The research is split into two parts, where the first part focuses on identifying a new set of parameters for the jumper hose design. In chapter 6 the static analysis of the preliminary design of the blue nodules document is performed. Chapter 7 presents the corresponding parameter study and the optimisation of the selected set of parameters from the multicriteria analysis. Subchapter 7.1 documents the selected parameters, restrictions of the design and the failure criteria. This subchapter also evaluates the influence of the parameters on the design and presents a ranking of the impact per parameter on the design.

Subchapter 7.2 elaborates the combination of parameters to a design set. The multicriteria analysis is clarified and established in subchapter 7.3. The multicriteria analysis results in an optimal set of parameters. Subchapter 7.4 performs a sensitivity study on this set, and subchapter 7.5 shows the optimal set of parameters.

This chapter builds up to and answers the third research question, *“How is the new/optimal set of parameters identified?”* and is summarised in subchapter 7.5.

*The design parameters have changed during the research and updates are described in subsection 8.2.4, table 8.15*

## 7.1. Parameters selection

In chapter 4 the static analysis of the initial design is performed. The implemented parameters are the hose's length, the number of modules, the number of modules per section, the height of the connection point of the VTS, the section where the buoyancy starts, the width between the crawler and the VTS and the mass of the slurry. The first goal of the thesis is to perform a new parameter study and find the optimal combination.

Some parameters were changed during the initial design study of the jumper hose. In the past years, the subsea harvesting system has been updated. So this stimulates the interest in examining the parameters of the jumper hose. In the scope of the project characteristic, the following parameters are mentioned, table 7.1. They are kept constant and are not included in the variation of the parameters.

In this part of the parameter study, only the static equilibrium is considered, the Matlab model of chapter 6. In part II, the dynamic analysis, the dynamic behaviour of the optimal set of parameters is examined, and if unexpected behaviour appears, the set of parameters is evaluated and updated.

Parameter	Varied or fixed
Total length hose	Varied
Length segment	Fixed
Element length	Fixed
Diameter hose	Fixed
Wall thickness	Fixed
Material/design hose	Fixed
Slurry density	Varied
Height VTS connection	Varied
Distance crawler to VTS	Varied
Height connection point Crawler	Fixed
Number of buoyancy modules	Varied
Type of buoyancy modules	Fixed
Distance between buoyancy modules	Varied
Location buoyancy modules	Limited
Starting point buoyancy modules	Varied

Table 7.1: All parameters of the jumper hose design

The following design elements are neglected for the parameter study because the hose's bending stiffness is not implemented in the Matlab.

- Bending restrictor at the connection of the crawler
- Bending restrictor at the connection of the VTS
- Bending stiffness of the segments and section
- Minimum bending radius of the section

These design elements are evaluated and implemented in the dynamic analysis part of the document.

### 7.1.1. Restrictions of the design

The current set of parameters consists of: the length of the hose, the number of modules, the number of modules per section, the height of the connection point of the VTS, the section where the buoyancy starts, the width between the crawler and the VTS and the mass of the slurry. Some of these parameters contribute to the operating limitations of the jumper hose. It is stated that:

- The jumper hose may not fail in the operational radius (distance between crawler and VTS)
- The jumper hose should operate when water is transported through the hose
- The jumper hose should operate when the slurry is transported through the hose

There are also failure scenarios described and the following demands are stated, so the hose does not fail:

- The jumper hose should never capsize the crawler. The maximum bending moment induced by the hose on the crawler is 120 kNm.
- The Jumper hose should never lift the crawler from the seafloor. The maximum pulling force induced by the hose on the crawler is 200 kN.
- The lazy s-shape of the jumper hose should never hit the seafloor
- The maximum tension in the hose should never pass 1600 kN
- The first meter of the hose, evaluated from crawler to hose, should stay between an angle of 45 degrees
- The first meter of the hose, evaluated from VTS to the hose, should stay between an angle of 45 degrees



### 7.1.2. Failure criteria

With the design, model segments and restrictions of the jumper hose, the fail criteria are determined and calculated.

Parameter	Value	Unit
Bending moment		[kNm]
Lifting force		[kN]
Update weight crawler for lift force		[kN]
Height lowest point lazy-s		[m above seafloor]
Angle between hose and crawler		[degree]
Angle between VTS and crawler		[degree]
Operation radius		[m]
Density water		[kg/m <sup>3</sup> ]
Maximum density slurry		[kg/m <sup>3</sup> ]
Maximum tension in the hose		[kN]

Table 7.2: Maximum workability values of the failure criteria

The values described in table 7.2 are the ultimate values of the workability of the jumper hose. These values are not desired, so the following workability values are updated and handled to consider safety and safety factors from the initial jumper hose design (table 7.3).

In the 'Detailed design of the Jumper hose', some recommendations regarding the further analysis of the jumper hose are stated. As an example, 'the generated uplift force of the assembly line a safety factor of 2 should be applied between  $T_{max}$  and the HV weight for further analysis' [15].

The fail criteria of the height of the lowest point of the lazy S is located at the ocean floor, but for workability, this is still very risky because the crawler could hit it. The new workability criteria are set on the crawler's height plus an additional 2.5 meters for safety.

Parameters	Value	Unit
Lift force crawler		[kN]
Height lowest point lazy-s		[m]
Maximum tension in the hose		[kN]

Table 7.3: Workability criteria

### 7.1.3. Variation and influence per parameter

The goal is to get an optimal set of parameters. This set is determined with a multicriteria analysis, but first, the maximum number of parameters set is constructed. This choice is an important decision because it influences the outcome of the multicriteria analysis. Evaluating the possible parameter combinations is a step before the multicriteria analysis. The combinations are limited by the parameters selected because, for example, a hose of 300 meters with buoyancy starting at the 4th section and 28 modules, three clamped per section, is not feasible. The combinations are not limitless, so the design's influence is elaborated below.

The current set of parameters consists of the length of the hose, the number of modules, the number of modules per section, the height of the connection point of the VTS, the section where the buoyancy starts, the width between the crawler and the VTS and the mass of the slurry. Some of these parameters contribute to the operating limitations of the jumper hose, the width between the crawler and the mass of slurry transported. The influence of the other parameters are looked at separately.

The parameter variation and influence are based on the initial design parameters of table 6.15

### Length hose

The hose length is an important variable because of the request of IHC to investigate if the hose can be longer than in the preliminary design. Figure 7.1 shows that the influence of the length has an impact on the shape of the hose. This height difference is because of the extra weight of scaling the hose from 250 to 450 meters. A longer hose also influences the distance between the crawler and the VTS. With a longer hose, the crawler could move further away from the VTS, limiting the minimum distance between the crawler and the VTS. IHC stated that the crawler should stay within a 200 meter radius with the VTS. In figure 7.1 the hose goes below the seabed, this would result in failure due to the set criteria. The consequences of the shape are taken into account in the establishment of the parameter sets in subchapter 7.5.

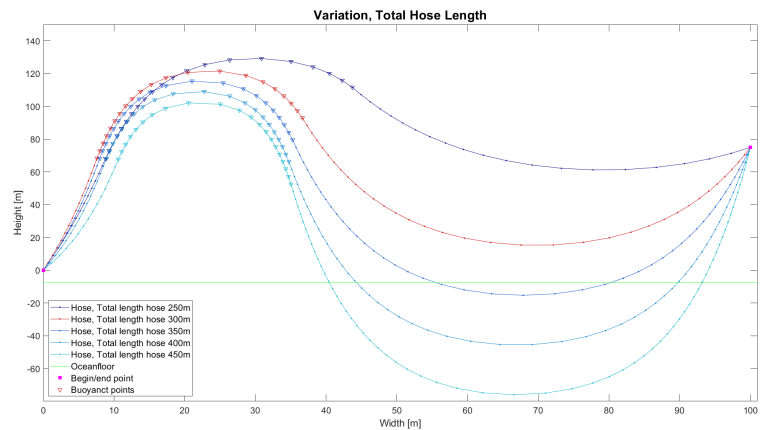


Figure 7.1: Length Variation

### Height connection point with VTS

Varying the height of the connection point of the VTS is displayed in figure 7.2. This height difference has a relatively small influence on the shape of the hose. The higher the connection point, the better for this hose, but it influences the working range of the crawler.

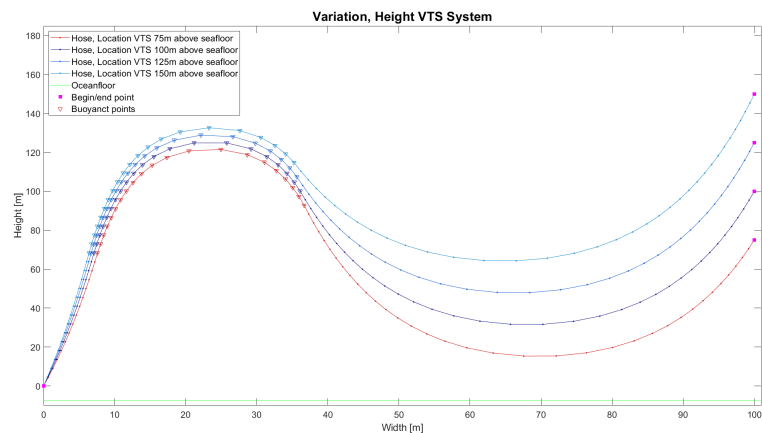


Figure 7.2: Height variation

### Total number of modules

Varying the total number of modules is displayed in figure 7.3. The first guess for the variation of the modules is based on a 300 meters hose with three modules per hose and a maximum number of loaded sections of eleven sections. The option of two modules per hose is not feasible because that would result in a maximum of 28 modules if all hoses were buoyant.

The fewer modules used, the lesser weight can be lifted. From figure 7.4, it can also be seen that the location and the number of modules per section play a huge influence on the shape of the hose in combination with the total number of modules.

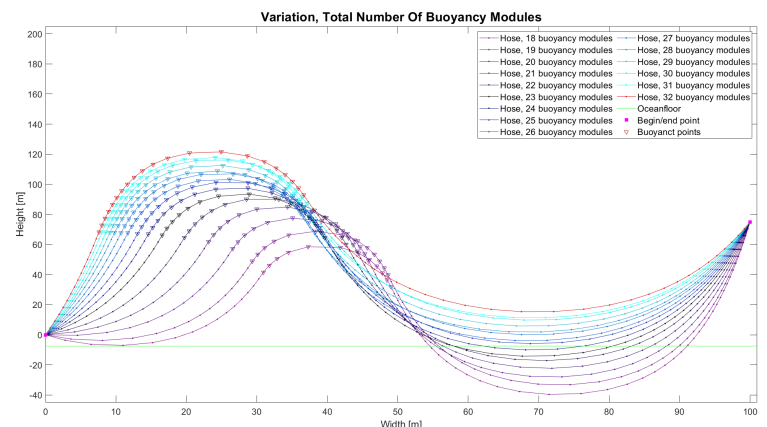


Figure 7.3: Number of modules variation

### Number of modules per section

Varying the number of modules per section is displayed in figure 7.4. The modules per hose vary from three per hose to eight per section. Eight per section is the maximum number of modules that can be clamped to the section, three per hose is the minimum value because 28 modules starting at the fourth section are designed for the preliminary design. 28 Modules with two per hose does not fit in the preliminary design. The modules per section have a big influence on the shape of the hose due to the spreading of the buoyancy force. It's also shown why the loading of the last section is not desirable for the shape of the hose.

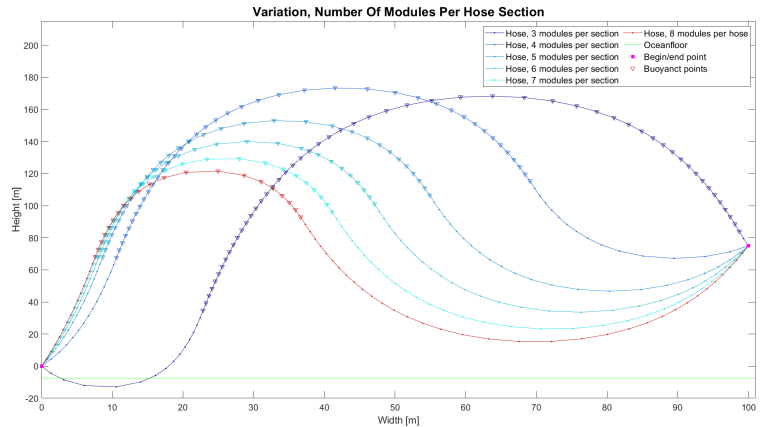


Figure 7.4: Modules per segment variation

### Starting segment of buoyancy

Varying the starting section of the buoyancy is displayed in figure 7.5. This variation influences the shape of the hose, just like the number of modules and number per section. The later the buoyancy starts, the more force the first part of the buoyancy needs to lift. It results in a reverse s-shape and a hanging first part of the hose. Also, if the buoyancy is not spread over the hose, it could result in a lower location of the lowest point of the VTS.

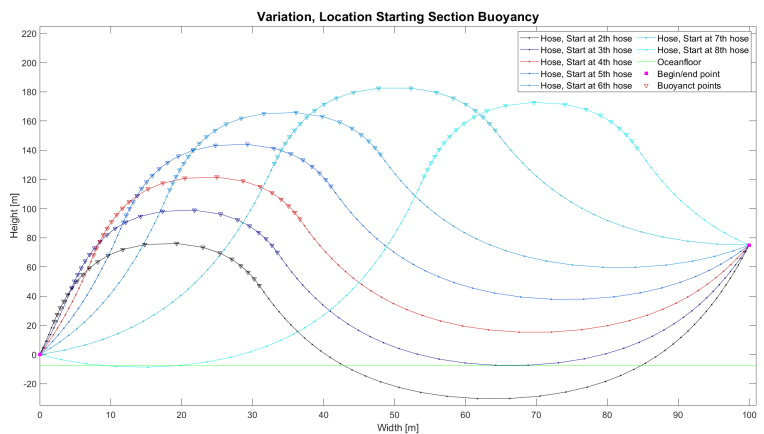


Figure 7.5: Starting segment variation

### Range of the crawler

The range of the crawler is an important requirement of the model. The range is selected from 50 meters up to 200 meters distance from the connection point with the VTS. The shape of the hose is influenced a lot. The further the crawler is away from the VTS, the bigger the momentum induced by the hose on the crawler, figure 7.6.

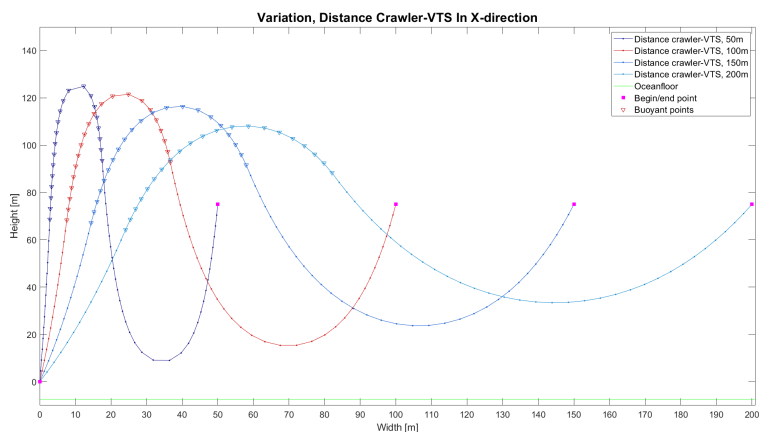


Figure 7.6: Range variation

## Hose substance

Varying the transported density of the fluid is displayed in figure 7.7. This influences the weight of the hose and so the height and shape of the hose. The submitted working range is from transporting water ( $1025 \text{ kg/mm}^3$ ) up to a slurry of  $1200 \text{ kg/m}^3$ . The design needs to operate in this range.

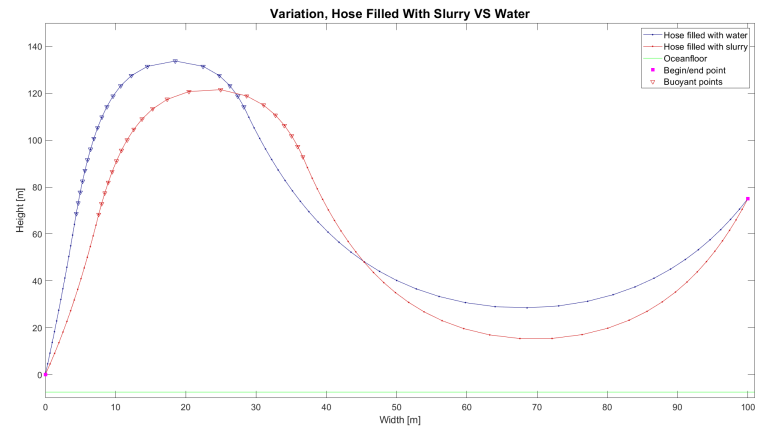


Figure 7.7: Hose substance variation

## Conclusion

The impact of variation of the parameters depends per parameter. The impact of the parameters also depends on the combinations between the parameters. This is investigated in subchapter 7.5

### 7.1.4. The ranking of the parameters

The workability parameters, the horizontal distance between the crawler and VTS and the transported density are operational parameters. The design of the jumper hose needs to operate within the given failure criteria; otherwise, the parameter sets are disregarded in the multicriteria design.

The other parameters are ranked from:

1. Number of modules per hose section
2. The starting point of the buoyant section
3. Length of the hose
4. Total number of modules
5. Height VTS

The number of modules per hose section is chosen as the most important design parameter. Figure 7.4 shows that this variation has the biggest difference in shape. It is shown that spreading out the buoyancy is beneficial for the height of the second part of the s- shape and shows some unrealistic options. Also, the starting point of the buoyancy is an important parameter due to the shapes shown in figure 7.5. This parameter also influences the moment induced by the hose on the crawler and the pulling force on the crawler or which part of the hose the buoyancy carries. Scaling the number of buoyancy increases the pulling force on the hose and the number of buoyant sections. But the influence per step size is quite small compared to the number of modules clamped per hose and the starting point of the buoyant section. The length of the hose has a direct influence on the range of the crawler and so on the moment induced by the hose. A large hose can reach further than a smaller hose but vice versa for the minimum distance. Also, the height of the VTS has the same influence, but the step size is chosen to be smaller due to the preliminary design of the VTS. Both are restricted by the diagonal length between the crawler and the connection with the VTS.

## 7.2. Combinations of parameters

The variations and combinations of parameters are not limitless. The range of the variation of the parameters is determined by the ranking of the parameters, the restrictions of IHC and the feasibility of combinations. For the parameters, the variation is reasoned by the preliminary design parameters.

The number of clamped modules could reach a maximum of eight and is set at a minimum of 3. This minimum is because two modules clamped per hose would result, for the minimum length of 300 m and 14 sections, in 12 section buoyant and 24 modules in total. This maximum does not reach the calculated number of the preliminary design. The starting point of the buoyancy is limited by the number of modules clamped, the total length and the total number of modules. This parameter is restricted by the maximum number of modules implemented and the minimum modules per section.

The hose length is set at 300 to 450 meters. The variation of the height of the VTS is set at 75 to 125 meters above the seafloor. These parameters are selected in consultation with IHC. The number of modules depends on the minimum number of clamped modules, starting section and the minimum length of the hose. The dependence is balanced between the starting point and the total number of modules. Figure 7.5 shows that the later the buoyancy starts, the higher the chance that the first part of the hose is not in tension. Another trade-off is the loss of modules if the starting section goes up three modules per section. For 28 modules, three per hose results in 9 1/3 section buoyant. The largest starting section is three and a maximum of 30 modules in total.

Subsection 4.2.4 states that 32 modules are needed to lift the full weight of the 300 meter long hose filled with slurry. When the length of the hose is varied, the weight increases. This extra weight results in an extra update for the total number of buoyancy modules parameter because by elongating the hose, the total weight of the hose increases. For the parameter study, this parameter is varied from 300 to 450 meters. Table 7.4 shows the minimum and the maximum number of buoyancy modules per hose length.

Hose length [m]	Total weight in kN	66% of the weight carried by modules	100% of the weight carried by modules
300			
350			
400			
450			

Table 7.4: Weigh of the hose per length and number of buoyancy modules

This analysis concludes in the following parameters variations and their range:

Parameter	Range	Unit
Number of clamped modules per segment	3 to 8	[pieces per segment]
Starting segment (after)	first to third	[segment]
Length of the hose	300 to 450	[meters]
Number of modules (300 m hose)		[modules]
Number of modules (350 m hose)		[modules]
Number of modules (400 m hose)		[modules]
Number of modules (450 m hose)		[modules]
Height VTS connection above seafloor		[meters]
Range crawler		[meters]
Mass of the slurry		[kg/m <sup>3</sup> ]

Table 7.5: Range width per parameter

### 7.2.1. Parameter sets

The parameter variation range is presented in table 7.5. The parameters are combined and varied in the following manner (Table 7.6):

Length [m]	Height [m]	Modules [pieces]	Clamped [pieces per segment]	After segment [after segment]	Hose substance, density [kg/m <sup>3</sup> ]	Range [m]
300	75	18	3	1st	1025	50
300	75	18	3	1st	1025	100
300	75	18	3	1st	1025	150
300	75	18	3	1st	1025	200
300	75	18	3	1st	1200	50
300	75	18	3	1st	1200	100
300	75	18	3	1st	1200	150
300	75	18	3	1st	1200	200
300	75	18	3	2st	1025	50
300	75	18	3	2st	1025	100
⋮	⋮	⋮	⋮	⋮	⋮	⋮

Table 7.6: Parameter set, composed of parameter combinations

A parameter set exists of a combination of the first five parameters presented in table 7.6 and the parameters for which the hose should operate. A parameter set exists of eight combinations due to the range of the crawler and the slurry or water operating scenario.

For these parameter sets, the following results are extracted from the static Matlab model:

- Tension at specific locations
- Coordinates of specific nodes

There are 26928 combinations presented, which result in 3366 sets of parameters. Some parameter combinations fail, based on the fail criteria represented in table 7.2. If a parameter combination fails, the whole parameter set fails, so the new number of feasible parameter combinations is 12322. The next selection is based on the workability parameters presented in table 7.3 this reduces the number of combinations to 10066. If one combination of the set fails, the total parameter set fails. This selection reduces the number of combinations to 4472, resulting in 559 parameter sets.

## 7.3. Multi-criteria analysis

Subsection 7.2.1 calculates the feasible parameters sets. A multicriteria analysis (MCA) is found suitable for further assessment of the parameter sets. 'A Multi-Criteria Analysis (MCA) can be used to identify and compare different policy options by assessing their effects, performance, impacts, and trade-offs' [12]. The design leans on three parameters: the tension at certain locations, the coordinate of certain locations, and the connection point's bending. All three are important, but there is no optimal solution for these parameters. A trade-off can be made using an MCA to find certain parameters that balance these criteria.

### 7.3.1. The criteria for the MCA

There are four criteria selected for the MCA. The criteria are described following the structure of the hose. The first criterion is the coupling between the crawler and the hose, the angle may not exceed 45 degrees. The parameter combinations that exceed this value are filtered out by the approach of subsection 7.2.1. An angle of 90 degrees is an optimum situation. The score evaluation is from 1 to 10, with 1 being 45 degrees and 10 being 90 degrees. The score of the angle per parameter combination is calculated with the following formula:

$$Score = \frac{(optimal\ angle - minimal\ angle) - (angle\ evaluated - minimal\ angle)}{angle\ evaluated - minimal\ angle} * 10 \quad (7.1)$$

The second criterion is the tension in the first part of the hose until the buoyancy starts. The tension is calculated with the static Matlab model. The tension may not exceed 60 kN due to the safety criteria

on the weight of the crawler. A Tension of 0 kN is not optimal, so a minimum tension in the hose is selected in consultation with IHC. 15 kN scores a 10 and 60 kN scores a 1. The score of the maximum tension per parameter combination is calculated with the following formula:

$$\text{Score} = \frac{(\text{maximum tension} - \text{minimum tension}) - (\text{tension evaluated} - \text{minimum tension})}{\text{tension evaluated} - \text{minimal tension}} * 10 \quad (7.2)$$

The third criterion is the lowest point of the lazy s-shape. The location of the lowest point is also calculated with the static Matlab model and may not exceed 10 meters above the seafloor. There is no optimal height of the lazy s-shape, so the maximum value is 30 meters. When the location is 30 meters or higher, the score is always 10. The minimum height of 10 meters gets a score of 1, and everything above 30 meters gets a 10. The score of the height of the lazy s per parameter combination is calculated with the following formula:

$$\text{Score} = \frac{(\text{maximum height} - \text{minimum height}) - (\text{height evaluated} - \text{minimum height})}{\text{height evaluated} - \text{minimal height}} * 10 \quad (7.3)$$

The last criterion is the maximum tension in the hose. This tension can be located anywhere on the hose. It has the same approach as the tension in the first part of the hose until the buoyancy starts, equation 7.2. Only the maximum tension in the hose is stated at 100 kN instead of 60.

### 7.3.2. Weighting method

For each parameter combination, the score is calculated per criterion. In case this score is just added, all the criteria are of the same importance, the equal weighting method. A weight factor is applied to incorporate the ranking of the criteria.

The weight factors are displayed in table 7.7.

Criteria	Weight factor
Angle crawler hose	2
Tension first part hose	3
Lowest location lazy s shape	1
Maximum tension hose	2

Table 7.7: Weight factors of MCA

The factors are chosen by the initial design criteria of chapter 3.2. An important limitation of the design is the bending moment and shear force exceeded by the hose on the crawler. The shear force is the biggest failure criteria because all parameter sets that exceed the minimum height of the lowest location on the lazy s-shape are already filtered out in subsection 7.2.1. Suppose the maximum tension is an extension of the crawler tension parameter and so is less important but still important. There is no bending or bend restrictor modelled in the static Matlab design, so the hose angle is not represented accurately. Still, a small angel is not desirable.

### 7.3.3. Performance of the MCA

The criteria grades are multiplied with the weight factors and added per parameter combination, creating a score per parameter combination. The scores of the eight parameter combinations are also added to get a score for the parameter set. There are multiple ways to select and rank these scores of the MCA:

- Selecting the best score of all the parameter combinations
- Selecting the best score of all the parameter sets
- Selecting the worst score of a parameter combination within a parameter set. Ranking them from best to worst

- Selecting the worst score of a parameter combination within a parameter set. Rankin them based on the best score of the total parameter set.

These four options have been discussed with IHC, and the third option is selected because the best parameter combinations do not result in a good operating parameter set. Selecting the best parameter set is a good start but could contain bad performing parameter combinations.

The worst-performing parameter combinations are selected by filtering out the worst operating parameter combinations of parameter sets. By ranking the filtered set from best to worst the best of the least favourable parameter combination of a set is selected. Meaning that the rest of the set is performing better.

The 25 best presenting worst combinations are rearranged on the total score of the parameter set. This ranking results in the order shown in Table 7.8. The selected parameter set is a hose of 300 meters, VTS located at 100 m above the seafloor, 24 modules, 3 clamped per hose starting after section 1.

Rank	Length [m]	Height [m]	Modules [pieces]	Clamped [per segment]	Starting after [segment]	Hose substance [kg/m <sup>3</sup> ]	Distance to VTS [m]	Score combination	Score set
1	300	100	24	3	1		200	61,29	531,51
2	300	125	23	3	1		200	60,94	532,49
3	350	100	25	3	1		200	59,96	537,58
4	300	125	26	3	1		200	59,75	510,65
5	300	125	24	3	1		200	59,64	532,49
6	300	75	26	3	1		200	59,33	519,82
7	300	100	23	3	1		200	59,24	515,88
8	300	125	22	3	1		200	59,17	532,79
9	350	125	25	3	1		200	58,63	501,69
10	350	100	28	3	1		200	58,49	504,53

Table 7.8: Ranking parameter sets based on lowest score of a parameter set

After discussion with IHC, a second ranking was made. The 25 best presenting worst combinations are rearranged on the total score of the parameter set. This ranking is shown in Table 7.9.

Rank	Length [m]	Height [m]	Modules [pieces]	Clamped [per segment]	Starting after [segment]	Hose substance [kg/m <sup>3</sup> ]	Distance to VTS [m]	Score combination	Score set
1	300	100	25	3	1		200	59,96	537,58
2	300	75	28	3	1		200	56,55	534,21
3	300	125	22	3	1		200	59,17	532,79
4	300	125	23	3	1		200	60,94	532,49
5	300	125	24	3	1		200	59,64	532,49
6	300	75	27	3	1		200	57,94	532,27
7	300	100	24	3	1		200	61,29	531,51
8	300	125	25	3	1		200	58,30	531,00
9	300	100	26	3	1		200	57,75	530,90
10	300	100	27	3	1		200	56,35	528,25

Table 7.9: Reranking 25 parameter set, first 10 presented

The selected parameter set is a hose of 300 meters, VTS located at 100 m above the seafloor, 25 modules, 3 clamped per hose starting after section 1. This set does not differ very much from the first selected set. The maximum score of a parameter set is 537,58.



## 7.4. Sensitivity study parameters

The optimal parameter set is presented below, in figure 7.8. This set consists of a hose length of 320.04 meters; the connection with the VTS is 100 meters above the seafloor. There are 25 modules present, starting after section one with three modules clamped per section.

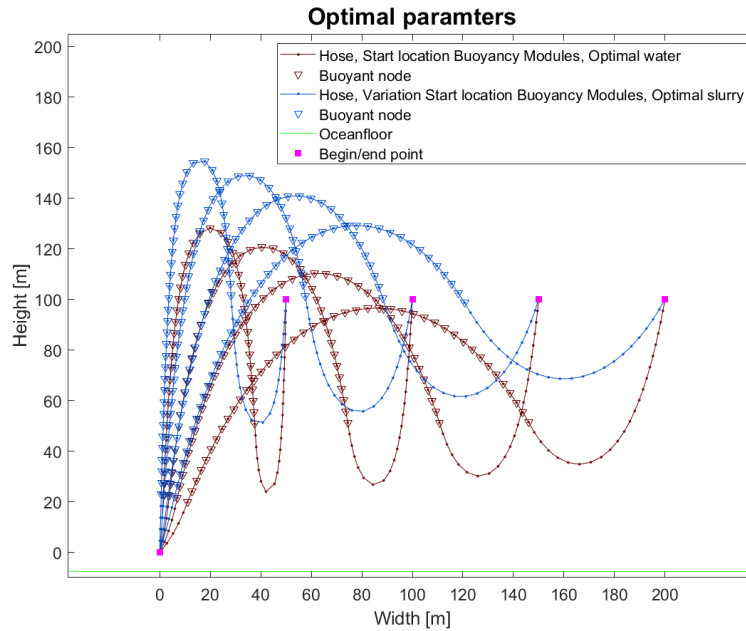


Figure 7.8: Optimal parameter set

### 7.4.1. Sensitivity of length parameter

The sensitivity of the parameters is evaluated per parameter. The first parameter evaluated is the length of the hose. In figure 7.9 and 7.10, the visual plots of the static equilibrium are displayed. Table 7.10 shows the MCA scoring. If the hose is scaled up in size, the height of the lazy s drops significantly.

Width	Hose substance	Score MCA: combinations		Score MCA: sets	
		300 meter	350 meter	300 meter	350 meter
50	Slurry	76,22	NaN	537,58	NaN
100	Slurry	75,16	NaN	537,58	NaN
150	slurry	72,29	NaN	537,58	NaN
200	Slurry	66,39	NaN	537,58	NaN
50	Water	62,87	NaN	537,58	NaN
100	Water	62,70	NaN	537,58	NaN
150	Water	61,99	NaN	537,58	NaN
200	Water	59,96	NaN	537,58	NaN

Table 7.10: MCA results for sensitivity study length

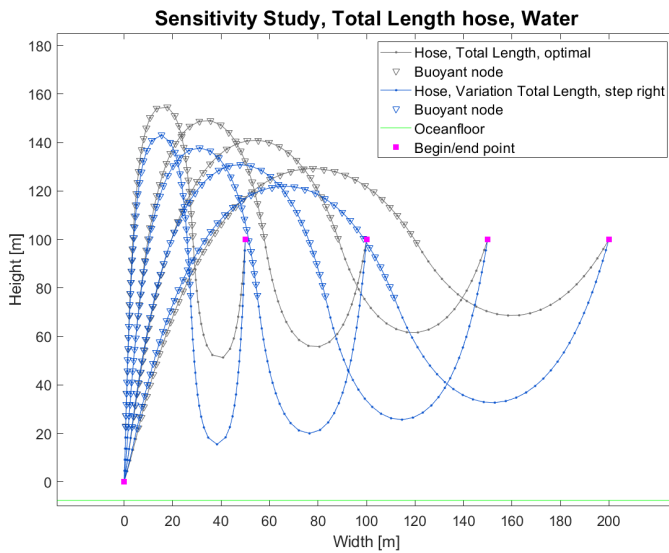


Figure 7.9: Sensitivity length hose (water)

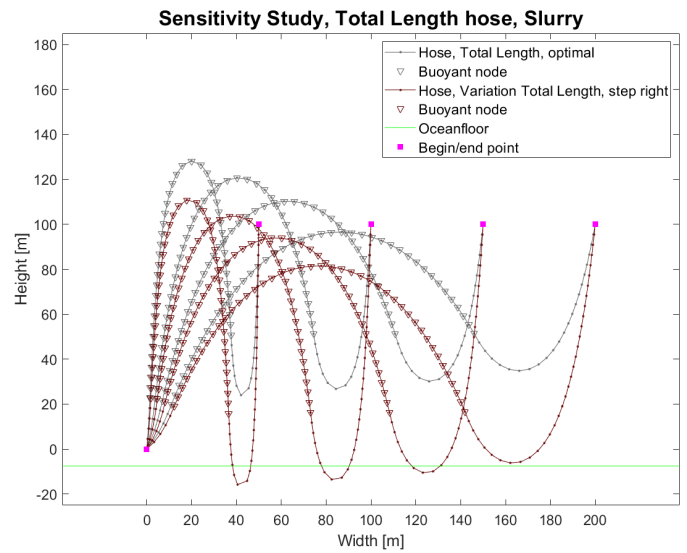


Figure 7.10: Sensitivity length hose (slurry)

### 7.4.2. Sensitivity of height VTS parameter

In figure 7.11 and 7.12, the visual plots of the static equilibrium of the height variation are displayed. Table 7.11 shows the MCA scoring. A lower connection point performs better with water, while a higher connection point performs better with the slurry substance. The overall set scores are lower than optimal. The visual difference shows a height difference corresponding to the connection difference.

Width	Hose substance	Score MCA: combinations			Score MCA: set		
		75	100	125	75	100	125
50	Slurry	NaN	76,22	75,75	NaN	537,58	531,00
100	Slurry	NaN	75,16	74,85	NaN	537,58	531,00
150	Slurry	NaN	72,29	72,29	NaN	537,58	531,00
200	Slurry	NaN	66,39	66,74	NaN	537,58	531,00
50	Water	NaN	62,87	61,38	NaN	537,58	531,00
100	Water	NaN	62,70	61,20	NaN	537,58	531,00
150	Water	NaN	61,99	60,48	NaN	537,58	531,00
200	Water	NaN	59,96	58,30	NaN	537,58	531,00

Table 7.11: MCA results for sensitivity study connection point VTS

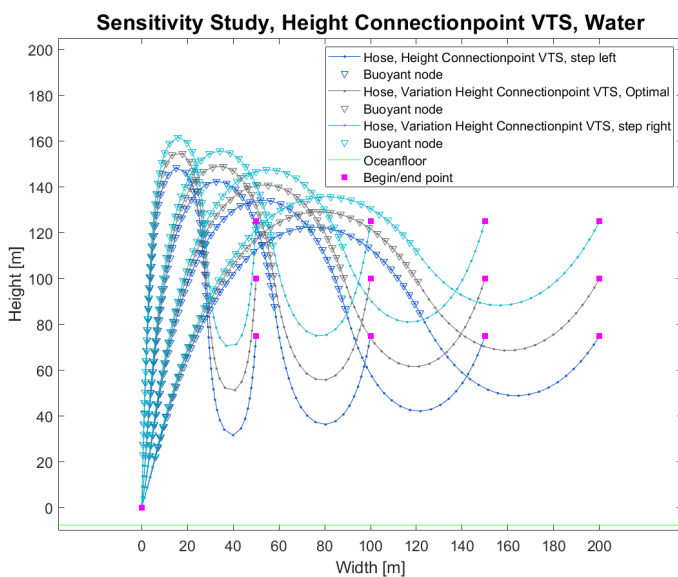


Figure 7.11: Sensitivity height connection VTS (water)

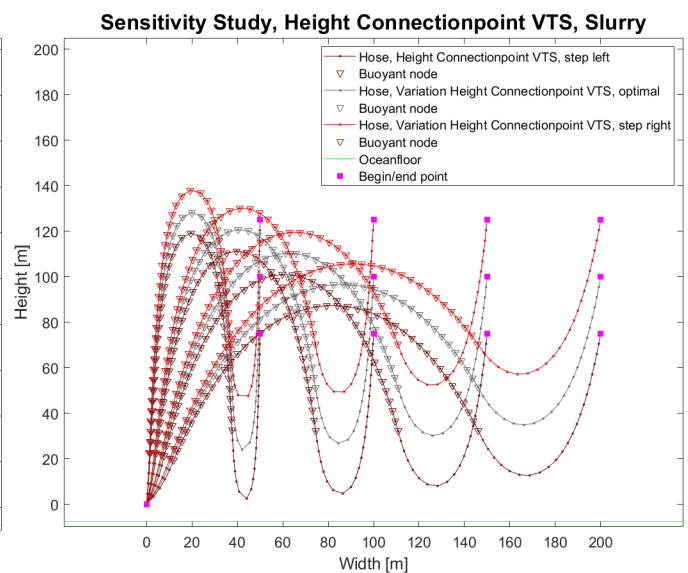


Figure 7.12: Sensitivity height connection VTS (slurry)

### 7.4.3. Sensitivity of modules parameter

In figure 7.13 and 7.14, the visual plots of the static equilibrium of the module variation are displayed. Table 7.12 shows the MCA scoring. A lower number of modules performs better when the hose is filled with water. The same applies to a hose filled with slurry and a higher number of modules. The overall set scores are a little bit lower than the optimal. The visual difference shows a height difference corresponding to the connection difference.

Width	Hose substance	Score MCA: combinations			Score MCA: set		
		24	25	26	24	25	26
50	Slurry	72,01	76,22	76,23	531,51	547,04	530,91
100	Slurry	71,94	75,16	75,39	531,51	547,04	530,91
150	Slurry	70,11	72,29	73,03	531,51	547,04	530,91
200	Slurry	64,15	66,39	67,91	531,51	547,04	530,91
50	Water	64,39	62,87	60,52	531,51	547,04	530,91
100	Water	64,19	62,70	60,37	531,51	547,04	530,91
150	Water	63,43	61,99	59,71	531,51	547,04	530,91
200	Water	61,29	59,96	57,75	531,51	547,04	530,91

Table 7.12: MCA results for sensitivity study total number of modules

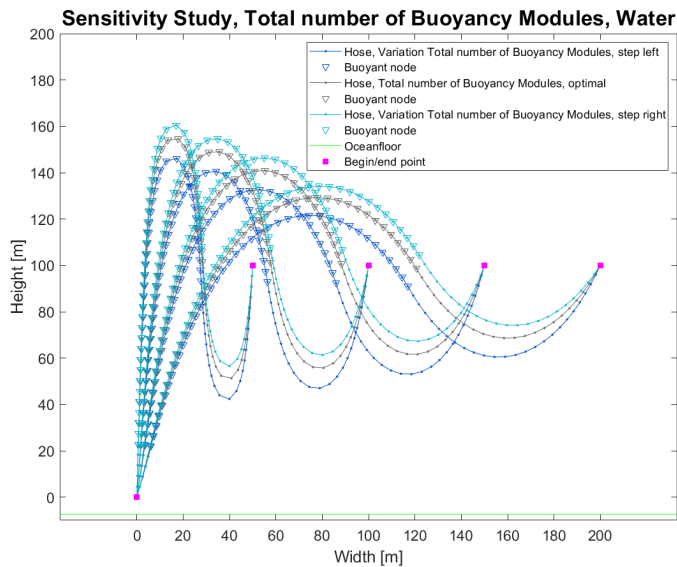


Figure 7.13: Sensitivity modules (water)

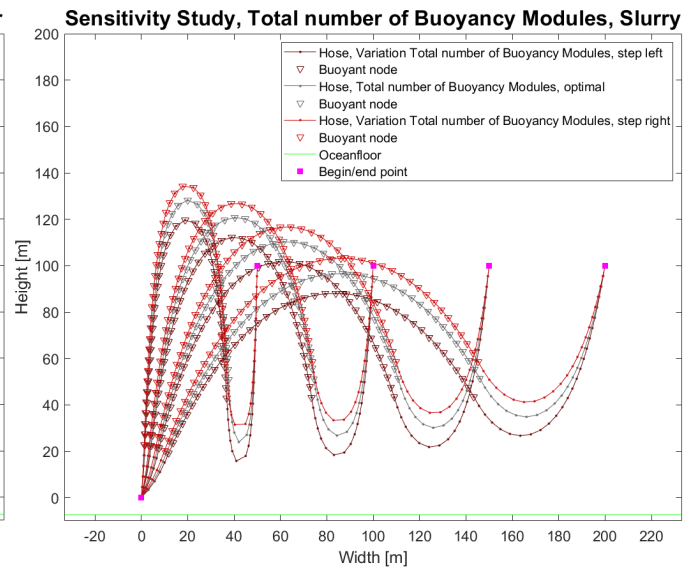


Figure 7.14: Sensitivity modules (slurry)

7.4.4. Sensitivity of distribution modules parameter

In figure 7.15 and 7.16 the visual plots of the static equilibrium of the module clamped per section and the variation are displayed. Table 7.13 shows the MCA scoring. Scaling up this parameter results in a failure based on the failure criteria. The substance slurry options result in a lazy s-shape that reaches the range of the 10 meter distance between the seafloor and the lazy s.

Width	Hose substance	Score MCA: combinations		Score MCA: sets	
		3 modules per hose	4 modules per hose	3 modules per hose	4 modules per hose
50	Slurry	76,22	NaN	537,58	NaN
100	Slurry	75,16	NaN	537,58	NaN
150	Slurry	72,29	NaN	537,58	NaN
200	Slurry	66,39	NaN	537,58	NaN
50	Water	62,87	NaN	537,58	NaN
100	Water	62,70	NaN	537,58	NaN
150	Water	61,99	NaN	537,58	NaN
200	Water	59,96	NaN	537,58	NaN

Table 7.13: MCA results for sensitivity study clamped modules per segment

Sensitivity Study, Number of Buoyancy Modules per Hose segment, Water

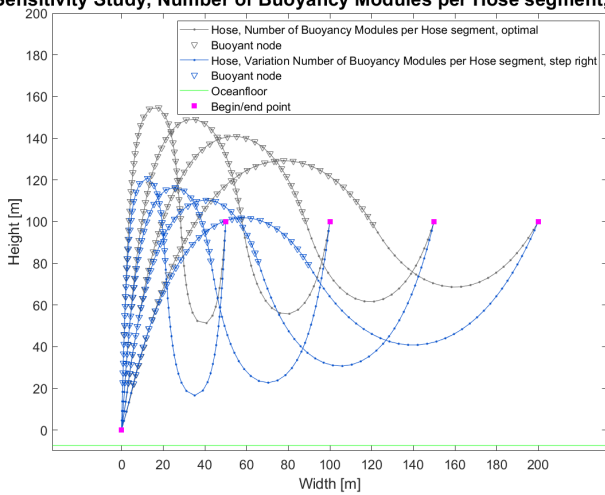


Figure 7.15: Sensitivity modules clamped per hose segment (water)

Sensitivity Study, Number of Buoyancy Modules per Hose segment, Slurry

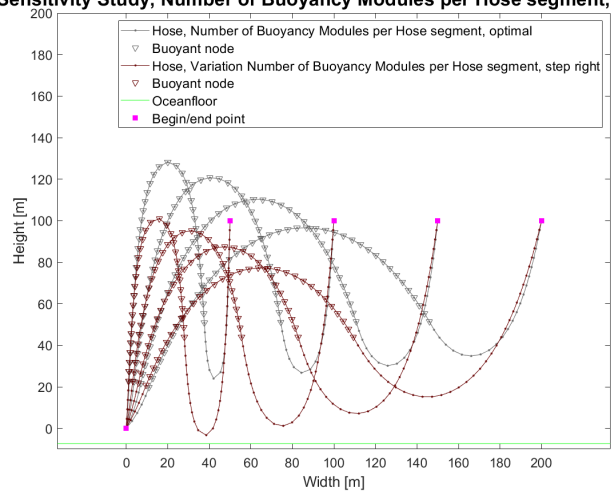


Figure 7.16: Sensitivity modules clamped per hose segment (slurry)

### 7.4.5. Sensitivity of starting parameter

In figure 7.17 and 7.18 the visual plots of the static equilibrium module clamped per section and the variation are displayed. Table 7.14 shows the MCA scoring. Scaling up this parameter results in a failure based on the failure criteria. The substance slurry options result in a starting section with an angle that oversteps the fail criteria on this angle.

Width	Hose substance	Score MCA: combinations		Score MCA: sets	
		Start at the second segment	Start at the third segment	Start at the second segment	Start at the third segment
50	Slurry	76,22	NaN	537,58	NaN
100	Slurry	75,16	NaN	537,58	NaN
150	Slurry	72,29	NaN	537,58	NaN
200	Slurry	66,39	NaN	537,58	NaN
50	Water	62,87	NaN	537,58	NaN
100	Water	62,70	NaN	537,58	NaN
150	Water	61,99	NaN	537,58	NaN
200	Water	59,96	NaN	537,58	NaN

Table 7.14: MCA results for sensitivity study segment buoyancy modules

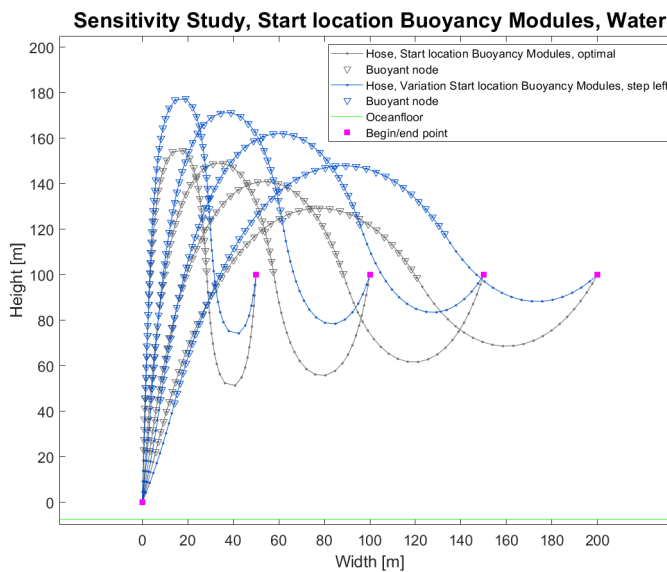


Figure 7.17: Sensitivity starting segment buoyancy modules (water)

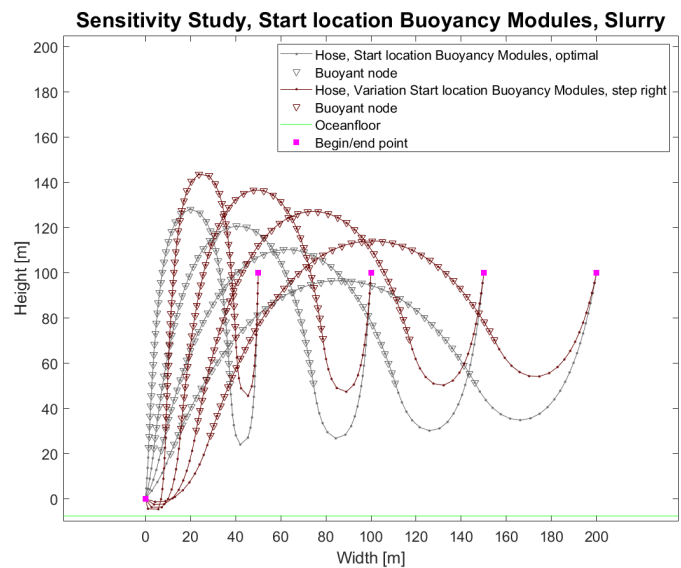


Figure 7.18: Sensitivity starting segment buoyancy modules (slurry)

## 7.5. New design parameter set for dynamic analysis

The optimal parameter set arises from the MCA, and this concluded in

- 300 meter long hose
- VTS connection 100 meter above the seafloor
- 25 buoyancy modules
- 3 buoyancy modules clamped per hose section
- Buoyancy modules starting after first hose section

Figure 7.19 shows the shape of the hose for the selected parameters. This chapter builds up to and answers the third research question, “*How is the new/optimal set of parameters identified?*” and is summarised in the next subsection.

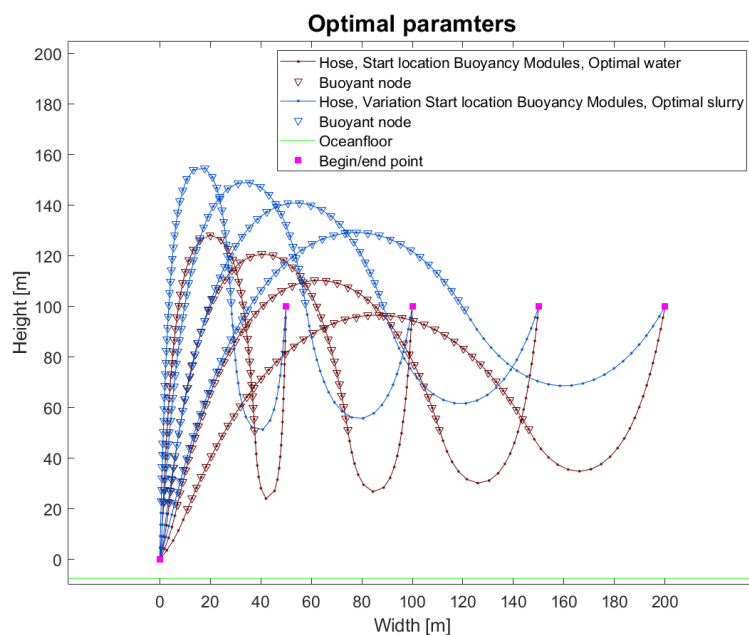


Figure 7.19: Optimal parameter set

### 7.5.1. Research question 3 and summary of the parameter study

To answer the research question, “*How is the new/optimal set of parameters identified?*” the following approach is detailed.

The parameter study is based on the 7 selected parameters presented in subsection 7.1.3. These parameters are combined and evaluated with the failure criteria that determine the performance of the parameter combination and the corresponding set.

- The bending moment is limited to 120 kNm but is not evaluated in the static analysis due to that the bending stiffness is not implemented in the static model
- The lift force at the crawler end is 200 kN, so the crawler can not be lifted from the seafloor
- The height of the lowest point of the s-shape should not hit the seafloor in any situation
- Angle between hose and crawler can not be lower than 45 degrees due to breakage and the pullover of the crawler

The parameters are also separately evaluated to investigate the influence and the combination of parameters. The influence of combinations is not included in the separate evaluation.

- The length has a major influence on the weight of the hose. This mainly influences the number of modules and modules per section parameter because these carry the weight
- The connection point of the VTS has a small influence on the shape of the hose, and the criteria lowest point s-shape and angle of the connection points
- The total number of modules influences the shape but also depends on the length of the hose and modules per section.
- The starting segment determines the shape, the tension in the hose and collision with the seafloor. This parameter heavily influences if a parameter combination complies or fails
- Every parameter combination must always operate for the given range of the crawler and hose substance.

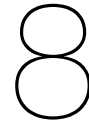
With the separated parameter analysis, the parameters are ranked to determine which parameter matters the most and guides the parameter combinations because of the limits and collaboration between certain parameters. For example, for a 300 meter long hose with 3 modules per segment, it is impossible to have more than 36 modules due to the space available for modules. For this combination, the variation of starting segment is not possible. These combinations of are determined in table 7.4 and result in a parameter coverage per parameter (table 7.5).

This range of variation per parameter results in 26928 combinations and these combinations are tested on the failure criteria. If a combination fails, a whole set fails. These are also filtered out and result in 59 sets of parameters evaluated with the multicriteria analysis.

The MCA is based on a weighted method, giving more weight to the critical requirements of the model. At last, the selection of a parameter set is determined. There are four options given for selecting the optimal parameter set. The worst score of a parameter combination within a set and then ranking these from high to low is chosen. This choice resulted in table 7.8. In elaboration with IHC, this table is then rearranged based on the score of the parameter set from high to low, resulting in the optimal parameter set.

This optimal parameter set is then evaluated with a visual sensitivity study to ensure that the closely related parameters set performs worse than the optimal set.





# Transition from Matlab to OrcaFlex

The static MatLab model is a starting point to elaborate the design with its new set of parameters. This chapter gives the background of implementing the MatLab model in OrcaFlex software. The implementation in Orcaflex is the base model and is improved by implementing design parameters that were omitted. This results in a better approximation of the design.

Subchapter 8.1 focuses on the implementation of the MatLab model in OrcaFlex, with the same approximations and design features. This model is then compared to the initial MatLab results. In subchapter 8.2 the omitted and simplified parameters are inserted step by step. Their influence is evaluated, resulting in possible alterations to the design parameters. Subchapter 8.3 presents the new design parameters and summarises the decisions and issues. Finally, an answer is given to research question 5, “*What are the results of the static analysis using the dynamic software?*”

## 8.1. Matlab Model implemented in OrcaFlex

In this chapter, the Matlab model set-up is replicated in OrcaFlex. The Matlab model is used to estimate and select the concept parameters of the jumper hose design. The Orcaflex model implements the parameters left out of the Matlab model. Orcaflex expands to more complex design parameters simplified in the Matlab model and conducts the dynamic analysis of the model. Therefore it is interesting how these models relate to each other and if it is valid to adopt the concept parameters to the OrcaFlex model.

Chapter 7 elaborates the theory and method used to create the parameter study, composed of the parameter selection, parameter combinations and the multi-criteria analysis. This parameter study resulted in a set of parameters for the concept design of the jumper hose. The concept design parameters are presented in table 8.1 and 8.2.

Parameter	Value	Unit
Total length of the hose	320,04	[m]
Segment hose length	22,86	[m]
Number of elements to evaluate segment	5	[-]
Element length	4,572	[m]
Height of VTS connection above seafloor	100	[m]
Range of crawler	200	[m]
Number of buoyancy modules	25	[-]
Buoyancy modules per hose segment	3	[-]
Starting point buoyancy	After 1st hose	
Content hose/Slurry density	1200	[kg/m <sup>3</sup> ]

Table 8.1: Concept design parameters, subchapter 7.5

With material hose parameters:

Parameter	Value	Unit
ID	300	[mm]
OD	473,3	[mm]
Material hose (Weight)	137	[kg/m]
Axial stiffness hose	20,408 e3	[kN]

Table 8.2: Design parameters jumper hose

### 8.1.1. Objects Orcaflex model

The OrcaFlex model is built with the line object and constraints objects. The line object represents a massless linear element with lumped masses on each end (figure 5.1 and subsection 5.2.1). This element only has axial and torsional properties and these masses carry other properties (mass, weight, buoyancy, etc.). This structure of the line matches the structure of the MatLab model.

The OrcaFlex line is constructed of 14 sections of 22,86 meter long hose. These sections are split up into five segments of 4,572 meters. This line has a force of 406,83 N per meter:

$$\begin{aligned}
 F_{weight} &= F_{hosematerial} + F_{contenthose} - F_{buoyancy} \\
 &= mass_{hose} * g + mass_{slurry} * A_{IDhose} * g - mass_{water} * A_{ODhose} \\
 &= 1343,5 \text{ N/m} + 831,8 \text{ N/m} - 1768,5 \text{ N/m} \\
 &= 406,8 \text{ N/m}
 \end{aligned} \tag{8.1}$$

The line also has the axial stiffness of table 8.3 implemented, the bending and torsional stiffness set to zero. At the end of each segment, a pinned joint is placed with zero bending stiffness, constructing a node with no resistance against rotation.

Pin Joint	z-position
1	4,57 m
2	9,14 m
⋮	⋮
68	310,90 m
69	315,47 m

Table 8.3: Pinned joint locations

25 modules with three modules per section result in 8,33 buoyant sections. This number of sections is converted to segments, and they are then converted to nodes:

$$\begin{aligned}
 \text{Number of section} &= \frac{25}{3} = 8,33 \text{ sections} \\
 \text{Number of segment} &= \frac{25}{5} = 41,66 \text{ segments} = 42 \text{ segments} \\
 \text{Number of nodes} &= 42 + 1 = 43 \text{ nodes}
 \end{aligned} \tag{8.2}$$

Each module lifts 427 kg (4187,4 N), resulting in a total buoyancy force of 104,67 kN. In subsection 6.2.2, it is stated that this buoyancy force is equally divided over the buoyant nodes. Resulting in 2,435 kN per buoyant node. This value is implemented in the line object as an applied load, table 8.4.

Node number	Buoyant node	Z-position	Applied force
6	1	22,86 m	2,435 kN
7	2	27,43 m	2,435 kN
⋮	⋮	⋮	⋮
47	42	210,31 m	2,435 kN
48	43	214,88 m	2,435 kN

Table 8.4: Buoyant nodes

The ends of the line object are connected to constraints which are fixed in all the DOF directions. For the two-step static calculation, the first step is done by the catenary analysis. The second step is a full statics calculation. This calculation is the same principle as used in the MatLab model.

### 8.1.2. Results OrcaFlex model

Pressing the run statics button results in the following shape of the hose. Reviewing figure 8.1, the shape of the OrcaFlex and MatLab calculation match. The node locations are compared in Appendix E.

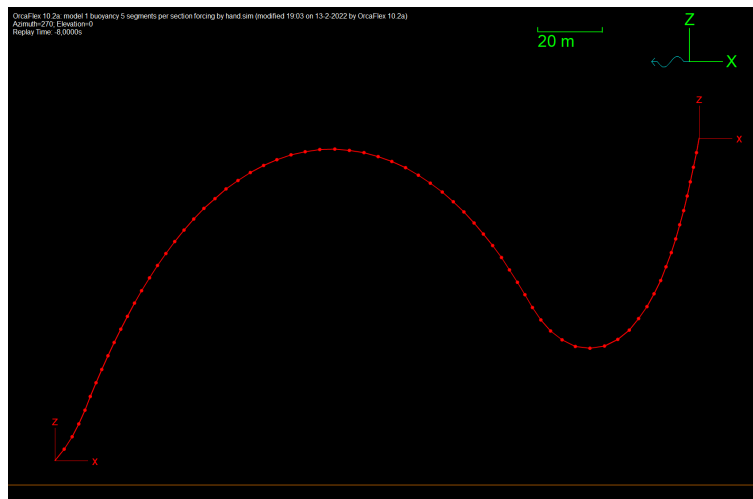


Figure 8.1: Design of subchapter 7.5 implemented in OrcaFlex

Another verification parameter is the tension in the elements. The MatLab tension is calculated with formulas 4.9, 4.10, 4.11 and 4.15 and the OrcaFlex axial tension is calculated with formula D.1. The difference per element is displayed in Appendix E and the maximal difference is 1,256 kN in node 5 (relative from the left connection point). This difference is relatively small compared to the axial tension of the element (15,6472 kN).

Comparing the OrcaFlex (figure 8.1) and MatLab (figure 7.19, blue line, width 200 m) model visually, on node locations (Appendix E) and in tension calculation (Appendix E) it can be concluded that both models appear to be similar. This confirms that the concept parameter study of subchapter 4.3 can be used as a basic assumption for the static and dynamic calculations.

## 8.2. The enlargement of the design parameters

The model presented in subchapter 4.3 is a simplified version of the real-world system. With OrcaFlex, it's possible to improve the engineering of the object. The following objects can gain much improvement:

- The buoyancy modules
- The segment length
- Implementation of parameters like bending stiffness
- The hose structure

Each option is elaborated and evaluated in the next subchapters with the MatLab base model as the starting point.

### 8.2.1. The buoyancy modules

In the simplified base model, figure 8.1, buoyancy is implemented on nodes by dividing the buoyancy force over the buoyant segments. In that model, many design parameters are not taken into consideration.

The design parameters of the buoyancy are displayed in table 3.9 and resulted in a buoyancy module design in OrcaFlex of the following parameters, see table 8.5.

Parameter	Value	Unit
Mass	427	[kg]
Volume	1	[m <sup>3</sup> ]
Drag area x & y	1	[m <sup>2</sup> ]
Drag area z	1	[m <sup>2</sup> ]
Drag coefficient x & y	1,1	[-]
Drag coefficient z	1,1	[-]
Added mass coefficient x & y	1	[-]
Added mass coefficient z	1	[-]

Table 8.5: Design parameters buoyancy modules in OrcaFlex

These modules were implemented on the hose base design and evaluated, resulting in the following model. The red model is the base design of MatLab.

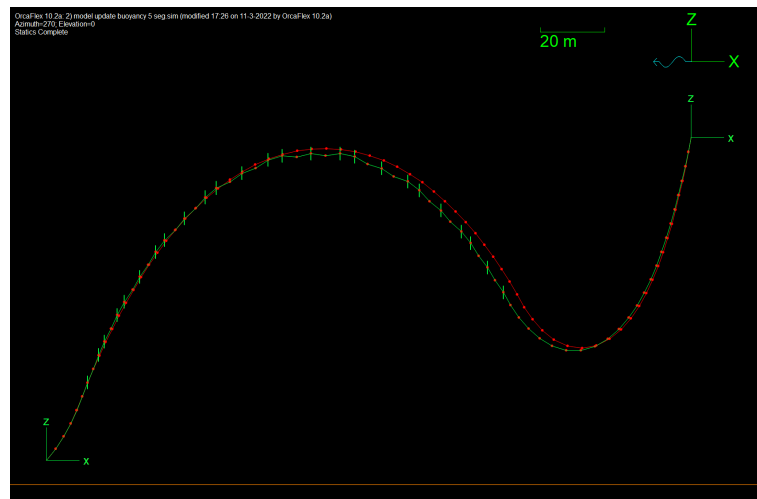


Figure 8.2: Implementing buoyancy modules in stead of applied forcing on the nodes

Figure 8.2 clearly shows that the initial design parameter of the buoyancy modules (table 8.5) does not correspond with the implemented buoyancy force of the MatLab base model. This difference is attributed to the fact that the volume of the module does not correspond to the lift force per module. The ratio volume weight is approximately 2:1. A detailed design of the buoyancy modules is absent in the provided preliminary design documents of the jumper hose, document [15]. In consultation with

Royal IHC, the following design parameters are adopted for the buoyancy modules, table 8.6.

$$\begin{aligned}
 F &= V * \rho * g \\
 m * a &= V * \rho * g \\
 m &= V * \rho \\
 V &= \frac{m}{\rho} = \frac{427}{1025} = 0,4166 \text{ m}^3
 \end{aligned}
 \tag{8.3}$$

Resulting in the following design parameters,

Parameter	Value	Unit
Mass	427	[kg]
Volume	0,8332	[m <sup>3</sup> ]
Drag area x & y	1	[m <sup>2</sup> ]
Drag area z	1	[m <sup>2</sup> ]
Drag coefficient x & y	1,1	[-]
Drag coefficient z	1,1	[-]
Added mass coefficient x & y	1	[-]
Added mass coefficient z	1	[-]

Table 8.6: Updated design parameters buoyancy modules in OrcaFlex

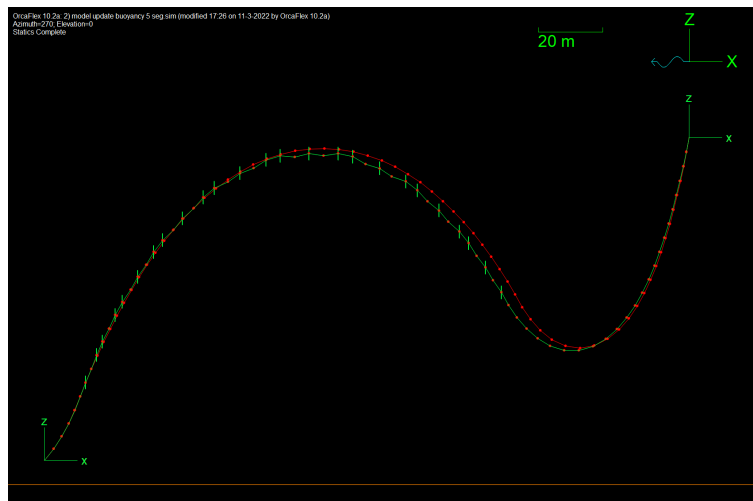


Figure 8.3: Implementing buoyancy modules in stead of applied forcing on the nodes

Figure 8.3 shows that the shape of the hose comes close to the base design of the hose. The next design update is to implement the drag area and update the volume parameter to the dimensions of table 3.9:

$$\begin{aligned}
 V_{OD} &= \pi * r^2 * H = \pi * \left(\frac{1,41}{2}\right)^2 * 1,2 = 1,874 \text{ m}^3 \\
 V_{ID} &= \pi * r^2 * H = \pi * \left(\frac{0,3}{2}\right)^2 * 1,2 = 0,212 \text{ m}^3 \\
 V_{buoy} &= 1,662 \text{ m}^3 \\
 F_{Volume} &= V_{buoy} * \rho * g = 1,662 * 1025 * 9,81 = 16706 \text{ N} \\
 F_{lift} &= 4187,44 \text{ N} \\
 F_{weight} &= 16706 - 4187,44 = 12518,6 \text{ N} = 1276,54 \text{ kg}
 \end{aligned}
 \tag{8.4}$$

The shape of the buoyancy model is a cylinder so the drag area becomes:

$$\begin{aligned}
 A_x &= A_{circle\ OD} - A_{circle\ ID} \\
 A_x &= \pi * r_{OD}^2 - \pi * r_{ID}^2 = \pi * \frac{1,41^2}{2} - \pi * \frac{0,3^2}{2} = 1,385\ m^2 \\
 A_z &= H * (2\pi * r_{OD}) = 1,2 * 2 * \pi * \frac{1,41}{2} = 5,315\ m^2
 \end{aligned}
 \tag{8.5}$$

Resulting in the following design parameters (table 8.7)

Parameter	Value	Unit
Mass	427	[kg]
Volume	0,8332	[m <sup>3</sup> ]
Drag area x & y	1,385	[m <sup>2</sup> ]
Drag area z	5,315	[m <sup>2</sup> ]
Drag coefficient x & y	1,1	[-]
Drag coefficient z	1,1	[-]
Added mass coefficient x & y	1	[-]
Added mass coefficient z	1	[-]

Table 8.7: Parameters

The Drag area in x and y are the same due to the symmetry of the cylinder. In consultation with Royal IHC, the Drag coefficients  $C_x$ ,  $C_y$  and  $C_z$  are set to 1,1 and added mass coefficients  $D_x$ ,  $D_y$  and  $D_z$  are set to 1.

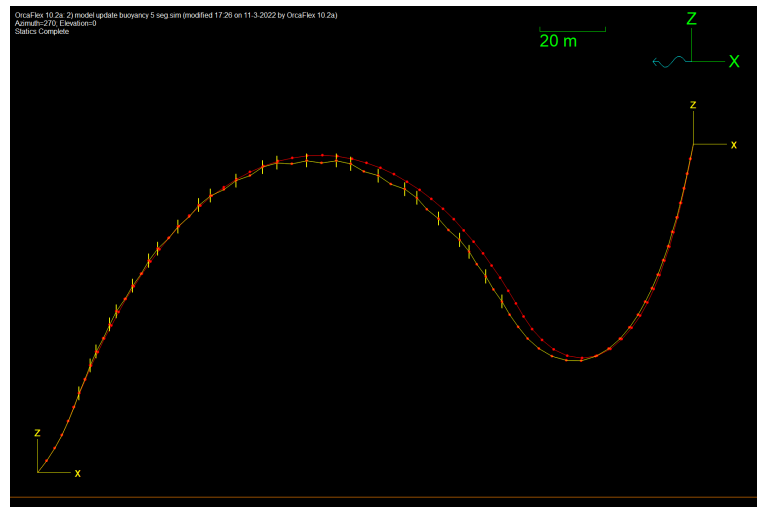


Figure 8.4: Implementing buoyancy modules in stead of applied forcing on the nodes

The modelling of the buoyancy modules results in a localised force on the hose, which influences the shape. There is a difference in node position between the MatLab model (red line, figure 8.4) and the implemented buoyancy model (yellow line, figure 8.4). This difference arises from the different methods of force implementation. The MatLab model implements force over the buoyant sections on all these nodes, and the implemented buoyancy model has buoyancy modules which are a localised force.

### 8.2.2. The segment length

The segment length is determined in subsection 6.1.4. However, this value is applied to the MatLab base model. The same principle also applies to the OrcaFlex models. The more segments used

to evaluate a section, the better the accuracy, but it needs a longer calculation time. This needs to be balanced, keeping in mind that a dynamic calculation needs way more calculations than a static analysis. The following scenarios are investigated

- The original five segments per section
- Doubling the number of segments per section
- Setting the segment length to one meter
- Reducing the original number of segments

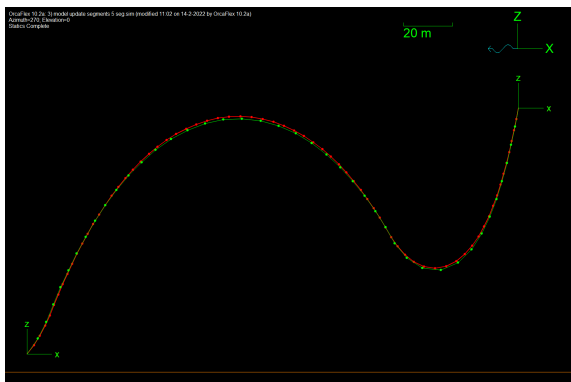


Figure 8.5: Base model, 3 segments per section vs 5

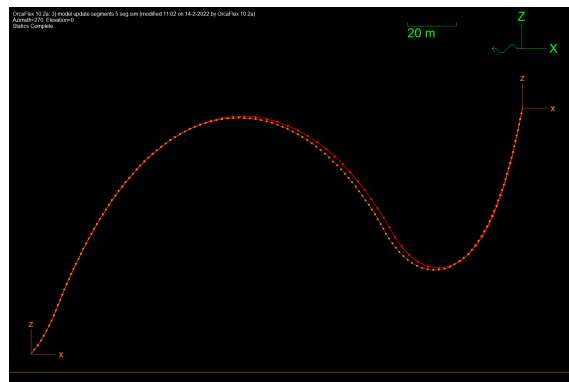


Figure 8.6: Base model, 10 segments per section vs 5

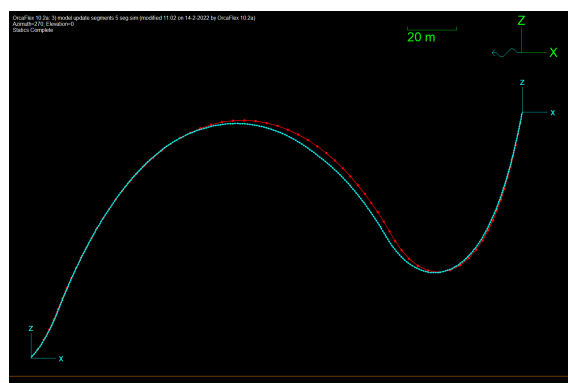


Figure 8.7: Base model, 23 segments per section vs 5

The 23 segment option gives a warning over foldbacks in the lines, resulting from an unstable equilibrium position of the nodes. This foldback is linked to the calculation method and the composition of the model. This warning is addressed in a later section. For the models with a smaller segment length, the shape is more fluid than for the wider segments. Also, the calculation time increases with adding more segments to a section.

Model	Simulation time
3 seg	1,54 sec
5 seg	2,03 sec
10 seg	3,54 sec
23 seg	7,56 sec

Table 8.8: Simulation time

The segment length also depends on the construction/structure of the hose and the added stiffeners. Figure 8.5 shows that a large segment length results in an angular shape with kinks at the end of the segments.

### 8.2.3. Bending stiffness

The base model is built with segments and nodes. These nodes can rotate freely with their segments, but this does not represent the real-world system. The next models represent the steps to evaluate the influence of the bending stiffness of the hose, starting with the removal of the pinned nodes. It's important to evaluate the system over the full range: 50 m to 200 m horizontal distance between VTS and crawler (see Figures 8.8-8.20, Table 8.9).

#### Horizontal distance crawler to VTS, 200 m

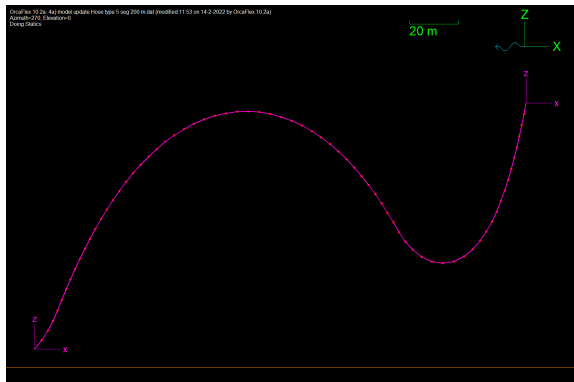


Figure 8.8: Hose with pinned nodes vs without pinned nodes

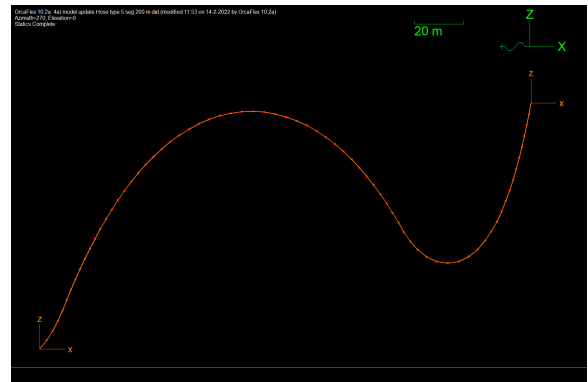


Figure 8.9: Continuous hose structure vs Matlab model

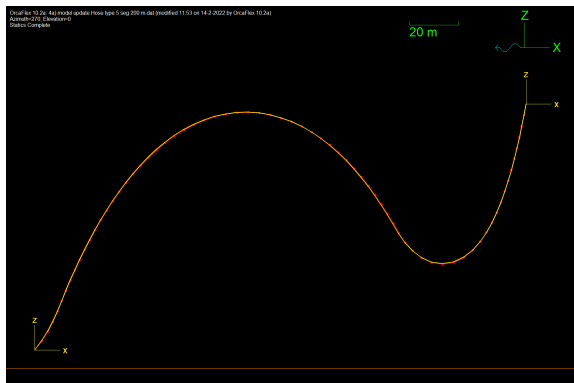


Figure 8.10: Bending stiffness incorporated vs MatLab model

Line colour	Representation	Figures
Red	MatLab model	figure 8.8 to 8.20
Pink	no pinned nodes	figure 8.8, 8.11, 8.14 and 8.17
Orange	Structure hose 1 section ipv 14	figure 8.9, 8.12, 8.15 and 8.18
Yellow	Bending Stiffness	figure 8.10, 8.13, 8.16, 8.19 and 8.20

Table 8.9: Colour labels, with corresponding figures



**Horizontal distance crawler to VTS, 150 m**

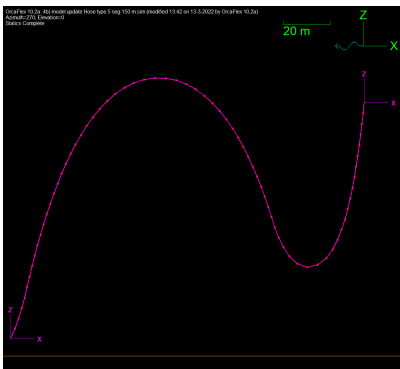


Figure 8.11: Hose with pinned nodes vs without pinned nodes

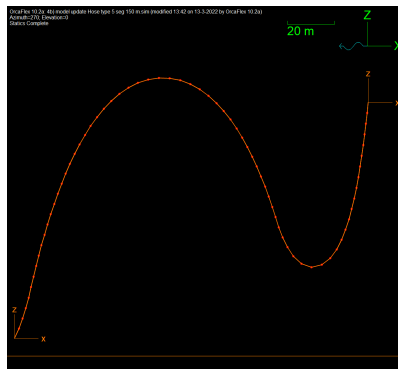


Figure 8.12: Continuous hose structure vs Matlab model

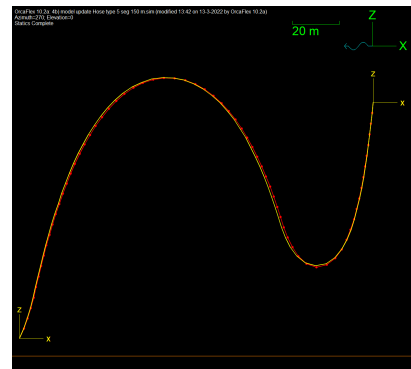


Figure 8.13: Bending stiffness incorporated vs MatLab model

**Horizontal distance crawler to VTS, 100 m**

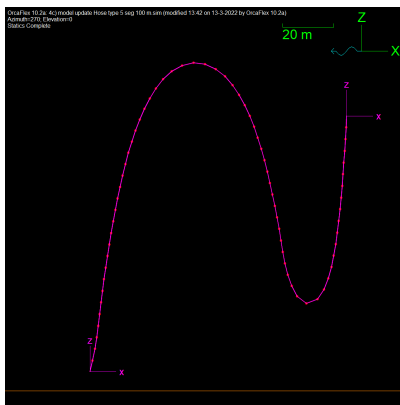


Figure 8.14: Hose with pinned nodes vs without pinned nodes

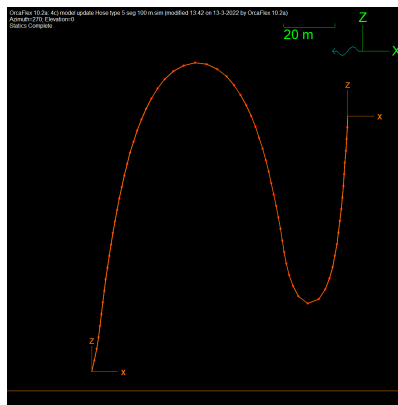


Figure 8.15: Continuous hose structure vs Matlab model

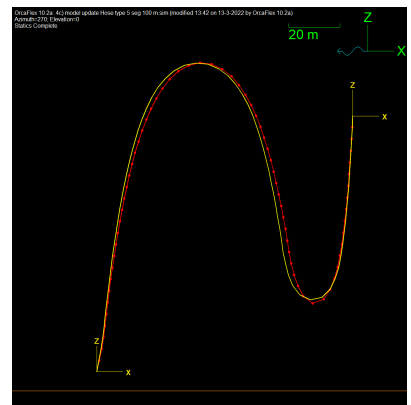


Figure 8.16: Bending stiffness incorporated vs MatLab model

**Horizontal distance crawler to VTS, 50 m**

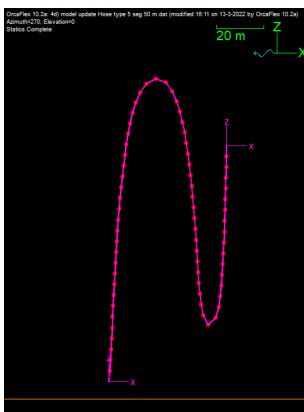


Figure 8.17: Hose with pinned nodes vs without pinned nodes

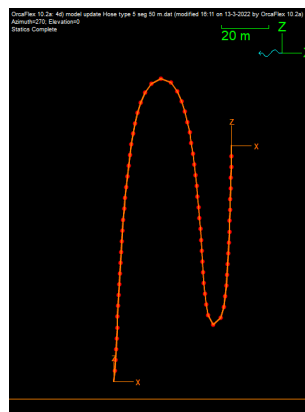


Figure 8.18: Continuous hose structure vs Matlab model

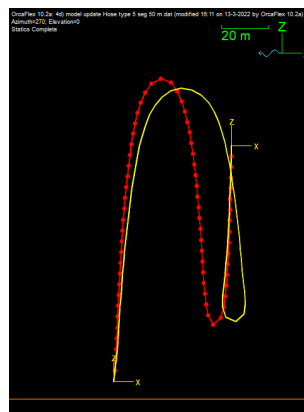


Figure 8.19: Bending stiffness incorporated vs MatLab model

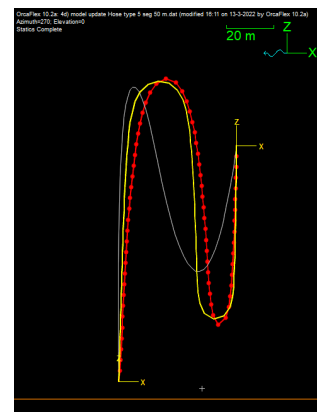


Figure 8.20: Bending stiffness incorporated vs MatLab model, spline method

Looking at the series of OrcaFlex plots, the removal of the pinned joints does not influence the shape of the hose. The same goes for the change in the number of sections. As described in subsection 5.2.1, OrcaFlex evaluates a line object by segments constructing a line with segments and nodes. These nodes are not drawn in the model, but clear kinks are visible. Adding the bending stiffness to the model influences the shape. The closer the crawler is to the VTS, the more the MatLab (red) and OrcaFlex bending stiffness model (yellow), diverge in shape (Fig 8.19). The 50-meter shape even wraps the s-shape the other way around. This shape forms because of the distributed applied force, starting at an arc length of 22,86 meters to 214,88 meters viewed from the left x-z axis. This force distribution occupies more than  $\frac{2}{3}$  of the total hose length, and the segment nodes now have a bending stiffness, so the segments can't rotate freely around the connecting nodes. The segments can't cross the node and hang in a gravity balance. Figure 8.20 shows another first step for the static analysis. This first step is the spline method, giving the hose an initial shape and then performing the static calculation. This approach gives an s-shape that is similar to the expected shape.

From this study, a few extra restrictions are added to the model, which is evaluated in the full static analysis presented in subsection 5.3.1. The parameter combinations influence the s-shape, especially the unsupported freely hanging part. The horizontal distance between the crawler and the VTS should not be at the end of the range.

### 8.2.4. The hose body

In the MatLab model, the hose is homogeneous. In the preliminary design, the structure of the hose is described, resulting in the following components of a section:

Hose components	Arclength	Unit
Coupling piece (ASME)	0,635	[m]
Neck reinforcement	1	[m]
Stiffener over hose	0,08	[m]
Hose (KI 1019-1001)	19,43	[m]
Stiffener over hose	0,08	[m]
Neck reinforcement	1	[m]
Coupling piece (ASME)	0,635	[m]

Table 8.10: Hose structure

These hose components all have a different mass, outer diameter, bending stiffness, axial stiffness etc. These values are presented in appendix B. The stiffeners are not described in the pre-laminar design, so in consultation with Royal IHC, it is assumed that these stiffeners are triangle connection pieces that connect the gap in OD variation. The weight of these stiffeners is:

$$A_{stiffener} = \frac{1}{2} * b_{stiff} * h_{OD\ variation} = \frac{1}{2} * 0,08 * 0,016 = 0,00064\ m^2$$

$$V_{stiffener} = A_{stiffener} * OD_{hose} = 0,00064 * 2 * \pi * \frac{0,3}{2} = 0,0098\ m^3 \quad (8.6)$$

$$Weight_{stiffener} = V_{stiffener} * material\ weight = 0,0098 * 1898 = 1,866\ kg\ per\ stiffener$$

Resulting in a total weight per section, table 8.11.

Hose component	Weight	Unit
Coupling	256,54	[kg]
Neck reinforcement	163	[kg]
Stiffener	10,96 +1,87	[kg]
Hose	2661,91	[kg]
Stiffener	10,96 +1,87	[kg]
Neck reinforcement	163	[kg]
Coupling	256,54	[kg]
<b>Total</b>	<b>3526,54</b>	<b>[kg]</b>

Table 8.11: Hose section, structure weight

When implemented in OrcaFlex:

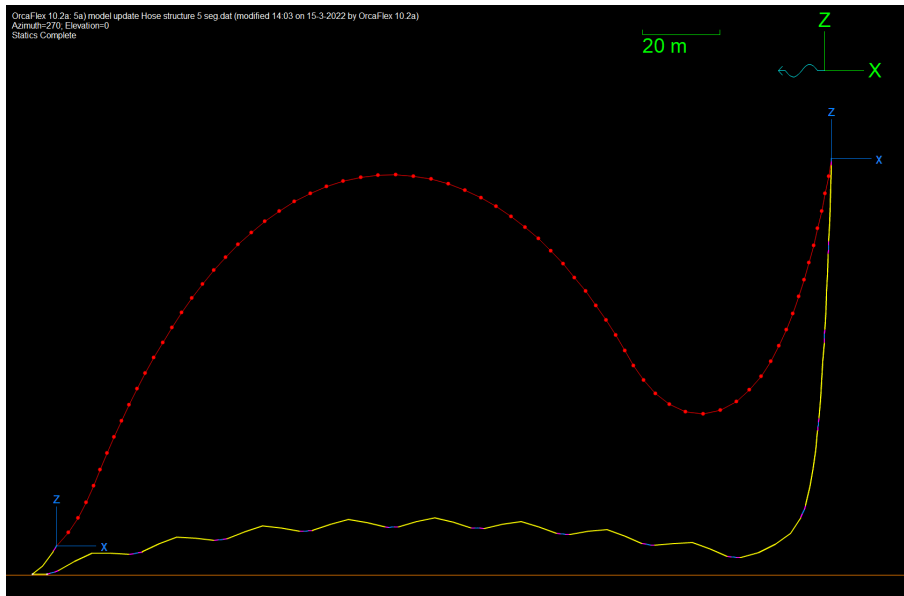


Figure 8.21: Hose sections composition vs MatLab model

Figure 8.21 shows the initial Matlab design vs the OrcaFlex, new hose structure. This new hose structure is heavier than in the Matlab design and resulted in the failure of the hose.

	MatLab design	OrcaFlex hose structure	Difference
Weight per section	3131,82 kg	3526,64 kg	394,82 kg
Weight per meter	154,27 kg/m	137 kg/m	17,27 kg/m

Table 8.12: Weight distribution jumper hose

Due to the failure of the design, a new examination of the parameter study is executed. The implementation and steps are executed in the same manner as described in chapter 7. Only the distributed weight of the hose material and design parameters are changed. The first design parameter that needed an update was the number of buoyancy modules. The range of this parameter is based on the total weight of the hose, resulting in the following calculation:

Total length hose	100 % of total weight hose	100 % of weight lifted by modules	66 % of total weight hose	66 % of weight lifted by modules
300 m	17345,66 kg	40,6	11448,14 kg	26,8
350 m	19823,61 kg	46,4	13083,58 kg	30,6
400 m	22301,56 kg	52,2	14719,03 kg	34,5
450 m	24779,51 kg	58,0	16354,48 kg	38,3

Table 8.13: Weight of the hose and buoyancy modules

The second design parameter is the number of modules per section. The maximum number of buoyancy modules and starting section influences the parameter, the number of modules per section. It's no longer feasible to have three modules per section because, for the combination, a total length of 300 m, three modules per hose and 41 modules in total results in that all hose sections have buoyancy modules. This number exceeds the boundary conditions of the design.

The new design parameters are:

Design parameter	Range	Unit	Stepsize
Total length hose	300 to 450	[meters]	50 meters
Number of clamped modules per section	4 to 8	[pieces per section]	1 module
Number of modules (300 m hose)	27 to 41	[modules]	1 module
Number of modules (350 m hose)	31 to 47	[modules]	1 module
Number of modules (400 m hose)	35 to 53	[modules]	1 module
Number of modules (450 m hose)	38 to 58	[modules]	1 module
Height VTS connection above seafloor	75 to 125	[meters]	25 meter
Range crawler	50 to 200	[meters]	50 meter
Mass of the slurry	1025 and 1200	[kg/m <sup>3</sup> ]	-

Table 8.14: Range design parameters for parameter study

After performing the parameter study, the parameters selected are shown in Table 8.15. The results are shown in Figure 8.22.

Design parameter	Value	Unit
Total length hose	300	[meters]
Buoyancy modules	32	[pieces]
Number of clamped modules per section	4	[pieces per section]
Height VTS connection above seafloor	100	[meters]
Range crawler	50 to 200	[meters]
Mass of the slurry	1025 and 1200	[kg/m <sup>3</sup> ]

Table 8.15: Best parameter combination after multicriteria analysis

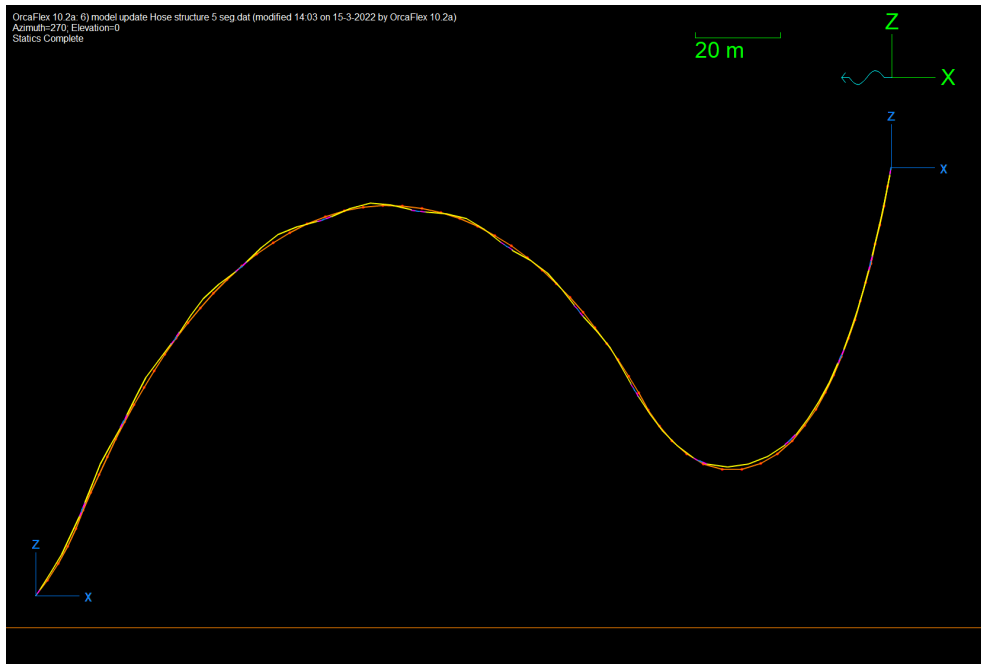


Figure 8.22: Shape of parameter set of table 8.15, applied forcing

## 8.3. Design Parameters, Static analysis

The static analysis model is built with the buoyancy modules extensions, bending stiffness and the hose structure. This results in the following OrcaFlex model,

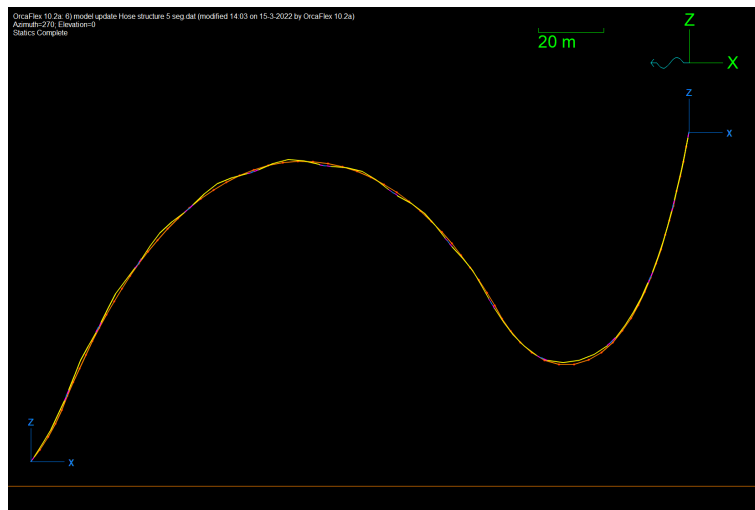


Figure 8.23: Overview OrcaFlex calculations, applied forcing

The segment length is not varied compared to the MatLab model and results in the buoyancy models being drawn at the segment nodes. The segment length is important for the force distribution, so the more segments, the closer the buoyancy force corresponds to the actual design location. The following segment lengths are considered:

- 4 segments per section
- 10 segments per section
- 19 segments per section

	Segment length	Location on line, arclength	Diff	Location on line, arclength	Diff
Buoyancy location		25,72 m		31,43 m	
4 segments	4,858 m	24,575 m	1,143 m	29,433 m	2,001 m
10 segments	1,943 m	26,518 m	0,800 m	32,347 m	0,914 m
19 segments	1,023 m	25,598 m	0,120 m	31,733 m	0,300 m
39 segments	0,499 m	25,571 m	0,147 m	31,550 m	0,117 m

Table 8.16: Difference in node locations

	Segment length	Location on line, arclength	Diff	Location on line, arclength	Diff
Buoyancy location		37,15 m		42,86 m	
4 segments	4,858 m	39,1475 m	1,999 m	44,005 m	1,142 m
10 segments	1,943 m	36,233 m	0,915 m	42,062 m	0,801 m
19 segments	1,023 m	36,847 m	0,301 m	42,982 m	0,119 m
39 segments	0,499 m	37,030 m	0,118 m	43,009 m	0,146 m

Table 8.17: Difference in node locations

When evaluating the number of segments against the distance between the designed buoyancy location and the nearest node, the calculation time is evaluated against the difference (difference results in table 8.16 and 8.17). When the segment length is cut in half, the difference goes down, but as shown in subsection 8.2.2, the calculation time increases. If the segment length is reduced from 1 meter to 0.5 meters, the difference between some buoyancy locations barely changes. 19 segments are chosen. The calculation time of all these segments is within 5 seconds, which is not a big difference.

### 8.3.1. Static Analysis

Static analysis examines the quantitative research data provided below. It gives data on the behaviour and equilibrium of the whole structure and the individual elements. It is used as a starting point of the dynamic design. The results are displayed over the full length of the hose in figure 8.26 and 8.25 and the main values in table 8.18. This table also compares the main values against the selected failure criteria.

OrcaFlex Design Parameter	Value	Unit	Failure Criteria	
			Value	Unit
Lift force Crawler	13,10	[kN]	60	[kN]
Tipping moment due to lift force	183,4	[kNm]	840	[kNm]
End moment connection point	0	[kNm]	120	[kNm]
Angle hose	53,9	[degree]	45	[degree]
Max tension Crawler to buoyancy	25,63	[kN]	60	[kN]
Height lowest point lazy s-shape	37,88	[m]	10	[m]
Maximum tension hose	45,53	[kN]	200	[kN]

Table 8.18: MatLab OrcaFlex model vs failure criteria

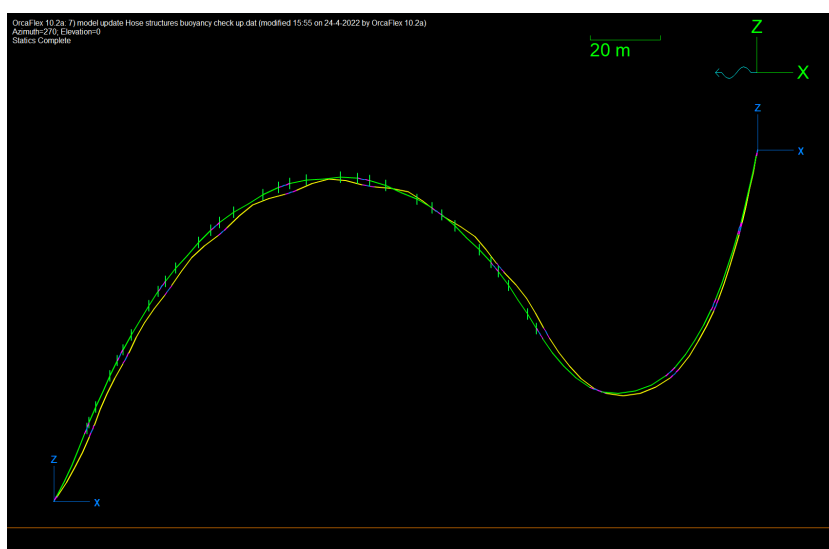


Figure 8.24: Static OrcaFlex model

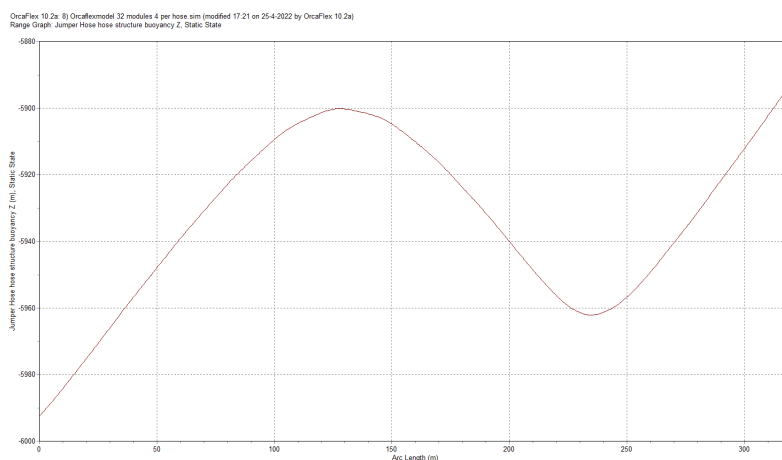


Figure 8.25: Static OrcaFlex model, z coordinate

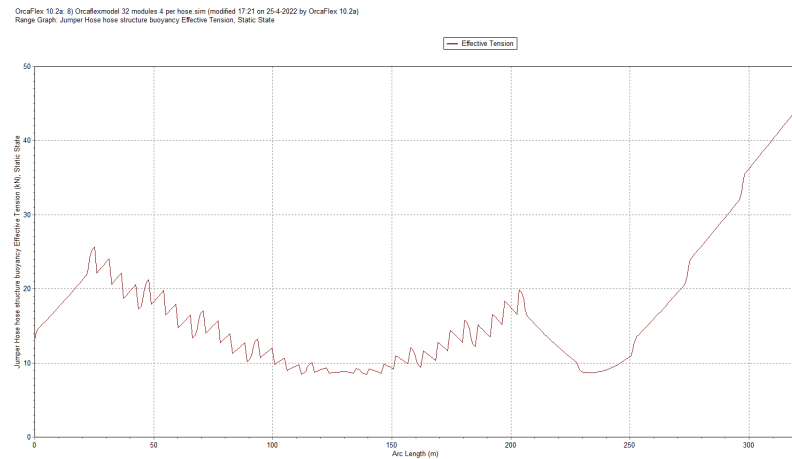


Figure 8.26: Static OrcaFlex model, effective tension

## 8.4. Research question 5 and summary of design choices

The answer to the research question, “What are the results of the static analysis using the dynamic software?” is presented in table 8.19 and visualised in figure 8.24. The improvement of the design parameters resulted in a new set of redesigned parameters. The implementation of the hose structure resulted in the greatest difference in design, which resulted in a renewed parameter study. The starting point for the dynamic analysis is the static OrcaFlex model and consists of the parameters shown in Table 8.19.

Parameter	Value	Unit
Hose length	320,088	[m]
VTS connection	100	[height above seafloor]
Total number of buoyancy modules	32	[modules]
Modules per hose section	4	[per section]
Starting after section	1st	[section]

Table 8.19: Design parameters for starting point dynamic analysis



# Dynamic analysis

The dynamic analysis is a sequel to subchapter 8.3, the static analysis. This chapter also reviews the implementation of the parameter set, which is derived from the static MatLab model in OrcaFlex. This review failed in the design and an update for the parameter set, followed by static analysis. The static analysis is used as a starting point for the dynamic analysis.

The first dynamic simulations are developed in this chapter and partially answer research question 6, *Which simulations are performed for the dynamic analysis?*. Chapter 11 gives enhanced models, which include movements and motions of the VTS connection and the crawler connection.

Subchapter 9.1 reviews the static base model and subchapter 9.2 reviews the requirements and criteria. This latter subchapter describes the system dynamics, explaining the motions and forces related to the system, which are included in this chapter's dynamic simulations. The criteria and the corresponding maximum and minimum values used to evaluate the models are elaborated and verified. These are summarized in subsection 9.2.5. Also, the terminology of the tension is interpreted in subsection 9.2.4.

The dynamic simulations start with a displacement in line with the model (2D, subchapter 9.3. This model is used to test the implementation of the displacement. It starts with a relatively small motion of 50 meters, and it then expands to the full range of the crawler parameter. Next, the perpendicular motion of the crawler is simulated (3D, subchapter 9.4). Combining the inline and perpendicular motion results in a range for the crawler relative to the VTS. This range also answers the last research question, *How manoeuvrable is the seafloor mining tool?*, subchapter 9.5.

## 9.1. Static model - base

From subsection 8.2.4 , table 8.15 the static analysis and parameter study resulted in the new design: subsection

Parameter	Value	Unit
Length jumper hose	320,04	[m]
Height VTS	100	[m]
Buoyancy modules	32	[-]
Modules per hose section	4	[-]
Starting after	1st	[hose section]

Table 9.1: Static design parameters

As shown in figure 1.1, the system consists of:

- Surface vessel
- Vertical transport system (VTS)
- Clump weight
- Jumper hose

- Crawler

The static analysis is used as the starting point for the dynamic analysis of two basic motions. First, the crawler is displayed in the x-direction and second, in the y-direction. These motions are analysed and evaluated against the failure criteria and requirements.

The focus of these motions is on the interaction between the jumper hose and the crawler. A swivel connects the jumper hose and crawler so the crawler can freely rotate around its vertical axis. A bend stiffener also reinforces this connection to prevent overbending at this point. The current demands on the design are displayed in table 9.2.

Parameter	Value	Unit
Bending moment at crawler	840	[kNm]
Lift force at crawler	200	[kN]
Update lift force due to weight crawler	60	[kN]
Height above seafloor lazy s	10	[m]
Angle between hose crawler	45	[degree]
Maximum tension hose	200	[kN]
Minimum bending radius	3	[m]

Table 9.2: Failure criteria jumper hose

## 9.2. Requirements and criteria

### 9.2.1. System dynamics

The system is subjected to several environmental dynamic effects and deep-sea mining operations. The system is described from the sea surface to the seafloor. The wind and wave's motions govern the motions of the surface vessel. These motions are transferred on the VTS and result in motions of the riser system and the clump weight. In this research, the main focus is the movement of the crawler and the interaction between the jumper hose and the crawler. The clump weight is modelled as a fixed point as a boundary condition. The motions of the clump weight are therefore disregarded for these simulations.

As discussed in subchapter 3.1.2, a current is present on the seafloor. This velocity is very low in the deep sea, and its effect on the jumper hose is therefore not considered.

The inertia of the jumper hose is simulated and influences the drag force when the crawler moves in a direction. These parameters significantly influence the shape of the hose and the forces on the crawler. The internal slurry dynamics are beyond this thesis's scope, so only a constant density inside the hose is used for the hose mass.

The design, dynamics and limitations of the crawler are adopted from the crawler design of the Blue Nodules project.

### 9.2.2. Crawler stability

In subsection 3.3.2 the requirements of the design are described to which the design fails. An important requirement is for the crawler to be kept upright during the operations and not tip over by the forces resulting from the movement of the jumper hose. The criteria are analysed by evaluating the bending moment and shear force exerted by the jumper hose on the crawler.

The maximum velocity of the crawler is set at 0,5 meters per second. 'In Blue Nodules report D2.4 "Initial Design of Vehicle Propulsion and Propulsion Test Performance", the drag forces of both ocean current and relative current due to the driving speed are taken into account for the propulsion design. A drag force of 13kN is calculated and, assuming it is equally divided over the clump weight and crawler, a drag force of 6.5kN remains for the crawler.' [17]

The crawler can topple sideways under the load of the jumper hose. The tipping moment in x-direction is calculated by taking the lever arm from the middle of the vehicle to the middle of the track. The crawler has a width of 16 meters, with 2 m wide tracks and an underwater weight of 200 kN, see subchapter 3.2.1.

$$\begin{aligned}
 M &= \text{weight} * \text{arm} \\
 M_x &= 200 * \frac{16 - 2}{2} = 1400 \text{ kNm} \\
 M_y &= 200 * \frac{16}{2} = 1600 \text{ kNm}
 \end{aligned}
 \tag{9.1}$$

This value is checked with OrcaFlex result: bend moment, x-bend moment, y-bend moment.

### 9.2.3. Jumper hose stability

In subsection 3.3.2 the requirements of the design are described to which the design fails. There are three main requirements for the jumper hose:

- The jumper hose is not allowed to touch or recline on the seafloor
- The jumper hose is not allowed to lift the crawler from the seafloor or make it tip over
- The jumper hose should never exceed the minimum bending radius
- The material of the jumper hose has a maximum effective tension force

The hose should never touch the seafloor and has a safety margin of 10 m above the seafloor. The lowest point in the S-shape is not a constant node number, so it is impossible to appoint a specific node in OrcaFlex. It is possible to plot to extract the z coordinate of all the nodes and then evaluate that with the set margin.

The criteria for the lift force on the crawler depend on the tension force located at the connection between the crawler and jumper hose and the top of the s-shape. These values are extracted with the Effective tension. The explanation of this term is described in subsection 9.2.4. Another option is to extract the shear force at the connection or end force.

The minimum bending radius is maintained in the design and implementation of the hose parameters of OrcaFlex. It is set such that the hose has a minimum bending radius which can not be exceeded. The minimum bending radius is also checked with the MBR results of OrcaFlex.

This benchmark result is four locations/segments/sections on the jumper hose that are important for analyses in the dynamic analysis. The first section is the connection point between the jumper hose and the crawler, with a maximum tipping moment, shear force and tension. Next is the node with the maximum force for the lift force in the first part of the s-shape and the maximum tension. The next section is the second bend of the s-shape to examine the lowest node. The last location is the connection point between the VTS and the JH. Here the maximum tension in the hose is reached. The minimum bending radius is considered over the full length of the hose.

### 9.2.4. Location maximum tension and effective tension

OrcaFlex reports two different types of tension, the effective tension and the wall tension. The tension in the axial spring-damper at the centre of the segment is the effective tension. This force vector points in the direction of the segment,  $s + n$  and is given by:

$$T_e = T_w + (p_o a_o - p_i a_i) \tag{9.2}$$

Where:

$T_w$  = wall tension

$p_i$  = internal pressure, calculated from the contents pressure.

$p_o$  = external pressure. Calculated by allowing the static pressure head due to the instantaneous height difference between the point and mean water level.

$a_i$  = internal cross section areas of stress annulus,  $= \frac{\pi}{4} ID_{stress}^2$

$a_o$  = external cross sectional areas of stress annulus  $= \frac{\pi}{4} OD_{stress}^2$

Figure 9.1 illustrates the tension and pressure forces acting at the mid-point of a segment.

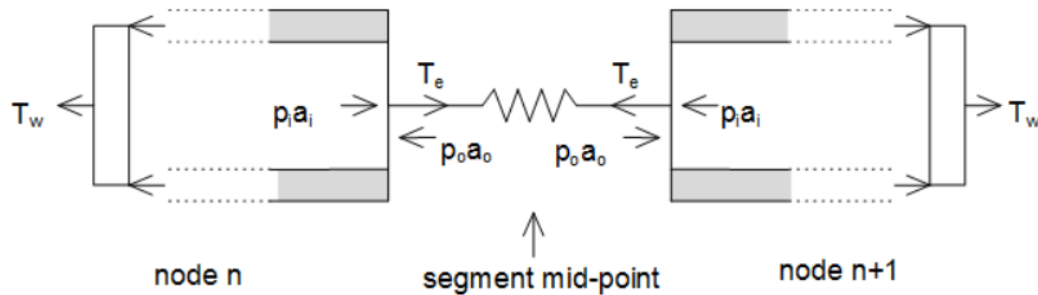


Figure 9.1: OrcaFlex, effective tension [22]

The effective tension is checked for the static model (table 9.3 and 9.4). Figure 9.2 shows the gradient of the effective tension.

Midsegment node	Arclength [m]	Internal pressure [kPa] $p_i$	Internal cross section [m <sup>2</sup> ] $a_i$	Force [kN] $p_i * a_i$	Internal pressure [kPa] $p_o$	External cross section [m <sup>2</sup> ] $a_o$	Force [kN] $p_i * a_i$
5 - 6	3,249	2345,702	0,0707	165,808	60208,948	0,176	10579,696
6 - 7	4,272	2335,757	0,0707	165,105	60200,453	0,176	10578,204
7 - 8	5,294	2325,721	0,0707	164,396	60191,881	0,176	10576,697
8 - 9	6,317	2315,597	0,0707	163,680	60183,233	0,176	10575,178
9 - 10	7,339	2305,387	0,0707	162,958	60174,512	0,176	10573,645
10 - 11	8,362	2295,095	0,0707	162,231	60165,721	0,176	10572,101

Table 9.3: Effective tension check

Midsegment node	Wall Tension [kN] $T_w$	Internal Force [kN] $p_i * a_i$	External Force [kN] $p_i * a_i$	Effective Tension [kN] $T_e$	Orcaflex Tension [kN] $T_e$
5 - 6	-10398,676	165,808	10579,696	15,213	15,213
6 - 7	-10397,541	165,105	10578,204	15,558	15,558
7 - 8	-10396,395	164,396	10576,697	15,906	15,906
8 - 9	-10395,240	163,680	10575,178	16,258	16,258
9 - 10	-10394,074	162,958	10573,645	16,613	16,613
10 -11	-10392,899	162,231	10572,101	16,971	16,971

Table 9.4: Effective tension check

OrcaFlex 10.2a 8) Orcaflexmodel 32 module 4 per hose sim (modified 17:21 on 25-4-2022 by OrcaFlex 10.2a)  
Range Graph: Jumper Hose hose structure buoyancy Effective Tension, Static State

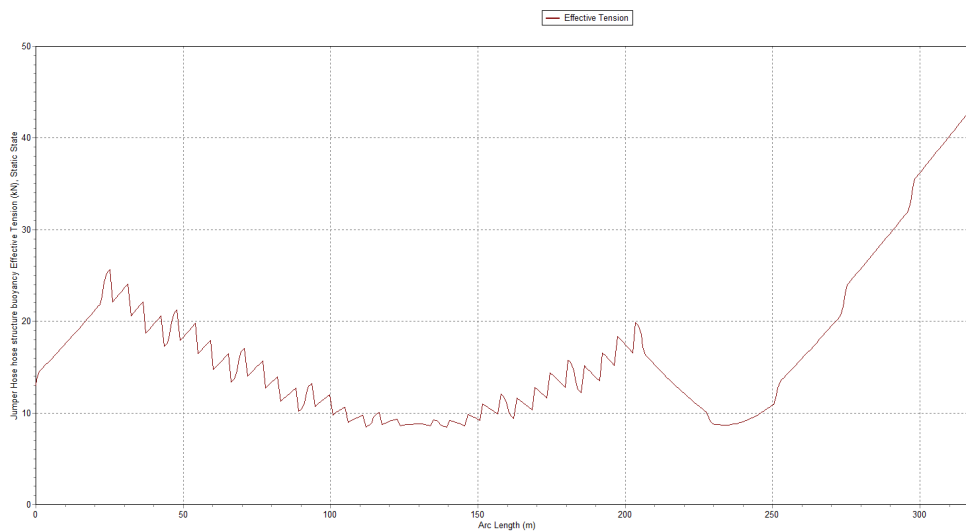


Figure 9.2: Effective tension over the acrlgnth, parameters table 9.1 [22]

The effective tension is the tension as described in subchapter 4.3 and the term effective tension is used from here on.

Figure 9.2 shows the effective tension, which has the same shape as the effective tension of the static analysis. Important tension locations are before the start segment of the buoyancy location, the segment after the end of the buoyancy locations, and the jumper hose’s connection point. The sawtooth pattern arises from the segments with buoyancy modules. These reduce the tension.

### 9.2.5. Summary specified requirements and limitations

The specified requirements and limitations of the dynamic design are summarised in Table 9.5.

Location	Requirement	Value	Unit
at connection crawler to jumper hose	Bend moment/Shear force	960	[kNm]
First section hose	Effective tension below	60	[kN]
	Angle	45	[degree]
at node maximum tension first part of the s shape	Effective tension below	60	[kN]
at the lowest point of the second part of the s shape	Z-location	-5990	[m]
at node maximum tension	Effective tension below	100	[kN]

Table 9.5: Limitations dynamic design

## 9.3. OrcaFlex simulations - In-line motion

This subchapter tests and updates the implementation of motions into OrcaFlex software. Starting at 50 meters and scaling up to the full range of the crawler. It also evaluates the set locations of subchapter 9.2 and the corresponding maximum indicators for certain criteria.

### 9.3.1. 50 meters in x direction

In simulation (Appendix E.5), the VTS node is fixed in place, and the crawler is modelled by a constraint, of which the motion is prescribed. The velocity of the crawler is set at 0,5 meters per second; 50 meters takes 100 seconds. Resulting in the following prescribed motion for the constraint table 9.6 and begin and end position on the motion illustrated in figure 9.3, 9.4 and 9.5.

Time [s]	X-coordinate [m]	Y-coordinate [m]	Z-coordinate [m]
-8	200	0	-5992,5
0	200	0	-5992,5
20	190	0	-5992,5
40	180	0	-5992,5
60	170	0	-5992,5
80	160	0	-5992,5
100	150	0	-5992,5
120	150	0	-5992,5
⋮	⋮	⋮	⋮
200	150	0	-5992,5

Table 9.6: Movement of the crawler node

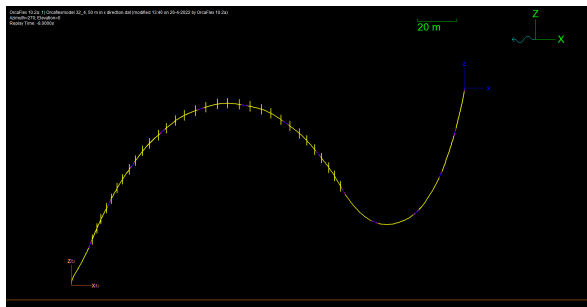


Figure 9.3: OrcaFlex model at 0 seconds

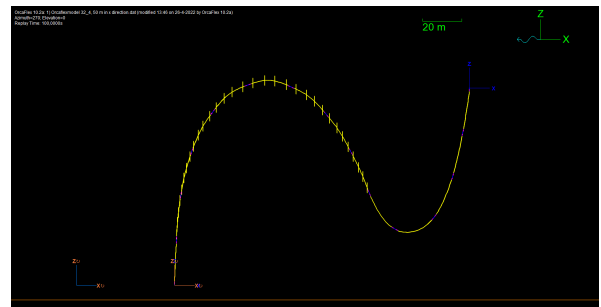


Figure 9.4: OrcaFlex model at 100 seconds

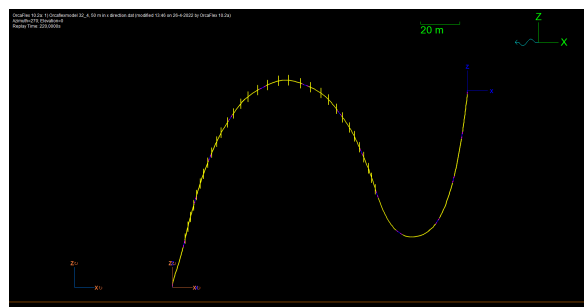


Figure 9.5: OrcaFlex model at 200 seconds

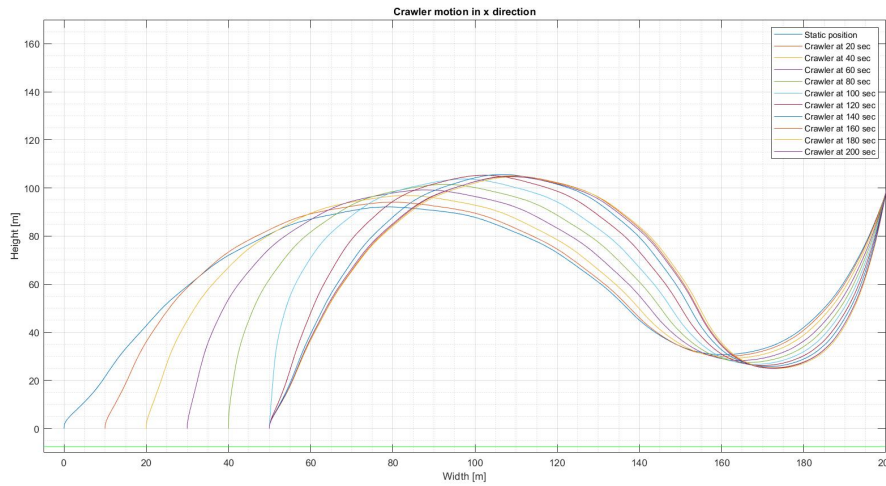


Figure 9.6: Timelapse OrcaFlex results

OrcaFlex Design Parameter	Value	Unit	Failure Criteria	
			Value	Unit
Lift force Crawler	16,208	[kN]	60	[kN]
Tipping moment due to lift force	226,9	[kNm]	840	[kNm]
End moment connection point	30,36	[kNm]	120	[kNm]
Angle 1st hose section	64,6	[degree]	45	[degree]
Max tension Crawler to buoyancy	28,536	[kN]	60	[kN]
Height lowest point lazy s-shape	-5967,59	[m]	-5990	[m]
Maximum tension hose	44,78	[kN]	200	[kN]

Table 9.7: Results OrcaFlex model 50 meters in x direction

The results of the connection between the jumper hose and the crawler are plotted in figures 9.7 to 9.12 for the period described in table 9.6. The maximum lift force on the connection point is extracted from these graphs. The effective tension and z-coordinate are evaluated for the hose length, and then the maximum, minimum, mean and allowable tension is plotted for the whole period.

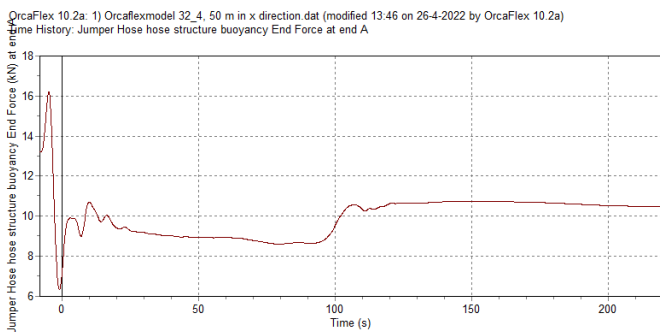


Figure 9.7: Results OrcaFlex model, position node A, End Force

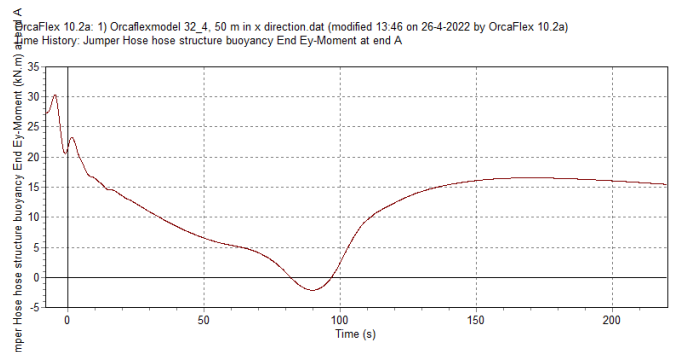


Figure 9.8: Results OrcaFlex model, position node A, Moment in y

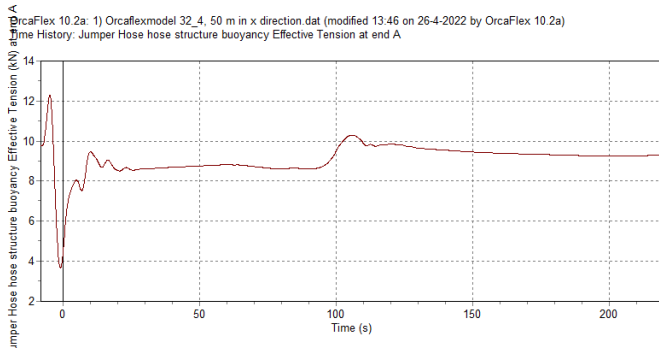


Figure 9.9: Results OrcaFlex model, position node A, effective tension

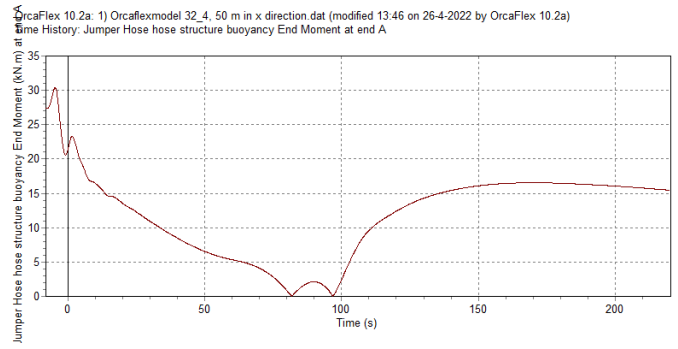


Figure 9.10: Results OrcaFlex model, position node A, end moment

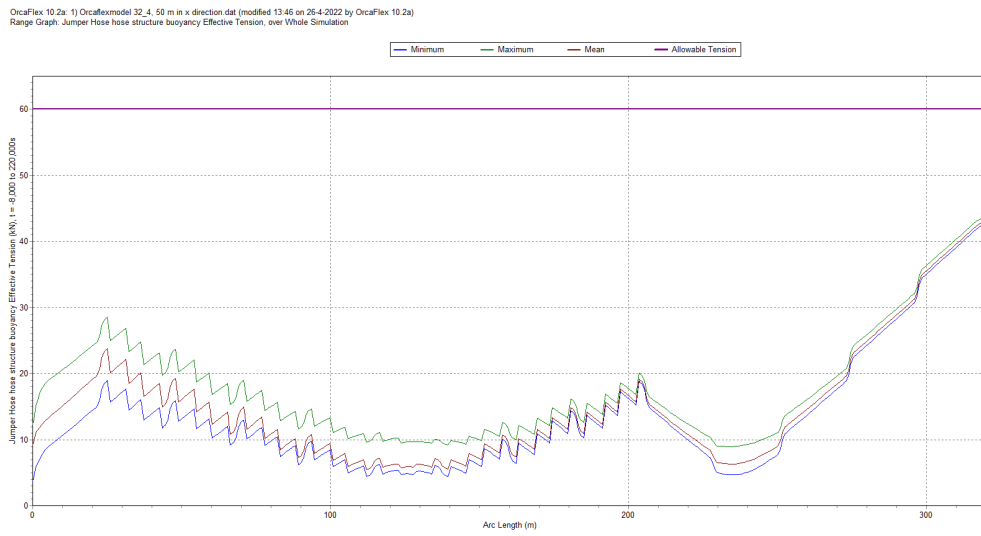


Figure 9.11: Results OrcaFlex model, position node A, end moment

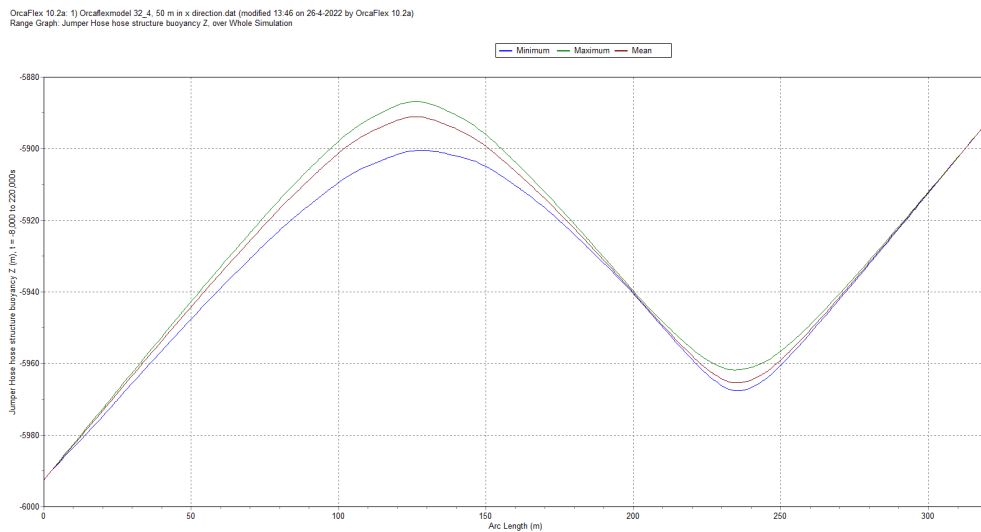


Figure 9.12: Results OrcaFlex model, position node A, end moment



The effective tension has the same shape as described in subsection 9.2.4 and the effective tension is also plotted for the connection point. It shows a fluctuation in the start-up period of the hose (-8 to 10 seconds). This fluctuation is because of the displacement induced on the end node of the jumper hose. The tension settles when the whole hose is in motion and only varies when the crawler is slowed down.

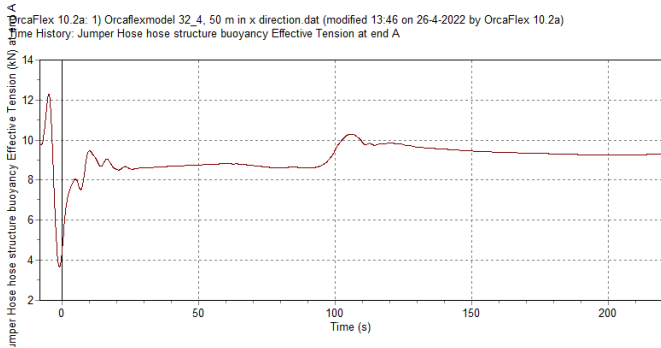


Figure 9.13: Results OrcaFlex model, position node A, effective tension

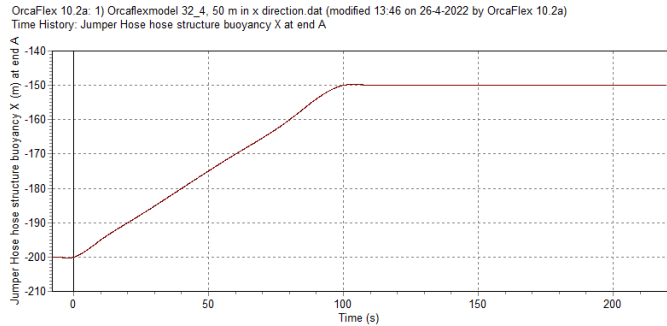


Figure 9.14: Results OrcaFlex model, position node A, x displacement

The end force and end moments are the total force and moment acting between the line end and the connected object. It exists as a summation of all the components present in the model. There are two types of moments considered in the model. The first is the tipping moment caused by the lifting force on the crawler, and the second is the bending moment induced by the line on the crawler.

### 9.3.2. 100 meters in x direction

In simulation (Appendix E.5), The VTS node is fixed in place, and the crawler is modelled by a constraint, of which the motion is prescribed. The velocity of the crawler is set at 0,5 meters per second; 100 meters takes 200 seconds. Resulting in the following prescribed motion for the constraint table 9.8 and begin and end position on the motion illustrated in figure 9.15, 9.16 9.17.

Time [s]	X-coordinate [m]	Y-coordinate [m]	Z-coordinate [m]
-8	0	0	-5992,5
0	0	0	-5992,5
20	10	0	-5992,5
40	20	0	-5992,5
80	40	0	-5992,5
120	60	0	-5992,5
160	80	0	-5992,5
200	100	0	-5992,5
220	100	0	-5992,5
⋮	⋮	⋮	⋮
320	100	0	-5992,5

Table 9.8: Movement of the crawler node

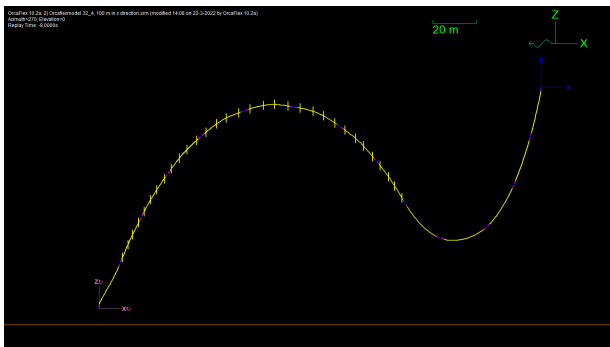


Figure 9.15: OrcaFlex model at 0 seconds

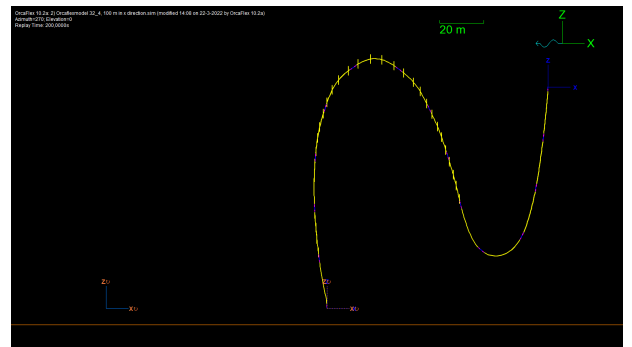


Figure 9.16: OrcaFlex model at 200 seconds

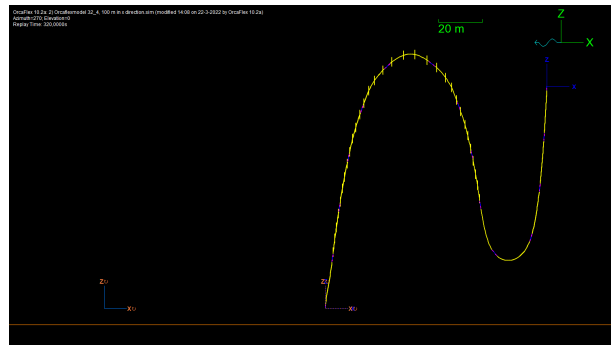


Figure 9.17: OrcaFlex model at 320 seconds

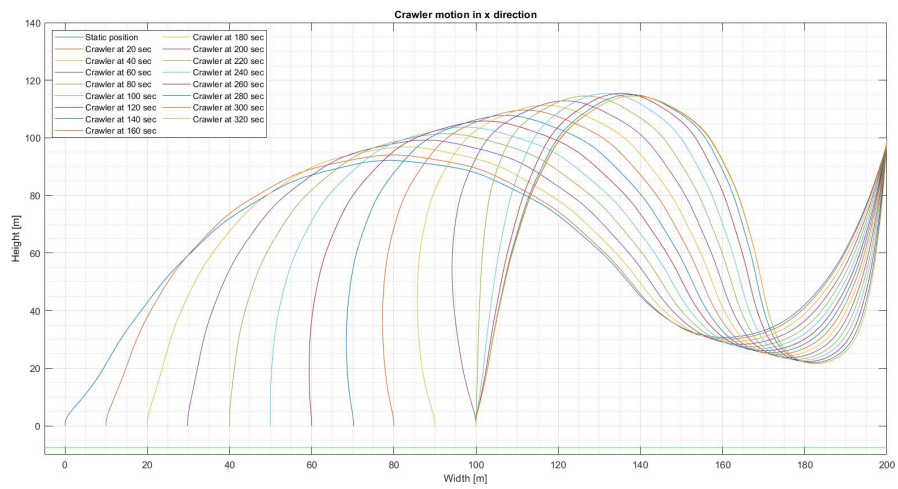


Figure 9.18: Timelapse OrcaFlex results

OrcaFlex Design Parameter	Value	Unit	Failure Criteria	
			Value	Unit
Lift force Crawler	15,16	[kN]	60	[kN]
Tipping moment due to lift force	212,24	[kNm]	840	[kNm]
End moment connection point	29,33	[kNm]	120	[kNm]
Angle 1st hose section	64,4	[degree]	45	[degree]
Max tension Crawler to buoyancy	27,35	[kN]	60	[kN]
Height lowest point lazy s-shape	-5970,79	[m]	-5982,5	[m]
Maximum tension hose	44,78	[kN]	200	[kN]

Table 9.9: Results OrcaFlex model 100 meters in x direction

The results of the connection between the jumper hose and the crawler are plotted in figures 9.19 to 9.24 for the period described in table 9.8. The same values as in subsection 9.3.1 are extracted from the model. The end moment is the absolute value of all the components that induce moment on the crawler.

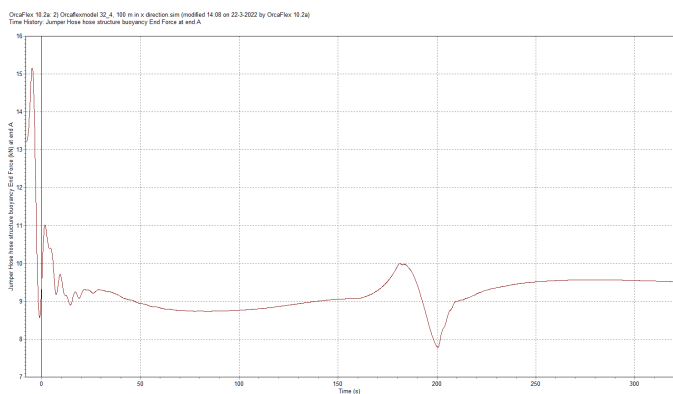


Figure 9.19: Results OrcaFlex model, position node A, End Force

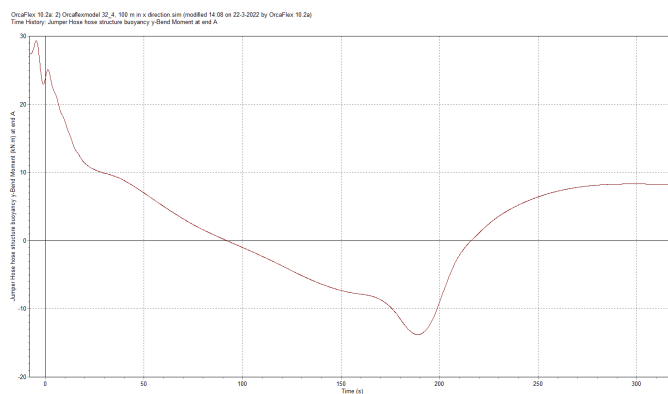


Figure 9.20: Results OrcaFlex model, position node A, Moment in y

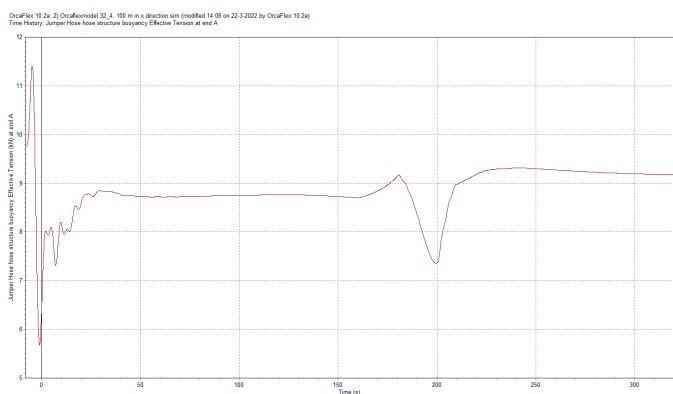


Figure 9.21: Results OrcaFlex model, position node A, effective tension

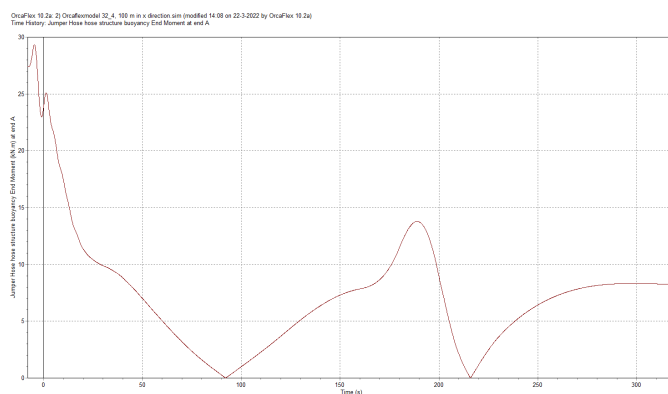


Figure 9.22: Results OrcaFlex model, position node A, end moment

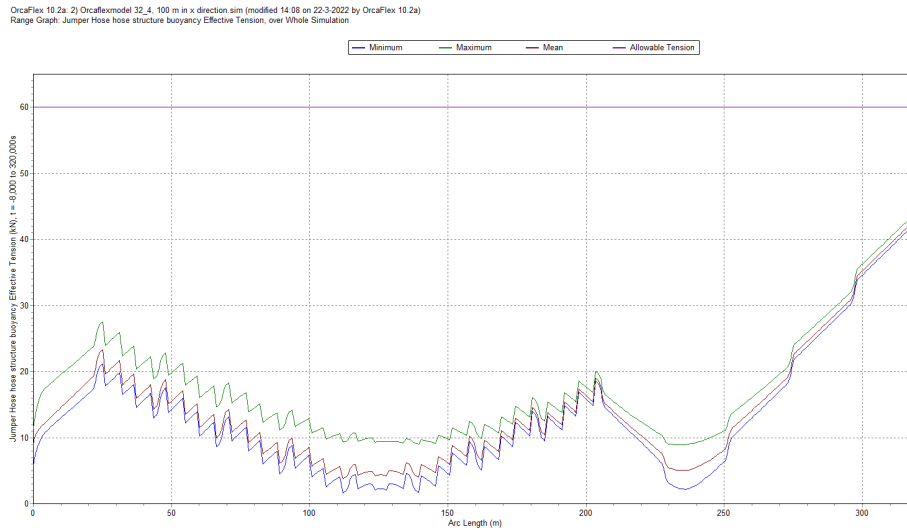


Figure 9.23: Results OrcaFlex model, Effective tension over arclength

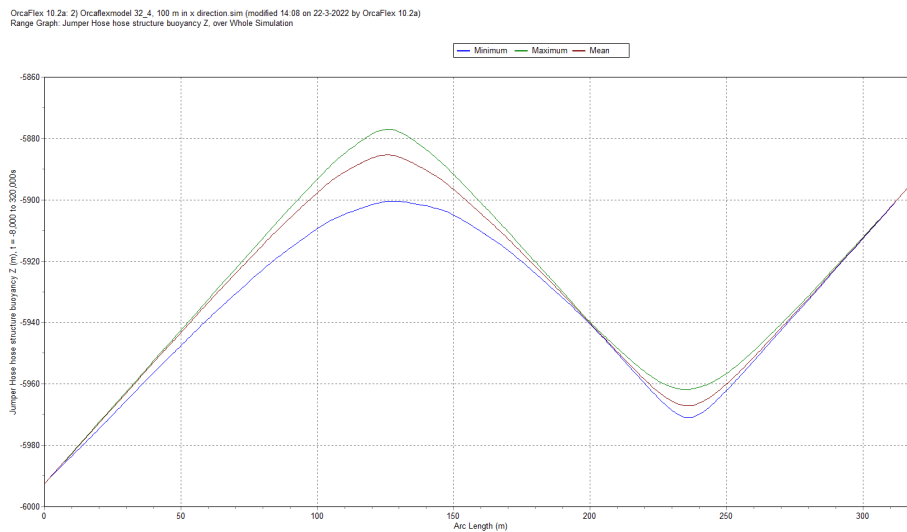


Figure 9.24: Results OrcaFlex model, z-position over arclength

### 9.3.3. 150 meters in x direction

In simulation (Appendix E.5), The VTS node is fixed in place, and the crawler is modelled by a constraint, of which the motion is prescribed. The velocity of the crawler is set at 0,5 meters per second; 150 meters takes 300 seconds. Resulting in the following prescribed motion for the constraint table 9.10 and begin and end position on the motion illustrated in figure 9.25, 9.4 and 9.26.

Time [s]	X-coordinate [m]	Y-coordinate [m]	Z-coordinate [m]
-8	0	0	0
0	0	0	0
20	10	0	0
40	20	0	0
80	40	0	0
120	60	0	0
160	80	0	0
200	100	0	0
240	120	0	0
280	140	0	0
300	150	0	0
320	150	0	0
⋮	⋮	⋮	⋮
420	150	0	0

Table 9.10: Movement of the crawler node

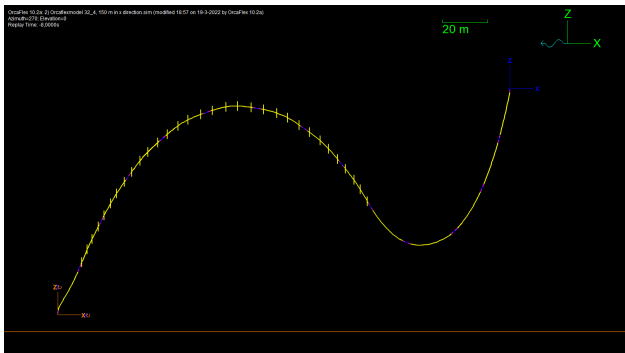


Figure 9.25: OrcaFlex model at 0 seconds

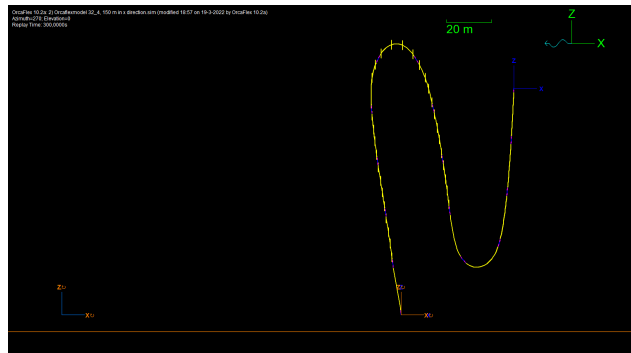


Figure 9.26: OrcaFlex model at 300 seconds

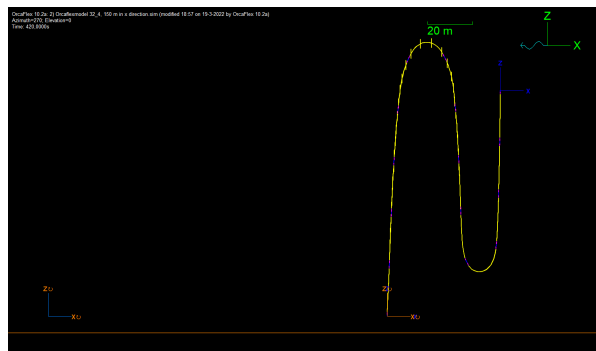


Figure 9.27: OrcaFlex model at 420 seconds

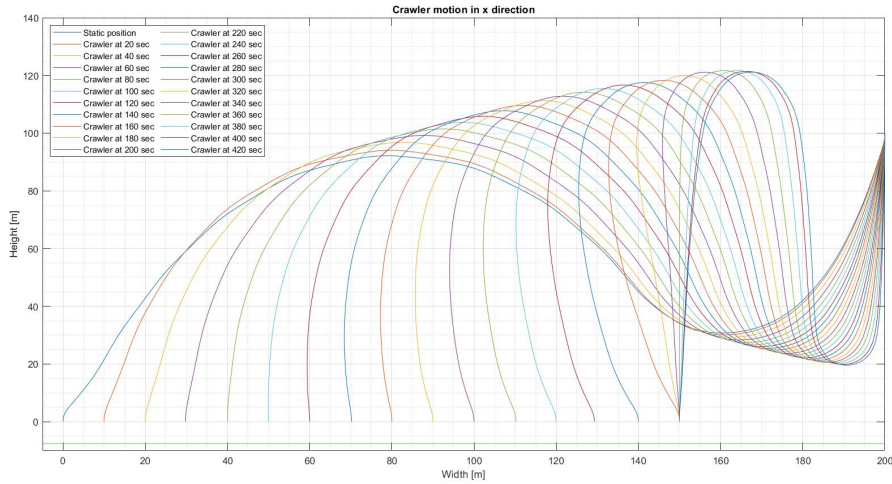


Figure 9.28: Timelapse OrcaFlex results

OrcaFlex Design Parameter			Failure Criteria	
Parameter	Value	Unit	Value	Unit
Lift force Crawler	15,16	[kN]	60	[kN]
Tipping moment due to lift force	212,24	[kNm]	840	[kNm]
End moment connection point	29,34	[kNm]	120	[kNm]
Angle 1st hose section	64,4	[degree]	45	[degree]
Max tension Crawler to buoyancy	27,53	[kN]	60	[kN]
Height lowest point lazy s-shape	-5972,80	[m]	-5982,5	[m]
Maximum tension hose	44,77	[kN]	200	[kN]

Table 9.11: Results OrcaFlex model 150 meters in x direction

The results of the connection between the jumper hose and the crawler are plotted in figures 9.29 to 9.34 for the period described in table 9.10. The same values as in subsection 9.3.1 are extracted from the model. The end moment is the absolute value of all the components that induce moment on the crawler.

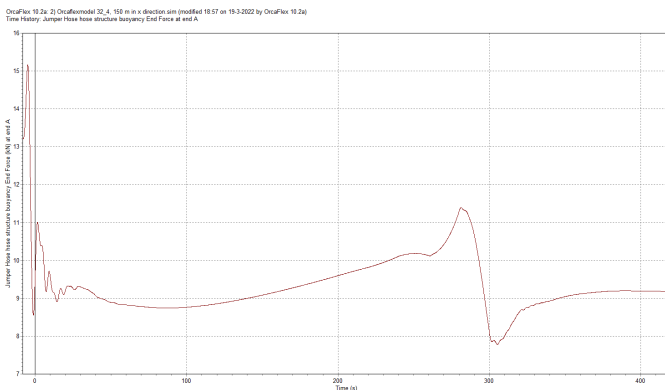


Figure 9.29: Results OrcaFlex model, position node A, End Force

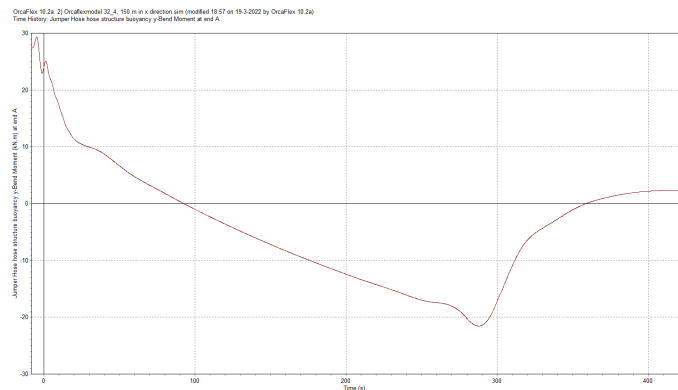


Figure 9.30: Results OrcaFlex model, position node A, Moment in y

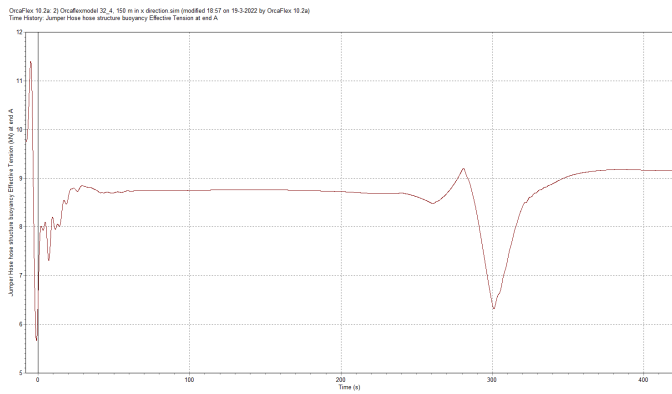


Figure 9.31: Results OrcaFlex model, position node A, effective tension

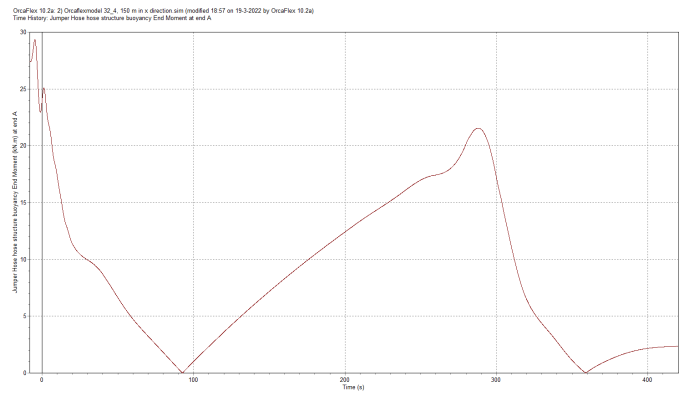


Figure 9.32: Results OrcaFlex model, position node A, end moment

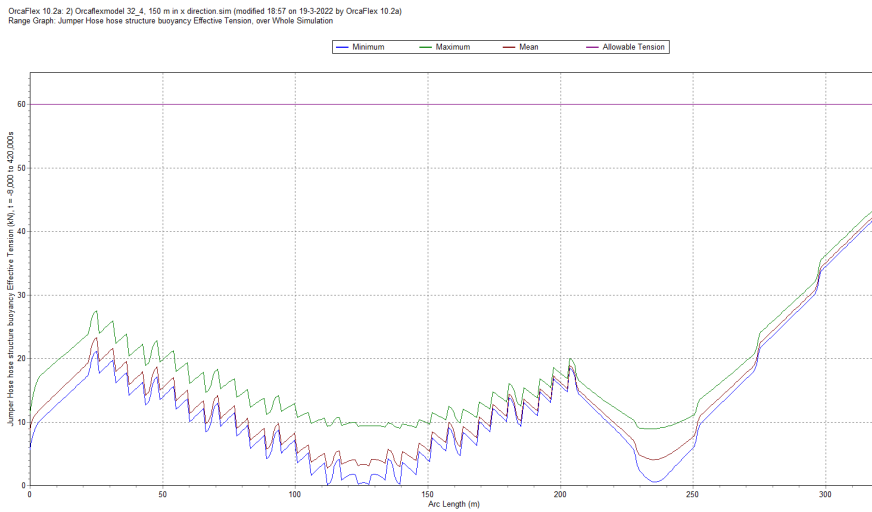


Figure 9.33: Results OrcaFlex model, Effective tension over arclength

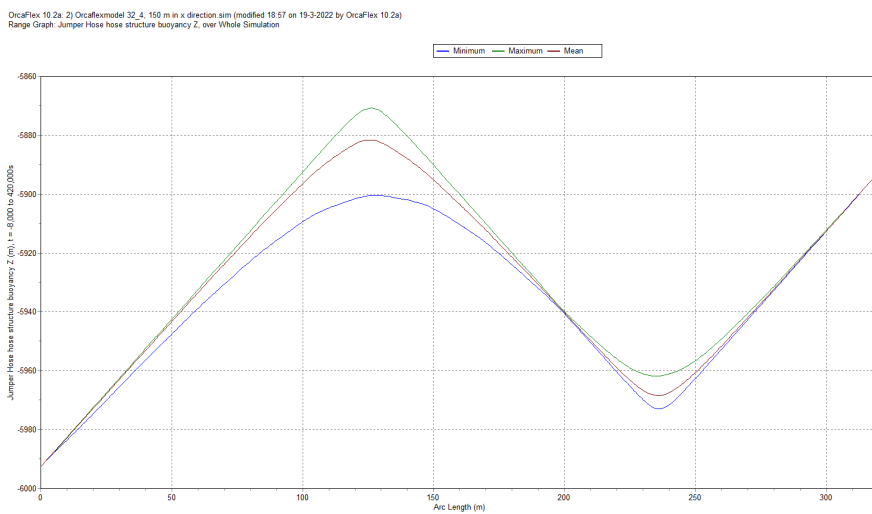


Figure 9.34: Results OrcaFlex model, z-position over arclength

### 9.3.4. Analysis conclusion

The critical evaluated values for the movement in the x-direction don't vary compared to each other. The only value that changes significantly is the height above the seafloor, which gets smaller if the displacement increases. Another observation was made during this analysis. The smoothness of the acceleration of the displacement influences all forcing parameters. These parameters decrease if the acceleration and deceleration of the system are smoother than the displacement shown in figures 9.35 to 9.38. Here the start-up of the system is left out. The results for the 50-meter case are developed in the following figures, 9.35 to 9.39.

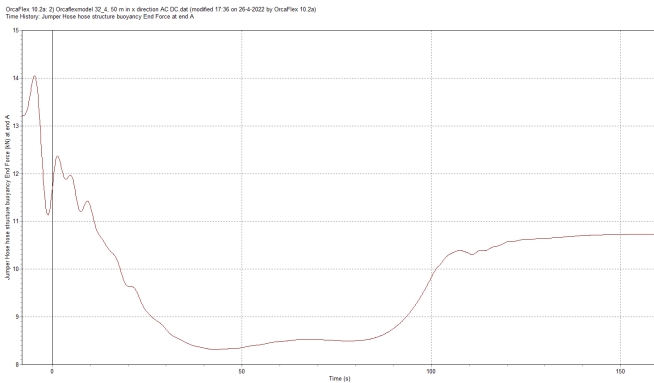


Figure 9.35: Results OrcaFlex model, position node A, End Force

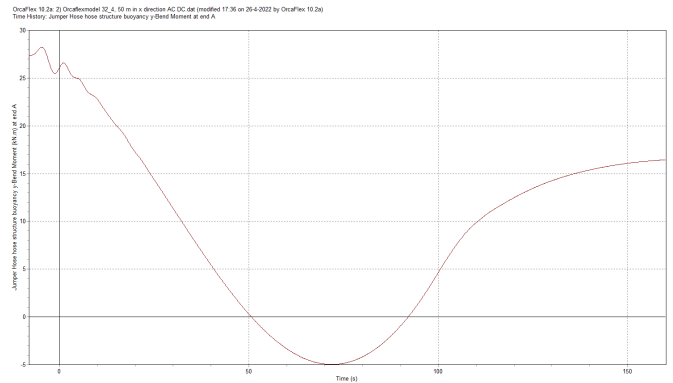


Figure 9.36: Results OrcaFlex model, position node A, Moment in y

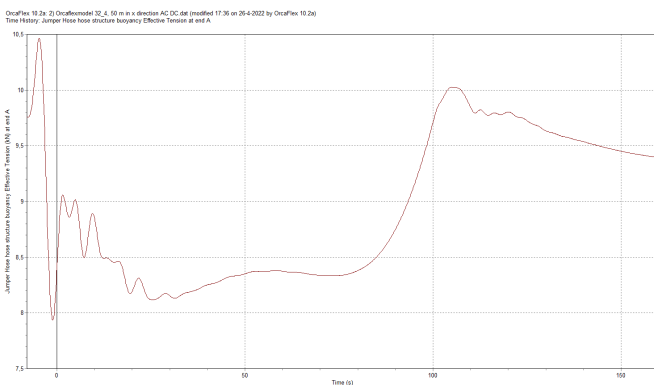


Figure 9.37: Results OrcaFlex model, position node A, effective tension

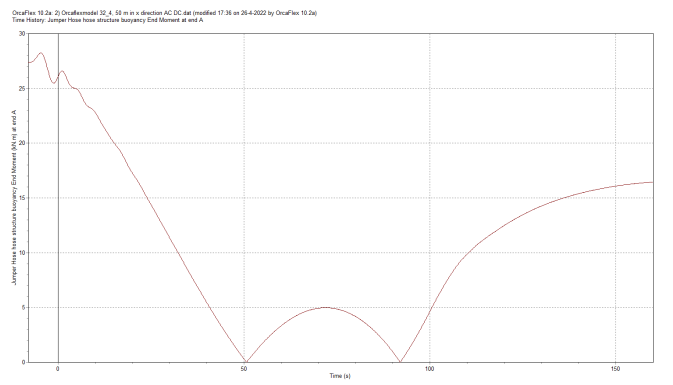


Figure 9.38: Results OrcaFlex model, position node A, end moment

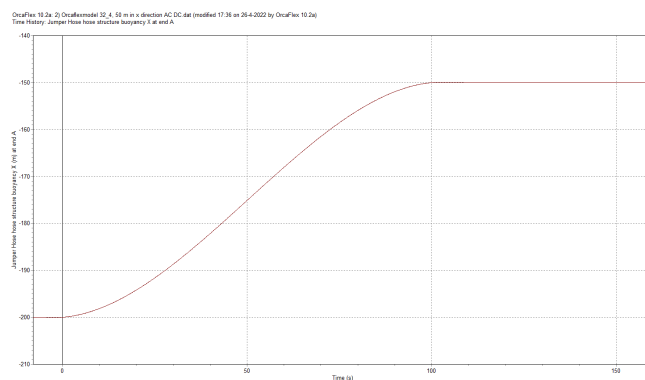


Figure 9.39: Results OrcaFlex model, position node A, x displacement



## 9.4. OrcaFlex simulations – Perpendicular motion

The second basic motion is the displacement of the crawler in y direction. The choice has been made to implement the y displacement on three different x distances. This is done because the distance between the crawler and the VTS has a major influence on the range of y displacement, due to the length of the hose. The range is determined in the following manner, where :

Where ,

$X_{distance}$  = the distance between the crawler and the VTS

$Y_{distance}$  = the range of the crawler motion

$L_{hose}$  = the length of the jumper hose

$S$  = distance between crawler and VTS

$H_{VTS}$  = distance between VTS and crawler

$$\begin{aligned} X_{distance}^2 + H_{VTS}^2 &= S^2 \\ (50^2 + 100^2) &= S^2 \\ 111,80 &= S \end{aligned} \tag{9.3}$$

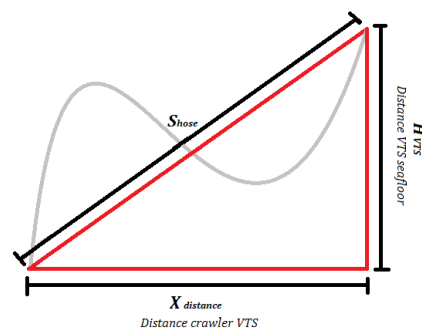


Figure 9.40: Schematic side view range

$$\begin{aligned} X_{distance}^2 + Y_{distance}^2 &= S_{flat}^2 \\ Y_{range\ flat}^2 &= S_{flat}^2 - X_{displacement}^2 \\ 316,1 &= Y_{range\ flat} \end{aligned} \tag{9.4}$$

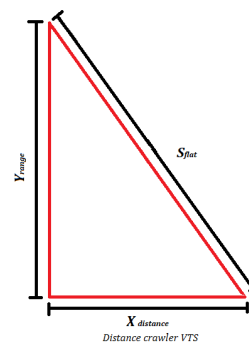


Figure 9.41: Schematic side view range

$$\begin{aligned} S_2^2 + H_{VTS}^2 &= S_{flat}^2 \\ S_2^2 &= S_{flat}^2 - H_{VTS}^2 \\ S_2^2 &= 304,2 \end{aligned} \tag{9.5}$$

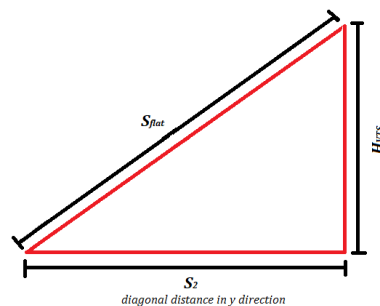


Figure 9.42: Schematic side view range

$$\begin{aligned}
 & \text{Updated flat} \\
 X_{distance}^2 + Y_{range}^2 &= S_2^2 \\
 Y_{range}^2 &= S_2^2 - X_{distance}^2 \\
 Y_{range\ updated} &= 287,1
 \end{aligned}
 \tag{9.6}$$

$$\begin{aligned}
 & \text{Ratio length hose to } S_{hose} \\
 & 320,04 : 11,80
 \end{aligned}
 \tag{9.7}$$

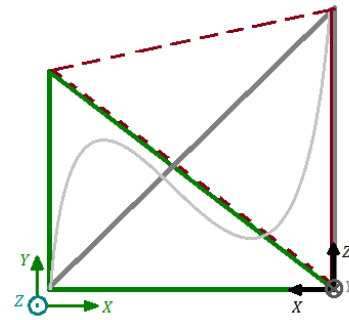


Figure 9.43: Schematic overview range

Distance between crawler and VTS	50 meters	100 meters	150 meters
H VTS	100 meters	100 meters	100 meters
$S_{hose}$	111,80	141,42	180,28
$L_{hose}$	320,04	320,04	320,04
$Y_{range\ flat}$	316,11	304,02	282,71
$S_2$	304,012	304,02	304,02
$Y_{range\ up}$	299,88	287,10	264,43
RATIO	2,86	2,26	1,78
New range	85,8 m	65 m	47 m
Test $\frac{1}{2}$	150 m	144 m	132 m
Test $\frac{1}{4}$	225 m	215 m	198 m

Table 9.12: Range calculations for evaluated distances

The 200-meter option is disregarded because its range in y direction is relatively small. The range Y 1 from table 9.12 is used as a starting point. The limit range is investigated per model.

#### 9.4.1. Y displacement at 50 meters

In simulation (Appendix E.6), The VTS node is fixed in place, and the crawler is modelled by a constraint, of which the motion is prescribed. The velocity of the crawler is set at 0,5 meters per second. The motion is implemented in the following manner. The approach is different from the implementation of the X-direction. OrcaFlex can calculate the acceleration and the deceleration of the motion. This approach results in less shaky plots. The rapidly varying amplitudes of the figures presented in subchapter 9.3 arise from the rapid varying accelerations or decelerations.

Time [s]	X-coordinate [m]	Y-coordinate [m]	Z-coordinate [m]
-8	50	-86	-5992,5
0	50	0	-5992,5
344	50	86	-5992,5
375	50	86	-5992,5
400	50	86	-5992,5
425	50	86	-5992,5
450	50	86	-5992,5
475	50	86	-5992,5
500	50	86	-5992,5

Table 9.13: Movement of the crawler node

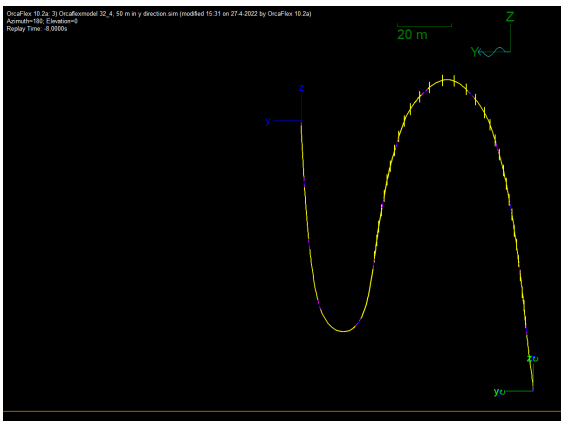


Figure 9.44: OrcaFlex model at 0 seconds

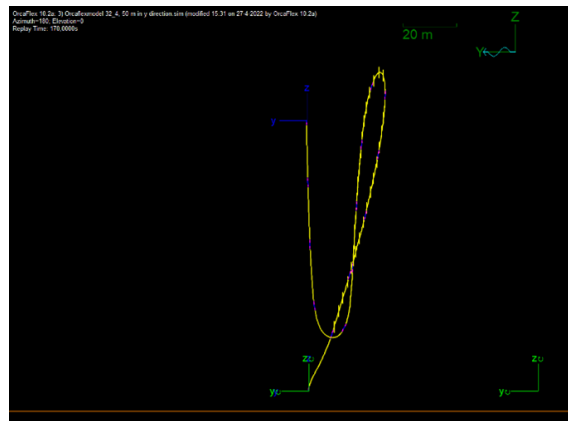


Figure 9.45: OrcaFlex model at 170 seconds

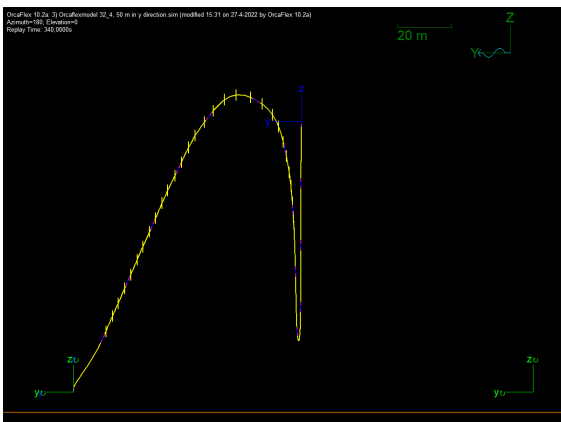


Figure 9.46: OrcaFlex model at 340 seconds

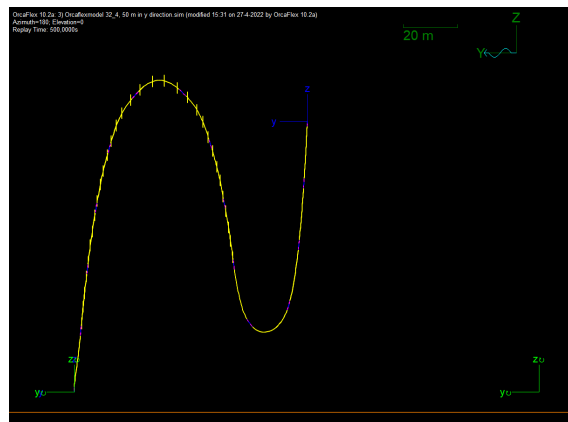


Figure 9.47: OrcaFlex model at 500 seconds

The overview figures illustrate that the crawler drags the hose through the water and the s shape follows with a delay. When the crawler stops, the hose needs about 160 seconds to form its final shape, and this shape is equivalent to the static equilibrium.

The model is evaluated on the same criteria and locations as the x displacement models, subsection 9.2.5. The outcome is presented in table 9.14.

OrcaFlex Design Parameter	Value	Unit	Failure Criteria	
			Value	Unit
Lift force Crawler	15,86	[kN]	60	[kN]
Tipping moment due to lift force	221,9,24	[kNm]	840	[kNm]
End moment connection point	41,29	[kNm]	120	[kNm]
Max tension Crawler to buoyancy	26,54	[kN]	60	[kN]
Height lowest point lazy s-shape	-5974,82	[m]	-5982,5	[m]
Maximum tension hose	45,18	[kN]	200	[kN]

Table 9.14: Results OrcaFlex model 50 meters in x direction, 50 m in y-direction

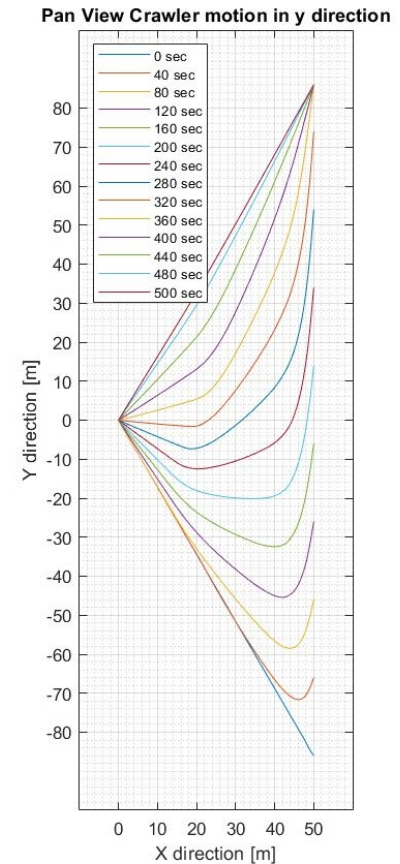


Figure 9.48: Time laps, x 50 m range 86 m, pan view

9.4.1.1. Y displacement at 50 meters, limit range

The displacement vector of simulation (Appendix E.6) is expanded to the 150-meter range and the 225-meter range. The range value is increased further. It is concluded that these ranges are feasible and that the forcing values do not reach the critical values.

OrcaFlex Design, 150 range Parameter	Range, 150		Range, 225		Failure Criteria	
	Value	Unit	Value	Unit	Value	Unit
Lift force Crawler	16,92	[kN]	18,25	[kN]	60	[kN]
Tipping moment due to lift force	236,88	[kNm]	255,5	[kNm]	840	[kNm]
End moment connection point	46,20	[kNm]	49,71	[kNm]	120	[kNm]
Max tension Crawler to buoyancy	26,78	[kN]	27,96	[kN]	60	[kN]
Height lowest point lazy s-shape	-5975,1	[m]	-5975,21	[m]	-5990	[m]
Maximum tension hose	45,37	[kN]	45,46	[kN]	200	[kN]

Table 9.15: Range test, critical values, model 9.48

The range is elevated with steps of 25 meters. Table 9.16 shows that the values increase slowly until the last step, from 275 meters to 300 meters. The 300-meter range model fails. All the evaluated values exceed the failure criteria. The high results are because the hose length is equal to the distance between the connection points of the crawler and the VTS. The range is scaled back with steps of 5 meters per step and the first step that stayed beneath the failure criteria is 285 meters, the values are shown in table 9.17.

OrcaFlex Design Parameter	250 range		Range, 275		Range, 300		Failure Criteria	
	Value	Unit	Value	Unit	Value	Unit	Value	Unit
Lift force Crawler	22,74	[kN]	33,60	[kN]	189,55	[kN]	60	[kN]
Tipping moment due to lift force	314,58	[kNm]	470,4	[kNm]	2653,7	[kNm]	840	[kNm]
End moment connection point	51,22	[kNm]	75,21	[kNm]	252,26	[kNm]	120	[kNm]
Max tension Crawler to buoyancy	31,69	[kN]	41,50	[kN]	192,81	[kN]	60	[kN]
Height lowest point lazy s-shape	-5975,24	[m]	-5975,25	[m]	-5974,75	[m]	-5990	[m]
Maximum tension hose	46,29	[kN]	55,25	[kN]	197,01	[kN]	200	[kN]

Table 9.16: Range test, critical values, model 9.48

OrcaFlex Design, 285 m Parameter	Value		Failure Criteria	
	Value	Unit	Value	Unit
Lift force Crawler	48,68	[kN]	60	[kN]
Tipping moment due to lift force	639,52	[kNm]	840	[kNm]
End moment connection point	95,66	[kNm]	120	[kNm]
Max tension Crawler to buoyancy	50,34	[kN]	60	[kN]
Height lowest point lazy s-shape	-5974,85	[m]	-5982,5	[m]
Maximum tension hose	64,32	[kN]	200	[kN]

Table 9.17: Results OrcaFlex model 50 meters in x direction, range y direction

The 295 and 290-meter range fail on the effective tension in the hose section between the crawler and the buoyancy. This failure happens because the hose is already under tension through its axial deformable catenary shape.

For the calculation of the dynamic model, the OrcaFlex model passes through three stages, of which two are static stages. The static stage is composed of a stage that calculates the line's shape without external forces and one end. The stage without external forces is set on a catenary calculation method. It shows a very small sag, leading to a relatively high horizontal component on the connection points. When external forces are added, the vertical and horizontal components grow and so the resultant force is in line with the hose (effective tension).

#### 9.4.2. Y displacement at 100 meters

In simulation (Appendix E.6), The VTS node is fixed in place, and the crawler is modelled by a constraint, of which the motion is prescribed. The velocity of the crawler is set at 0,5 meters per second. The motion is implemented in the following manner. The approach is different from the implementation of the X-direction. OrcaFlex can calculate the acceleration and the deceleration of the motion. This approach results in less shaky plots. The rapidly varying amplitudes of the figures presented in sub-chapter 9.3 arise from the rapid varying accelerations or decelerations.

Time [s]	X-coordinate [m]	Y-coordinate [m]	Z-coordinate [m]
-8	100	-65	-5992,5
0	100	0	-5992,5
130	100	65	-5992,5
260	100	65	-5992,5
285	100	65	-5992,5
310	100	65	-5992,5
335	100	65	-5992,5
360	100	65	-5992,5

Table 9.18: Movement of the crawler node

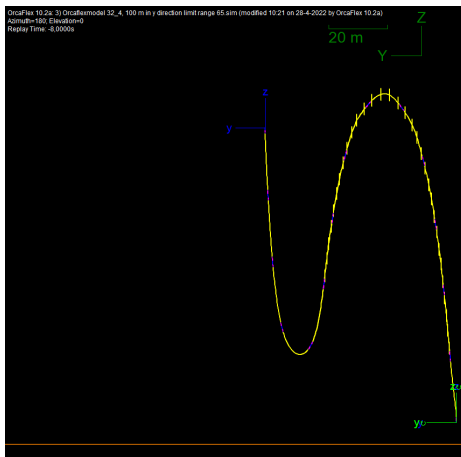


Figure 9.49: OrcaFlex model at 0 seconds

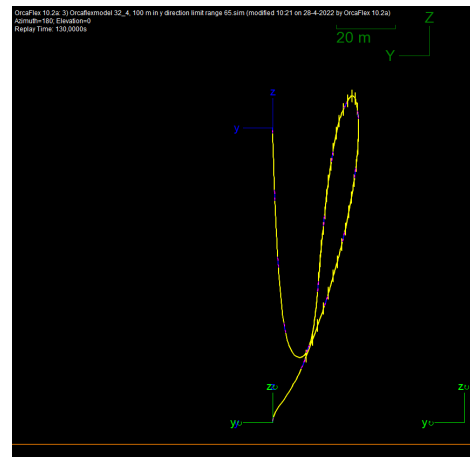


Figure 9.50: OrcaFlex model at 130 seconds

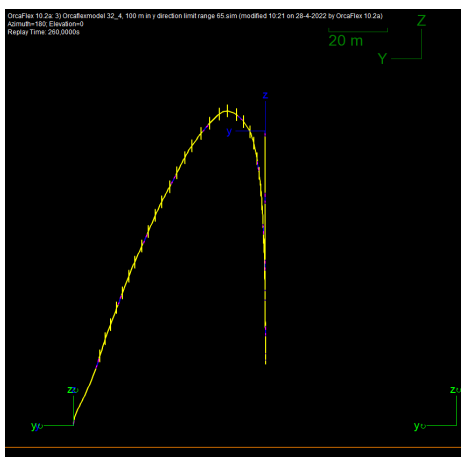


Figure 9.51: OrcaFlex model at 260 seconds

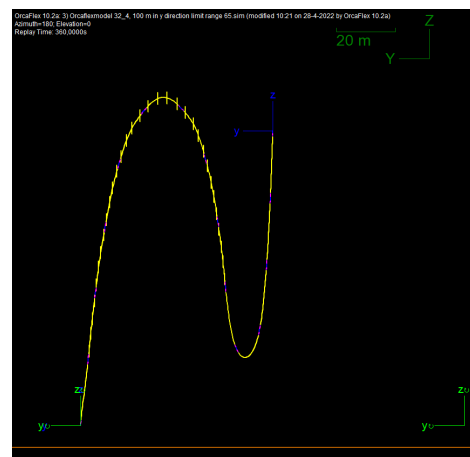


Figure 9.52: OrcaFlex model at 360 seconds

The overview figures illustrate that the crawler drags the hose through the water and the s shape follows with a delay. The hose needs about 130 seconds to form its final shape when the crawler stops. This shape is equivalent to the static equilibrium.

The model is evaluated on the same criteria and locations as the x displacement models, subsection 9.2.5. The outcome is presented in table 9.19.

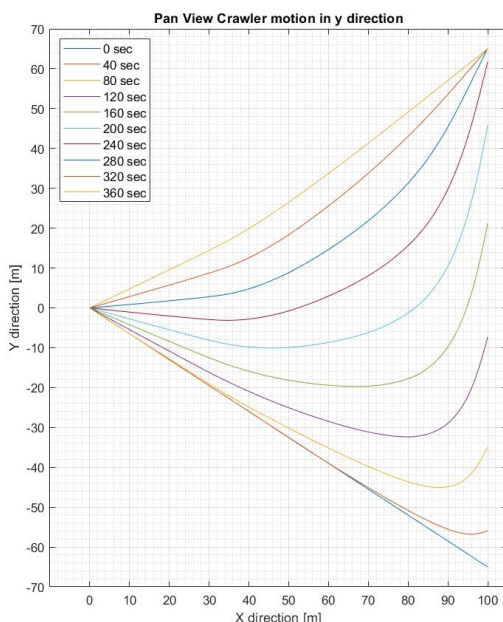


Figure 9.53: Time laps, x 100 m range 65 m, pan view

OrcaFlex Design Parameter	Value	Unit	Failure Criteria	
			Value	Unit
Lift force Crawler	15,15	[kN]	60	[kN]
Tipping moment due to lift force	212,10	[kNm]	840	[kNm]
End moment connection point	38,34	[kNm]	120	[kNm]
Max tension Crawler to buoyancy	26,29	[kN]	60	[kN]
Height lowest point lazy s-shape	-5971,76	[m]	-5982,5	[m]
Maximum tension hose	44,76	[kN]	200	[kN]

Table 9.19: Results OrcaFlex model 100 meters in x direction, 65 m in y-direction

9.4.2.1. Y displacement at 100 meters, limit range

The displacement vector of simulation (Appendix E.6) is expanded to the 144-meter range and the 215-meter range. It is concluded that these ranges are feasible and that the forcing values do not reach the critical values. The range value is increased further.

OrcaFlex Design, 144 range Parameter	Value		Unit		Range, 215		Failure Criteria	
	Value	Unit	Value	Unit	Value	Unit	Value	Unit
Lift force Crawler	16,95	[kN]	18,48	[kN]	60	[kN]	60	[kN]
Tipping moment due to lift force	237,30	[kNm]	258,72	[kNm]	840	[kNm]	840	[kNm]
End moment connection point	46,03	[kNm]	50,08	[kNm]	120	[kNm]	120	[kNm]
Max tension Crawler to buoyancy	26,76	[kN]	28,71	[kN]	60	[kN]	60	[kN]
Height lowest point lazy s-shape	-5971,99	[m]	-5972,09	[m]	-5990	[m]	-5990	[m]
Maximum tension hose	45,13	[kN]	45,51	[kN]	200	[kN]	200	[kN]

Table 9.20: Range test, critical values, model 9.53

The range is elevated with steps of 25 meters to a maximum of 285 meters. The length of a straight line can not be higher than the total length of the jumper hose. Table 9.21 shows that the values increase slowly until the last step, from 265 meters to 285 meters. From the 285 meter model, the range is scaled back with steps of 5 meters per step, and the first step that stayed beneath the failure criteria is 270 meters, and the values are placed in table 9.22.

OrcaFlex Design Parameter	240 range		Range, 265		Range, 285		Failure Criteria	
	Value	Unit	Value	Unit	Value	Unit	Value	Unit
Lift force Crawler	23,31	[kN]	37,14	[kN]	118,77	[kN]	60	[kN]
Tipping moment due to lift force	326,34	[kNm]	519,96	[kNm]	1662,8	[kNm]	840	[kNm]
End moment connection point	54,92	[kNm]	81,36	[kNm]	188,35	[kNm]	120	[kNm]
Max tension Crawler to buoyancy	32,93	[kN]	44,89	[kN]	124,41	[kN]	60	[kN]
Height lowest point lazy s-shape	-5972,11	[m]	-5972,13	[m]	-5972,15	[m]	-5982,5	[m]
Maximum tension hose	48,22	[kN]	57,94	[kN]	132,76	[kN]	200	[kN]

Table 9.21: Range test, critical values, model 9.53

OrcaFlex Design, 270 m Parameter	Value		Failure Criteria	
	Value	Unit	Value	Unit
Lift force Crawler	43,00	[kN]	60	[kN]
Tipping moment due to lift force	602,00	[kNm]	840	[kNm]
End moment connection point	91,47	[kNm]	120	[kNm]
Max tension Crawler to buoyancy	50,50	[kN]	60	[kN]
Height lowest point lazy s-shape	-5972,13	[m]	-5982,5	[m]
Maximum tension hose	62,81	[kN]	200	[kN]

Table 9.22: Results OrcaFlex model 10 meters in x direction, range y direction, model 9.53

### 9.4.3. Y displacement at 150 meters

In simulation (Appendix E.6), The VTS node is fixed in place, and the crawler is modelled by a constraint, of which the motion is prescribed. The velocity of the crawler is set at 0,5 meters per second. The motion is implemented in the following manner. The approach is different from the implementation of the X-direction. OrcaFlex can calculate the acceleration and the deceleration of the motion. This approach results in less shaky plots. The rapidly varying amplitudes of the figures presented in subchapter 9.3 arise from the rapid varying accelerations or decelerations.

Time [s]	X-coordinate [m]	Y-coordinate [m]	Z-coordinate [m]
-8	150	-47	-5992,5
0	150	-47	-5992,5
130	150	0	-5992,5
260	150	47	-5992,5
285	150	47	-5992,5
310	150	47	-5992,5
335	150	47	-5992,5
360	150	47	-5992,5

Table 9.23: Movement of the crawler node



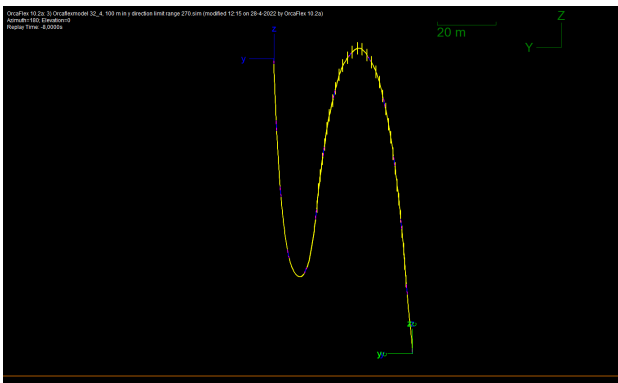


Figure 9.54: OrcaFlex model at 0 seconds

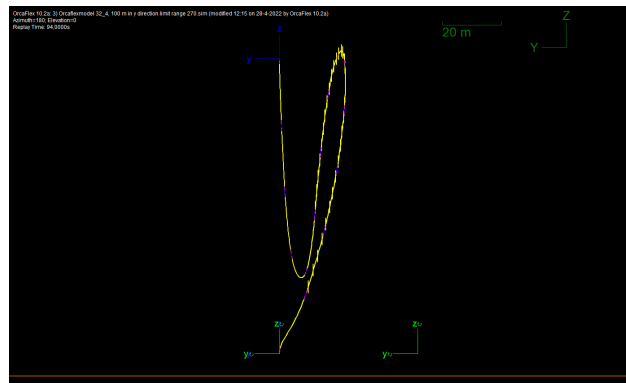


Figure 9.55: OrcaFlex model at 94 seconds

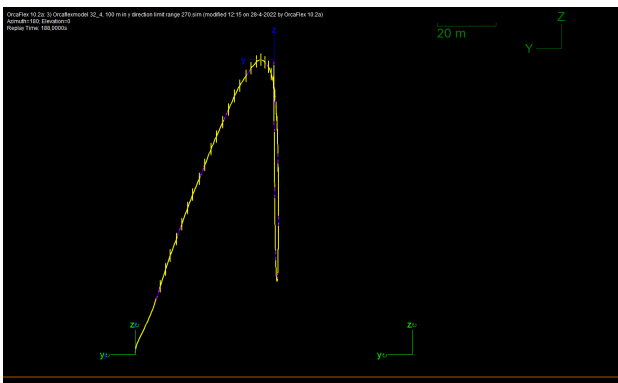


Figure 9.56: OrcaFlex model at 188 seconds

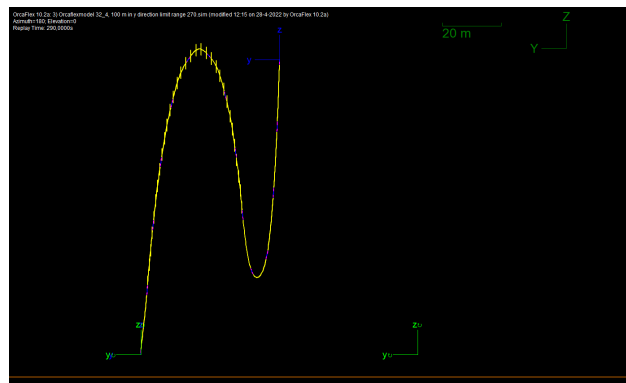


Figure 9.57: OrcaFlex model at 290 seconds

The overview figures illustrate that the crawler drags the hose through the water and the s shape follows with a delay. The hose needs about 100 seconds to form its final shape when the crawler stops. This shape is equivalent to the static equilibrium.

The model is evaluated on the same criteria and locations as the x displacement models, subsection 9.2.5. The outcome is presented in table 9.24.

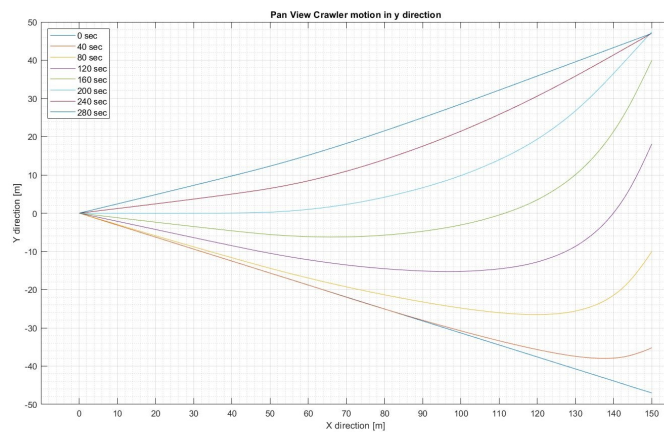


Figure 9.58: Time laps, pan view

OrcaFlex Design Parameter	Value		Failure Criteria	
	Value	Unit	Value	Unit
Lift force Crawler	14,99	[kN]	60	[kN]
Tipping moment due to lift force	209,86	[kNm]	840	[kNm]
End moment connection point	36,59	[kNm]	120	[kNm]
Max tension Crawler to buoyancy	26,18	[kN]	60	[kN]
Height lowest point lazy s-shape	-5967,62	[m]	-5982,5	[m]
Maximum tension hose	44,79	[kN]	200	[kN]

Table 9.24: Results OrcaFlex model 150 meters in x direction, 47 m in y-direction

#### 9.4.3.1. Y displacement at 150 meters, limit range

The displacement vector of simulation (Appendix E.6) is expanded to the 132-meter and 198-meter ranges. It is concluded that these ranges are feasible and that the forcing values do not reach the critical values. The range value is increased further.

OrcaFlex Design, Parameter	OrcaFlex Design, 132 range		Range, 198		Failure Criteria	
	Value	Unit	Value	Unit	Value	Unit
Lift force Crawler	17,75	[kN]	20,04	[kN]	60	[kN]
Tipping moment due to lift force	248,50	[kNm]	280,56	[kNm]	840	[kNm]
End moment connection point	47,34	[kNm]	51,69	[kNm]	120	[kNm]
Max tension Crawler to buoyancy	27,04	[kN]	30,38	[kN]	60	[kN]
Height lowest point lazy s-shape	-5967,29	[m]	-5967,38	[m]	-5982,50	[m]
Maximum tension hose	45,13	[kN]	46,40	[kN]	200	[kN]

Table 9.25: Range test, critical values, model 9.58

The range is elevated with steps of 25 meters to a maximum of 260 meters. The length of a straight line can not be higher than the total length of the jumper hose. Table 9.26 shows that the values increase slowly until the last step, from 248 meters to 260 meters. From the 260 meter model, the range is scaled back with steps of 5 meters per step, and the first step that stayed beneath the failure criteria is 250 meters, and the values are placed in table 9.27.

OrcaFlex Design Parameter	223 range		Range, 248		Range, 260		Failure Criteria	
	Value	Unit	Value	Unit	Value	Unit	Value	Unit
Lift force Crawler	26,83	[kN]	46,55	[kN]	92,06	[kN]	60	[kN]
Tipping moment due to lift force	375,62	[kNm]	651,70	[kNm]	1288,84	[kNm]	840	[kNm]
End moment connection point	62,42	[kNm]	97,02	[kNm]	158,67	[kNm]	120	[kNm]
Max tension Crawler to buoyancy	35,83	[kN]	53,61	[kN]	97,58	[kN]	60	[kN]
Height lowest point lazy s-shape	-5967,40	[m]	-5967,42	[m]	-5967,43	[m]	-5982,5	[m]
Maximum tension hose	50,54	[kN]	65,56	[kN]	107,61	[kN]	200	[kN]

Table 9.26: Range test, critical values, model 9.58

OrcaFlex Design, 250 m Parameter	Value	Unit	Failure Criteria	
			Value	Unit
Lift force Crawler	50,15	[kN]	60	[kN]
Tipping moment due to lift force	702,10	[kNm]	840	[kNm]
End moment connection point	91,47	[kNm]	120	[kNm]
Max tension Crawler to buoyancy	56,88	[kN]	60	[kN]
Height lowest point lazy s-shape	-5967,42	[m]	-5982,5	[m]
Maximum tension hose	68,61	[kN]	200	[kN]

Table 9.27: Results OrcaFlex model 10 meters in x direction, range y direction

### 9.4.4. Analysis conclusion

The displacement in the y-direction is analysed and results in a wider range than predicted. The assumption made for an x-position of 200 meters is no longer valid and is analysed below. The lift force at the connection point with the crawler is the governing parameter for the design. It reaches its critical value if the hose experiences stress because the distance between the connection points is close to the actual hose length. These disclosures are further elaborated in chapter 11

## 9.5. Range Crawler and research question 6 and 7

As stated in subsection 9.4.4, the connection point between the jumper hose and crawler is elaborated to find the full range of motion of the crawler if the VTS is kept in place. This calculation resulted in the following ranges for different offsets. The same procedure and criteria are maintained as in subchapter 9.3 and 9.4. The results are plotted in figure 9.59 and corresponding table 9.28

This

Distance between crawler and VTS		Perpendicular range			
Value	Unit	Value	Unit	Value	unit
0	[m]	-290	[m]	290	[m]
25	[m]	-290	[m]	290	[m]
50	[m]	-285	[m]	285	[m]
100	[m]	-270	[m]	270	[m]
150	[m]	-250	[m]	250	[m]
200	[m]	-210	[m]	210	[m]
225	[m]	-185	[m]	185	[m]
250	[m]	-150	[m]	150	[m]
275	[m]	-100	[m]	100	[m]
290	[m]	-30	[m]	30	[m]

Table 9.28: Range Crawler to VTS

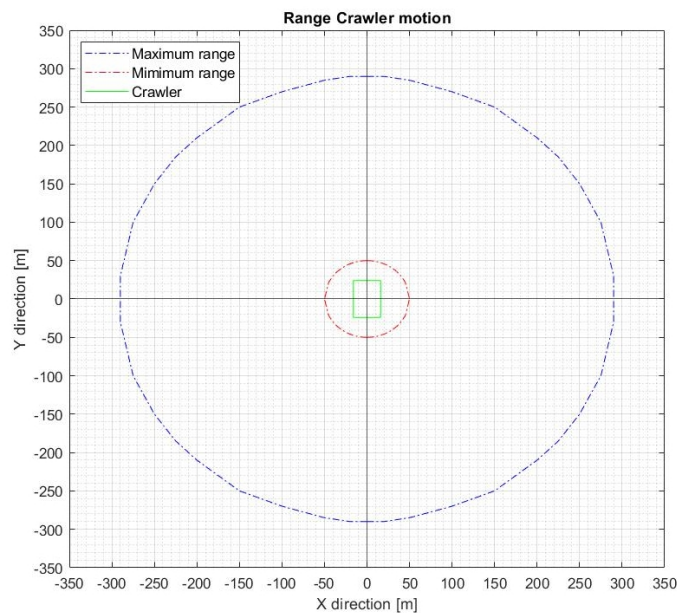


Figure 9.59: Floor plan, range crawler

subchapter concludes the answer to the last research question: *How manoeuvrable is the seafloor mining tool?* The range of the crawler is presented in figure 9.59 and shows the combination between an inline and perpendicular motion. To make an optimal choice for the range and path of the crawler need to be compared to other parameters. Chapter 11 includes a constant motion of the clump weight, which influences the range of the crawler.

There are 2 models made that give a good starting point for the jumper hose dynamic analysis. These 2 models partly answer the 6th research question *Which simulations are performed for the dynamic analysis?*. In chapter 11 extensions of this model are evaluated, and a critical evaluation of

parameters which were approximated and excluded.

## Conclusion

The main objective of this thesis was to investigate the jumper hose design of the blue nodule project and develop a model for the dynamic analysis. The objective is linked to the research question:

*Determine the dynamical behaviour of flexible hose in three dimensions and its interaction with the moving seafloor mining tool during different scenario's.*

The conclusion of this thesis is formulated based on the research questions and the sub research questions. The sub research questions are answered at the end of a related chapter and are listed in the contents and subchapter title. The research is divided into two main objectives to get an insight into the jumper hose's dynamical behaviour. First, a static model is developed to analyse the governing parameters. This analytical model aims to perform a parameter study to obtain an improved set of design parameters for the jumper hose. IHC requested to vary the length of the hose. Second, the analytical model is implemented in the OrcaFlex software. OrcaFlex is a software program for static and dynamic analysis for offshore systems. It resulted in three models to create the final dynamic model. The first model was the implementation of the analytical model and the verification. The second model was to update the simplified parameters to the real design parameters and elements. The last model was the implementation of the possible movements of the crawler. Finally, the dynamic model was analysed and the design assumptions were critically appraised.

### Analytical model

- The Matlab model is composed of geometrically non-linear elements and is approached as an axial deformable catenary model.
- Resulted in the first parameter design set, table 10.1 and figure 10.1

Parameter	Value
Total length of the hose	320,04
Segment hose length	22,86
Number of elements to evaluate segment	5
Element length	4,572
Height of VTS connection above seafloor	100
Range of crawler	200
Number of buoyancy modules	25
Buoyancy modules per hose segment	3
Starting point buoyancy	After 1st
Content hose/Slurry density	1200

Table 10.1: Concept design parameters, subchapter 7.5

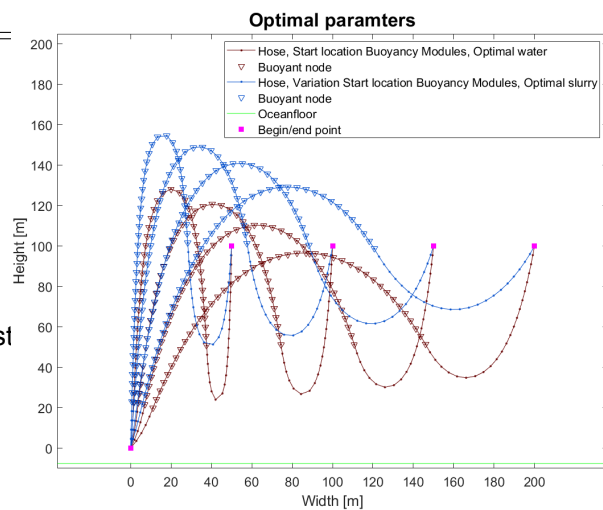


Figure 10.1: Optimal parameter set, static analysis

### Orcaflex model

- The first model verified the analytical MatLab model and its design parameters
- The second model implemented design parameters: the buoyancy modules, bending stiffness, segment length, and jumper hose assembly and its material parameters. This implementation resulted in a collapse of the design parameters of the analytical model, mainly due to the material parameters and the weight.
- The second model caused a do-over of the parameter study and produced a new set of parameters, which did comply with the set criteria, table 10.2.
- The implementation of the possible movements of the crawler is split up into detached displacements to create the displacement scenarios. Starting with the conduction of a 2D inline motion and the extracting of significant results.
- The conduction of the 2D motion was successful and a perpendicular motion was implemented and analysed. From the combination of the inline and perpendicular motion a range arises, figure 10.2.

Parameter	Value	Ur
Total length of the hose	320,04	[m]
Segment hose length	22,86	[m]
Number of elements to evaluate segment	5	[-]
Element length	4,572	[m]
Height of VTS connection above seafloor	100	[m]
Range of crawler	200	[m]
Number of buoyancy modules	32	[-]
Buoyancy modules per hose segment	4	[-]
Starting point buoyancy	After 1st ho:	
Content hose/Slurry density	1200	[kg]

Table 10.2: Final design parameters, subchapter 8.2

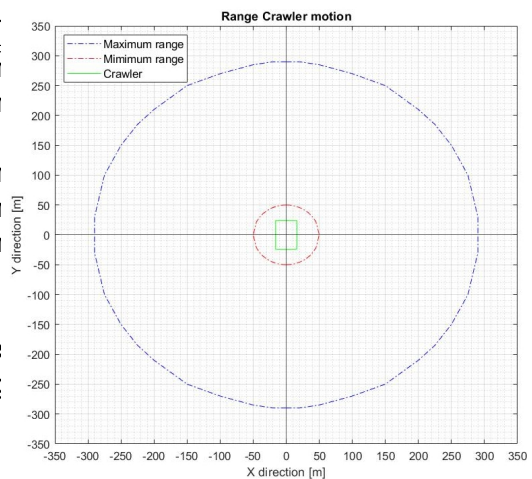


Figure 10.2: Floor plan, range crawler

The last model is an approximate version of the actual design of the Blue nodule project. The assumptions and simplifications are assessed and judged in chapter 11. This chapter also included some extra scenarios and gave concept plans for further developments/extensions for this research.

## 10.1. Conclusion to the research question

The new design parameters for the jumper hose are displayed in table 10.2 and comply with the set failure design criteria. The dynamical behaviour and movements of the jumper hose are manageable (comply with the failure criteria) for the impost displacement of the crawler in the X-Y-plane if the displacement stays in the predetermined range. Other scenarios like displacement of the clump weight and influence of environmental sea surface motions are discussed in chapter 11.

# Recommendations and Review

In this chapter, the simplifications and assumptions are reviewed and criticized. Subchapter 9.2 describes the system dynamic and the dynamic motions that affect the jumper hose design. The movements of the clump weight due to the wind, waves and currents are disregarded. In this chapter, a few of the approximated or disregarded motions are developed to investigate the effect on the jumper hose design.

Starting with the review of the parameter study and the design criteria in subchapter 11.1. During the dynamic study, it is discovered that the tension stays roughly constant until the distance between the crawler and the clump weight gets close to a straight line. The following subchapters evaluate some simplifications and extend the dynamic analysis. Subchapter 11.2 develops the heave motions and their effect on the jumper hose. The summarized conclusion is written in subchapter 11.6. Subchapter 11.3 implements a 3D motion of the crawler and analysis its results. The motions are visually plotted and the production is calculated. The set speed for the crawler is evaluated and the value is elaborated. The summarized conclusion is written in subchapter 11.6. Subchapter 11.4 analyses the Displacement of the clump weight and the crawler in the same direction. The summarized conclusion is written in subchapter 11.6. And the last subchapter that contributes to the dynamic design is subchapter 11.4, which combines the 3D motion of the crawler and the Displacement of the clump weight. This results in a new working area of the crawler. The summarized conclusion is written in subchapter 11.6.

The last section provides a more complete answer to the last to research questions.

## 11.1. Review of the parameter study

The parameter study is performed with a set of design criteria from the blue nodule design. But minimum values for the tension are not taken into account. This could lead to a new set of design parameters. The three main parameters that influence the tension in the hose are:

- total number of modules
- modules per section
- start section buoyancy modules

The total number of modules results in the hose's total uplift force and directly influences the crawler's range because the tension is higher for the same static parameters. To a lesser extent, they influence the shape of the hose, which results in a greater distance between the hose and the seabed.

The number of modules per section influences the steepness of the s-shape curve and the number of modules that directly lift the crawler, which again influences the range of the crawler.

At last, the starting section of the buoyancy models parameter. This parameter influences the shape of the hose and the lift force on the crawler. Suppose the buoyancy modules start a section later than in the initial design. In that case, the modules (located until the s-shapes top) need to lift more material, resulting in a lesser tension. This parameter depends heavily on the other two.

It is recommended to reevaluate the failure criteria and the importance of the criteria

## 11.2. Heave motion on clump weight

A neglected motion on the jumper hose is the environmental movement of the sea surface. The mining system is subjected to several environmental dynamic effects. Document "Blue nodules 2.10" stated the metocean data for the sea surface, table 3.6. It described that the VTS is not heave compensated and the surface vessel motions are transferred through the VTS to the clump weight. The harvesting vehicle is steady on the seafloor. No motions from the sea surface were considered on the harvesting vehicle. This means that the clump weight and the jumper hose absorb these motions. The surface vessel is subjected to waves, wind and current and experiences motions in all 6 DOF. For this model, only the heave motion is analysed.

The maximum regular wave height is 5.73 meters from crest to trough. This wave height is directly transferred to the clump weight due to the lack of heave compensation in the VTS. From the clump weight's neutral position, this results in a displacement of 2,865 meters in the positive and negative z-direction. The crawler is kept in place.

Figure 11.1 and 11.2 is shown that at the connection point of the clump weight and the jumper hose, the imposed heave motion creates a fluctuating tension force due to the pulling and pushing of the clump weight. The results of table 11.1 show that the imposed motion does not affect the maximum values of the evaluated parameters. Still, figure 11.3 and 11.4 show the same oscillating movement in node 29 (maximum tension) and at the connection point between the crawler and the jumper hose. These results support the recommendation of implementing the full mining system and the environmental parameters.

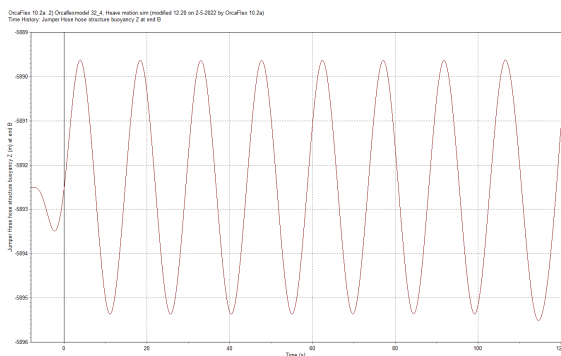


Figure 11.1: OrcaFlex model Motion connection point clump weight

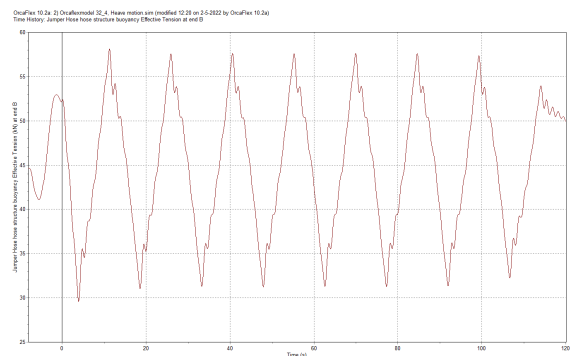


Figure 11.2: OrcaFlex model Tension connection point clump weight

OrcaFlex Design Parameter	Value	Unit	Failure Criteria	
			Value	Unit
Max tension Crawler to buoyancy	27,096	[kN]	60	[kN]
Height lowest point lazy s-shape	-563,52	[m]	-5982,5	[m]
Maximum tension hose	58,11	[kN]	200	[kN]

Table 11.1: Results OrcaFlex model heave motion at connection clump weight



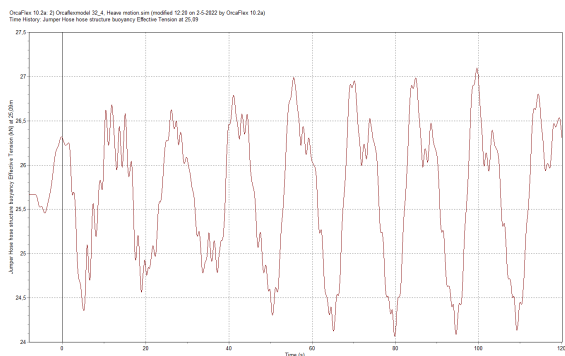


Figure 11.3: OrcaFlex model Tension node 29 connection point clump weight

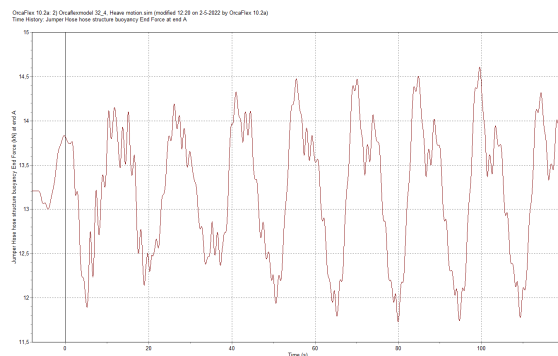


Figure 11.4: OrcaFlex model End force connection point

### 11.2.1. Heave motion and crawler motion

The same crawler motion as described in subchapter 9.3 is implemented at the end node of the jumper hose model. Combined with the heave motion on the connection point with the clump weight, this movement results in the following effective tension graph over the arc length.

It still shows the oscillations as described above. The overall maximum and minimum at the connection point with the clump weight show the same values as figure 11.4. Another remarkable result is shown in figure 11.5, the minimum tension in the hose goes towards and comes close to 0 kN. This could lead to a switch from tension to compression in the hose and could have consequences for the material parameters of the hose. This thesis's minimum tension and value are not clearly established or investigated. Its recommended to evaluate and perhaps update the failure criteria.

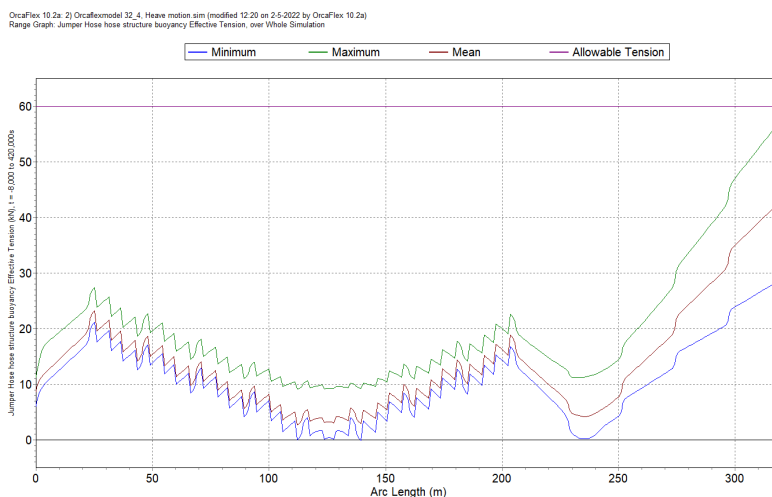


Figure 11.5: OrcaFlex model Effective tension over arce length

### 11.2.2. Heave motion and crawler motion, perpendicular

To analyse a perpendicular motion in combination with a heave motion the model form subchapter 9.4.2 is chosen and the same crawler motion is implemented at the end node of the jumper hose model. Combined with the heave motion on the connection point with the clump weight, this movement results in the following effective tension graph over the arc length.

This model is created to evaluate the influence of the heave motion on the range of the crawler. Table 11.4 presents the values of this model. Suppose the values are compared to the values of table

9.19. In that case, it is concluded that implementing the heave motion results in a higher maximum tension for both evaluated locations. It concluded that a heave force results in a smaller range of the crawler, and it is recommended to investigate the following elements of the model:

- The connection point between the crawler and the jumper hose. Can it handle the fluctuating forces?
- The minimum tensions in the hose

OrcaFlex Design Parameter	Value	Unit	Failure Criteria	
			Value	Unit
Liffforce at crawler	27,61	[kN]	60	[kN]
Max tension Crawler to buoyancy	40,013	[kN]	60	[kN]
Maximum tension hose	63,94	[kN]	200	[kN]

Table 11.2: Results OrcaFlex model, heave motion and perpendicular motion

### 11.3. Pattern crawler for collection nodules

In subchapter 9.3 and 9.4, the inline and perpendicular movements of the crawler are analysed. This resulted in a range of motion for the crawler. The "Blue Nodules Deliverable report D2.7 Detailed design of jumper hose" developed a path for the crawler to collect nodules. The initial design parameters are:

Parameter	Value	Unit
Width of the crawler	16	[m]
Maximum speed of the crawler	0,5	[m/s]
Average nodule collection for maximum speed	145	[kg/s]
Minimum radius to turn crawler	30	[m]
Set range crawler	80	[m]

Table 11.3: Design parameters path of the crawler

These parameters resulted in the following pattern for the crawler, keeping the initial parameters in mind.

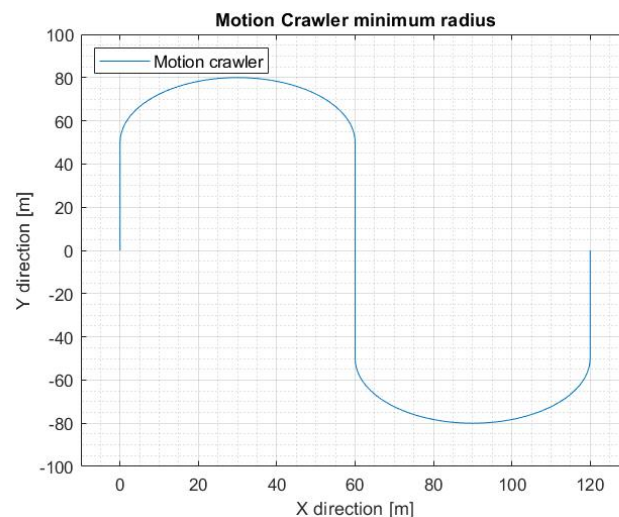


Figure 11.6: Path crawler with minimum rotation radius

One rotation means that the crawler travels two turns and ends up on the same coordinates beside its x-position. The length of the path is 388.50 meters long and is completed in 778 seconds if the

crawler has a constant speed. This motion is implemented in OrcaFlex and results in the following results if the connection with the clump weight is kept in place (starting location crawler (200 m, 0 m)).

OrcaFlex Design Parameter	Value		Failure Criteria	
	Value	Unit	Value	Unit
Liftforce at crawler	18,52	[kN]	60	[kN]
Max tension Crawler to buoyancy	29,33	[kN]	60	[kN]
Maximum tension hose	45,15	[kN]	200	[kN]

Table 11.4: Results OrcaFlex model, heave motion and perpendicular motion

It is assumed that the crawler can handle the maximum set speed and does not tip over in the curves. This is also not part of the scope of the thesis.

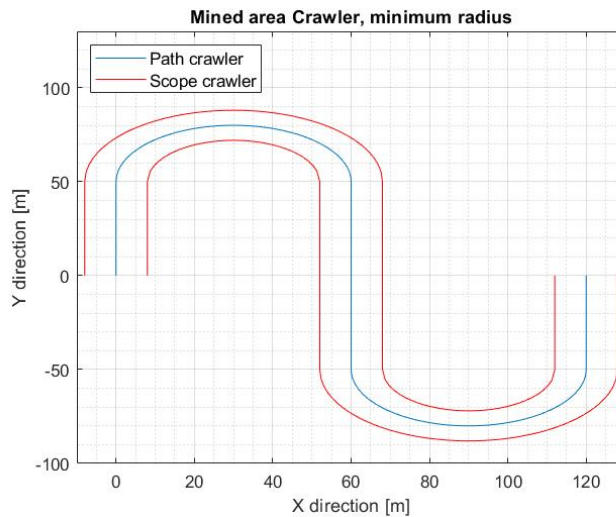


Figure 11.7: Path crawler with minimum rotation radius, scope crawler

One rotation results in approximately 112810 kg nodules (wet density) and covers 6216 m<sup>2</sup> of the seabed.

### 11.3.1. Crawler speed

The maximum speed of the crawler is set to 0,5 m/s. The results of the enhanced speed of the crawler to 1 m/s are displayed below and compared to the 0,5 m/s option. The Force and moment on the connection between the crawler and the jumper hose are higher than the recommended maximum velocity. This is because the crawler pulls faster and more on the jumper hose than the hose moves through the water, resulting in a larger pulling force. It is also derived from figure 11.8 showing the shape of the hose at 75 seconds in the simulation and the crawler has travelled 75 meters. The hose is situated behind the crawler. Figure 11.9 shows the shape of the hose at 150 seconds and 150 meters travelled. Concluding that the speed of the crawler influences the forcing on the connection point.

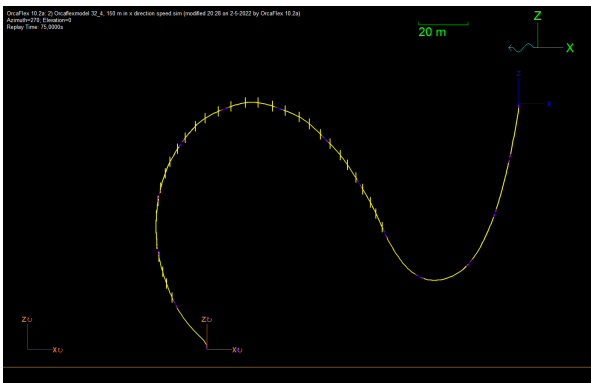


Figure 11.8: OrcaFlex model crawler speed 1 m/s, 75 sec

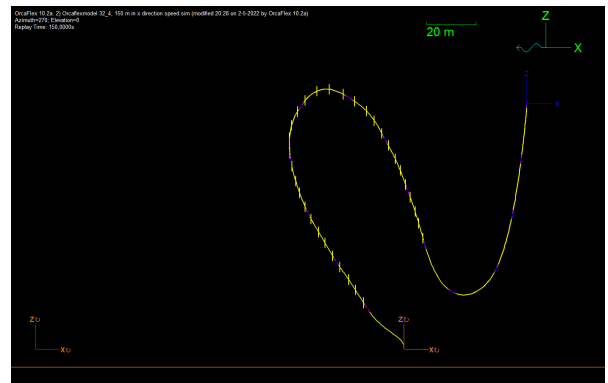


Figure 11.9: OrcaFlex model crawler speed 1 m/s, 150 sec

## 11.4. Displacement of clump weight

As designed by the blue nodules project, the subsea harvesting equipment collects nodules by driving the crawler over a predetermined path on the seafloor and sailing with the vessel to displace the VTS and the clump weight and the jumper hose. The displacement of the clump weight is not incorporated in chapter 9. But it influences the jumper hose and specific the connection between the clump weight and the jumper hose and the next hose sections.

The clump weight displacement arises from the vessel's sailing and the vessel dragging the VTS through the water. In this model, the VTS and vessel are disregarded. It is decided that the properties of the clump weight are also disregarded. The clump weight is modelled as a constraint with a predetermined constant motion.

The crawler has a maximum velocity of 0,5 m/s, so the velocity of the clump weight can hold up to 0,5 m/s for an inline motion. If the clump weight moves faster, the spacing between the crawler and the clump weight grows and as shown in subchapter 9.4, there is a maximum distance feasible before the model fails due to the set criteria.

Figure 11.12 shows the shape of the hose at time intervals of 250 seconds. The results from OrcaFlex are approximately the same as for the inline motions of subchapter 9.3. After 250 seconds, the hose constantly moves through the water. Effects like vortex-induced vibrations are not incorporated into the model.

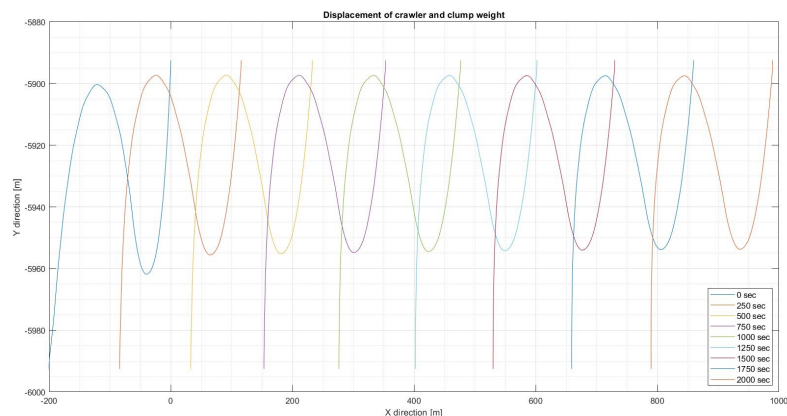


Figure 11.10: OrcaFlex model time steps VTS and crawler same speed in x-direction

## 11.5. Displacement of clump weight and crawler

To complete the design, the constant velocity of the clump weight and the pattern of the crawler are combined. First, a model based on the crawler motions of the blue nodule project, as presented in subchapter 11.3 is implemented. The crawler travels 778 seconds to complete 1 rotation. In x-direction, a distance of 120 meters is crossed. This results in a constant clump weight speed of 0,154 m/s.

OrcaFlex Design Parameter	Value	Unit	Failure Criteria	
			Value	Unit
Liftforce at crawler	21,25	[kN]	60	[kN]
Max tension Crawler to buoyancy	31,50	[kN]	60	[kN]
Maximum tension hose	49,97	[kN]	200	[kN]

Table 11.5: Results orcaflex model

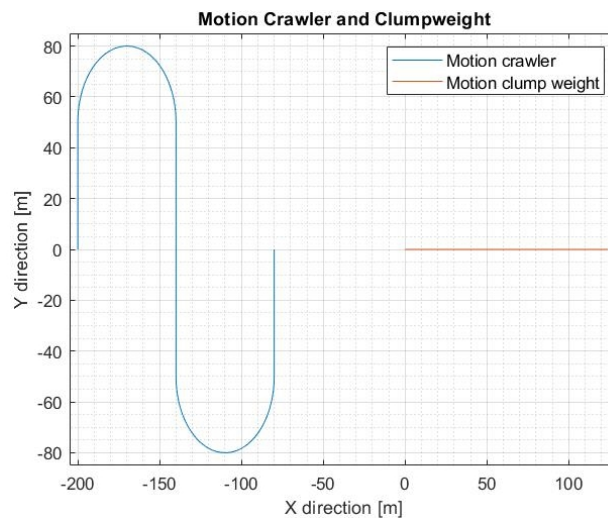


Figure 11.11: OrcaFlex model, crawler pattern and clump weight pattern

The determination of the limit range of the combined motions is approached with the same strategy as for subchapter 9.3 and 9.4. The displacement vector of the zigzag pattern is expended based on the straight part of the pattern. The displacement vector of the clump weight is matched to the new length of the path and the time needed to complete one rotation. The length of the straight part is the starting point of the calculation and is reflected as the path's starting point just before the radius kicks in.

Length straight part [m]	Width 1 rotation [m]	Total length path [m]	Expired time [s]	Clump weight speed [m/s]
50	120	388,50	776,99	0,16
60	120	428,50	830,80	0,144
70	120	468,50	910,8	0,132
80	120	508,50	1016,99	0,121
90	120	548,50	1070,8	0,112
100	120	588,50	1176,99	0,104
120	120	668,5	1310,8	0,092
140	120	748,50	1496,99	0,082
160	120	828,5	1630,8	0,074
180	120	908,5	1816,99	0,067
200	120	988,50	1950,8	0,062
220	120	1068,50	2110,8	0,057
240	120	1148,5	2270,8	0,053
250	120	1188,50	2350,8	0,051

Table 11.6: Length path with corresponding clump weight speeds

Table 11.6 shows the associated clump weight speeds to the length of the zigzag pattern of one rotation. The expired time is calculated based on the implemented locations in OrcaFlex, so the implemented pattern is not a perfect circle. The range is calculated from a set distance between the crawler and the clump weight. Starting at 200 meters distance and then taking step sizes of 20 meters results in the following range plot, which compares the range of subchapter 9.5 to the new range. It shows that the combination of the two motion decreases the maximum range of the crawler.

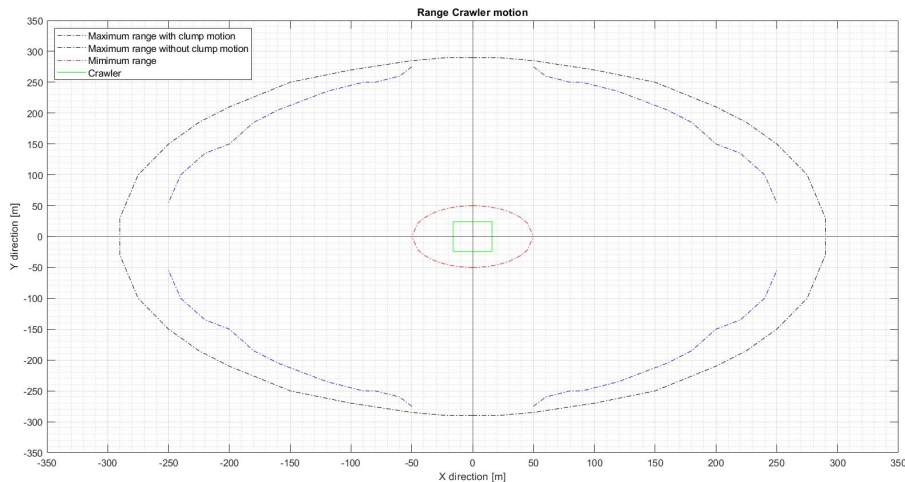


Figure 11.12: OrcaFlex model, crawler pattern and clump weight pattern

This new scope area will reduce when the additional heave motion from the environmental conditions is implemented because subchapter 11.2 shows that the tension fluctuates around the set mean. The mean is equal to the tension from the same crawler motions of subchapter 9.3. It shows that the combination of the two motions decreases the maximum range of the crawler.

It is recommended to investigate the minimum velocity of the sea surface vessel and the delay between the motion of the sea surface vessel and the clump weight. Because for the longest path of table 11.6 the vessel would sail with a 0,1 knot. It is also discouraged to aim for the largest path of one rotation because the jumper hose would go close to the maximum set values for tension and moment

for every rotation. For further design, an assessment could be made to balance out the length of the path, desired area to work in for the crawler and the minimum speed of the sea surface vessel.

## 11.6. Summarized conclusions of investigated parameters

Chapter 11 investigated some of the simplifications and assumptions. It also evaluated motions which were outside of the scope. Starting with the elaboration of implementation of environmental forces. Waves, wind and current were disregarded in the initial design. These mainly influence the surface vessel and the upper part of the VTS. It states that the VTS is not heave compensated and this motion is implemented in a model where the crawler is kept in place and the clump weight experiences the heave motion. The results show that the disregarded environmental forces influence the set failure criteria. It shows that the forces oscillate around a mean value. A question that arises from these results is on: the influence of negative tension, compression in the hose and if there should be a minimum tension in the hose. It is recommended that this is further investigated and analysed.

In chapter 9.5, the range of the crawler is determined, and the working conditions of the blue nodule project show a zigzag pattern for the crawler to scope up the nodules. It was stated that if the crawler travels in a straight line behind the VTS, it experiences many environmental forces that arise from the movement of the VTS. Also, the case where the crawler travels and the VTS follow resulted in failure. This is because the crawler would drag a larger part of the hose forward due to the delay of the clump weight motion through the VTS. This zigzag pattern is incorporated in chapter 11.3 which is a combination of the inline motions and the perpendicular motions.

First, the speed of the crawler was analysed. In "Blue Nodules Deliverable report D 2.7 Detailed design of jumper hose", the maximum velocity of the crawler was set at 0,5 m/s. The influence of a velocity increment is presented, and it is concluded that a higher speed results in a larger pulling force from the crawler and a lagging shape of the hose. These are unwanted effects.

Next, the speed of the VTS was examined by implementing a simple constant motion which has the same speed as the crawler if the crawler and VTS move in the same direction. The environmental effects from the VTS travelling through the water are disregarded. It is concluded that if the clump weight and crawler move in the same direction with the same speed, the hose reaches a constant shape.

To incorporate the movement of the clump weight and the zigzag pattern of the crawler, an average speed is calculated over the travelled distance in the x-direction and the time needed to travel one rotation with the crawler. This approach is selected because if the clump weight moves slower than the crawler, the hose could potentially hit itself when the distance between the crawler and the clump weight decreases. The same goes for if the clump weight moves faster than the crawler, this leads to an increase in distance and a growing tension force, as investigated in the perpendicular motion.

The zigzag pattern and movement of the clump weight are implemented and verified with the failure criteria. Subsequently, the range of this model is tested and results in a new range plot. The next step would be to implement the heave motion because that motion resulted in a higher tension value than for a stationary point.

### 11.6.1. Other disregarded effects

Some effects were disregarded at the start of the set-up of the design models. An example is that the hose transports slurry from the crawler to the vertical transport system. In the models created in chapter 9, the slurry is assumed to be stationary and has a constant density per meter. This is not realistic, so a hypothetical outcome is presented. Transporting the fluid through the s-shape hose leads to a centrifugal force in the bending parts of the hose. This force influences the shape of the hose. It probably also influences the tension force in the hose because of the speed of the slurry and its direction. This motion and effect should not be disregarded for the next design model.

The connection/endpoints of the jumper hose are modelled as swivel connections, meaning they can rotate freely around their axis. In the initial design documents of IHC, nothing is presented over these connection points, so it was assumed to swivel connections to minimize the effect of torsion on

the hose. Suppose the connections were modelled not to be able to rotate around their axis and resist the resulting forces. Then the effect of torsion should be analysed. When the crawler makes a turn, the hose rotates around its axis and these motion / resultant forces are transported through the hose to the connection point at the clump weight. The clump weight can offer resistance to this motion because it is assumed that the connection between the VTS and the clump weight is stronger than the connection between the clump weight and the jumper hose. This results in a resistance force.

## 11.7. Research question 6 and 7

In subchapter 9.5, a layout of the range of the crawler was presented. This displayed the range of the crawler but not its manoeuvrability. Subchapter 11.3 makes a start with the elaboration of the zigzag pattern of the Blue nodules design. The crawler has a minimum rotation radius of 30 meters, resulting in an intermediary distance of 48 meters. For a further design, an efficient study could be done to see which radius and total track result in the most efficient collection of nodules. This thesis answers the last research question: How manoeuvrable is the seafloor mining tool? It is expressed in the range of the crawler.

Chapter 11 elaborated some extra models which contributed to the dynamic analysis of the jumper hose. It also stated that a lot more scenarios are of interest to the design. Appendix E shows a list of Orcaflex models and their goals. This answers the research question, Which simulations are performed for the dynamic analysis?





# Appendix



# A

## Matlab iteration plots

### A.1. Matlab iteration steps, catenary

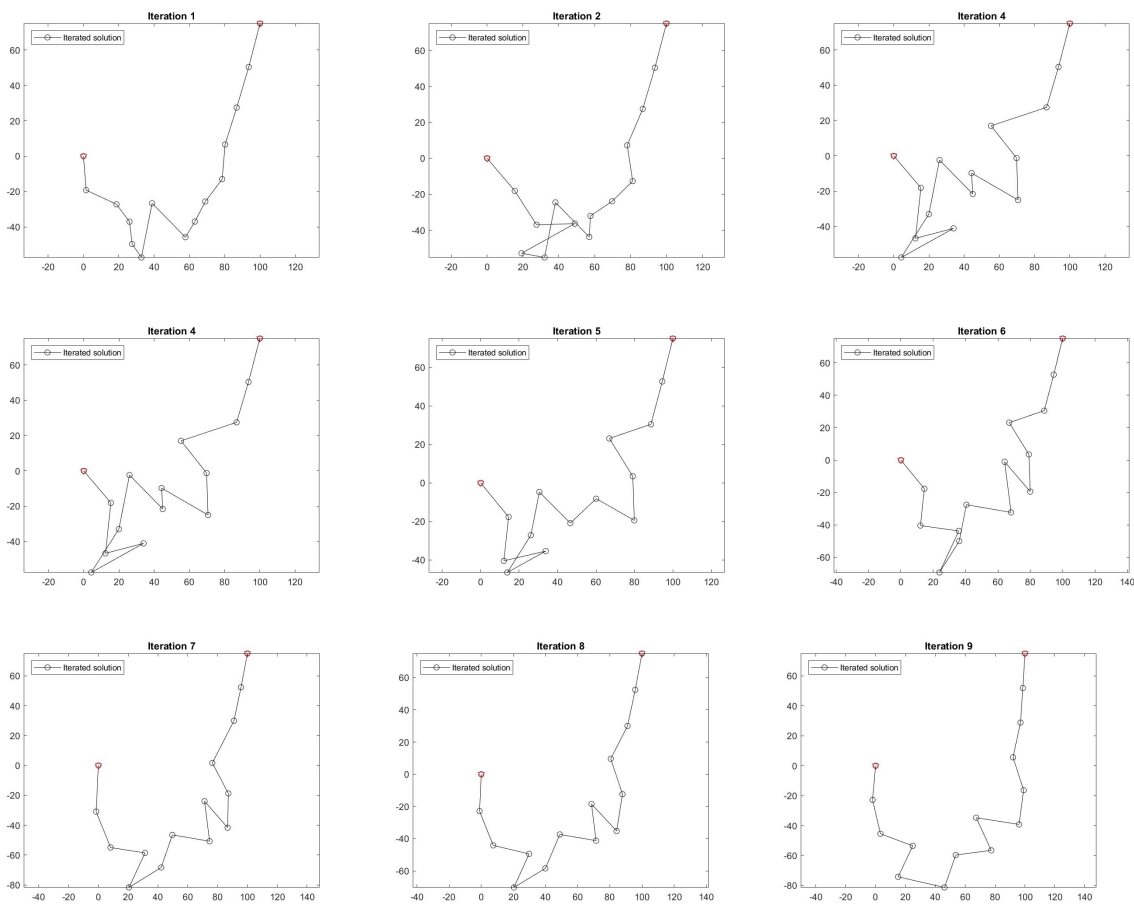
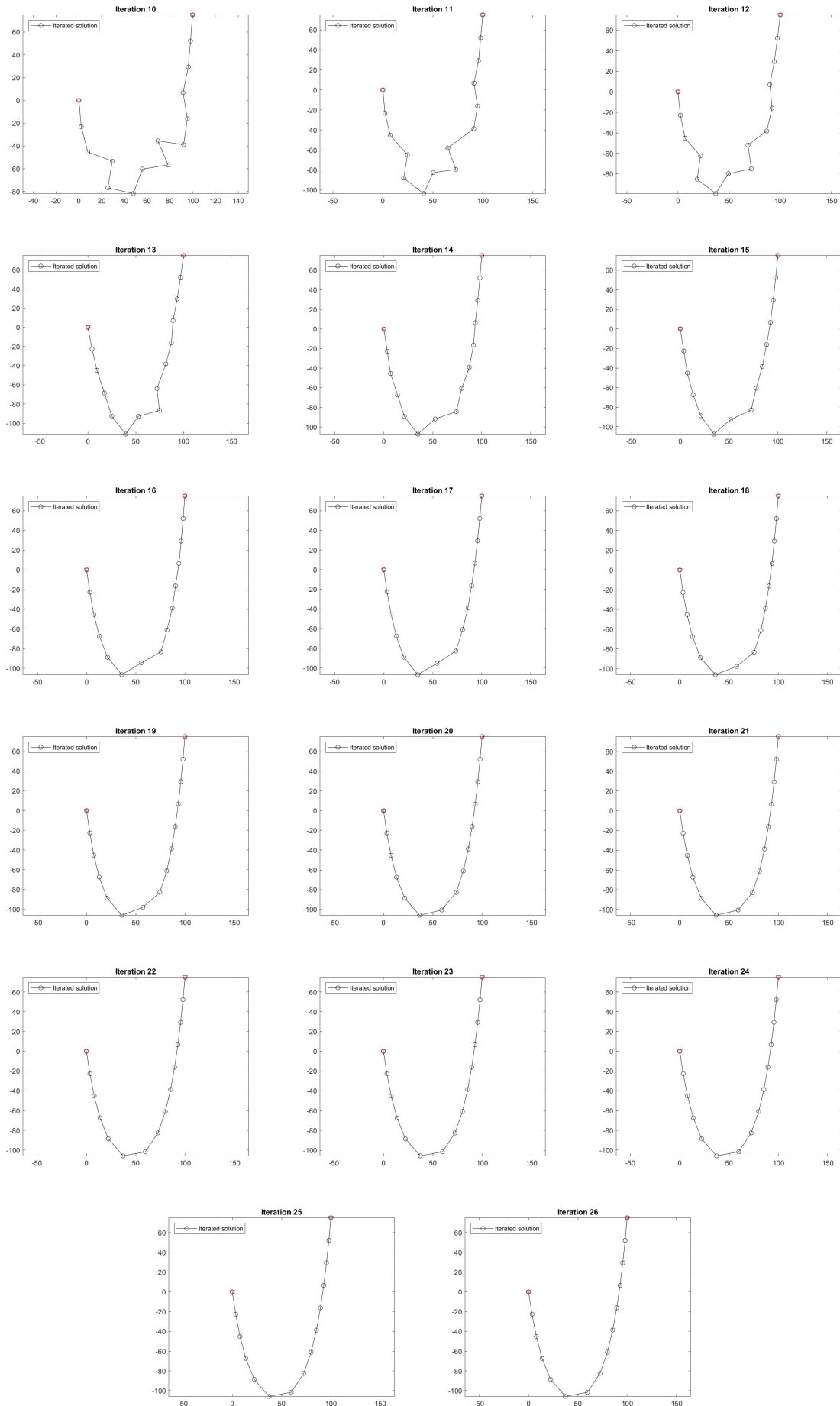


Figure A.1: Iterations Catenary, number 1 to 9



# A.2. Matlab iteration steps, catenary 1 node buoyant

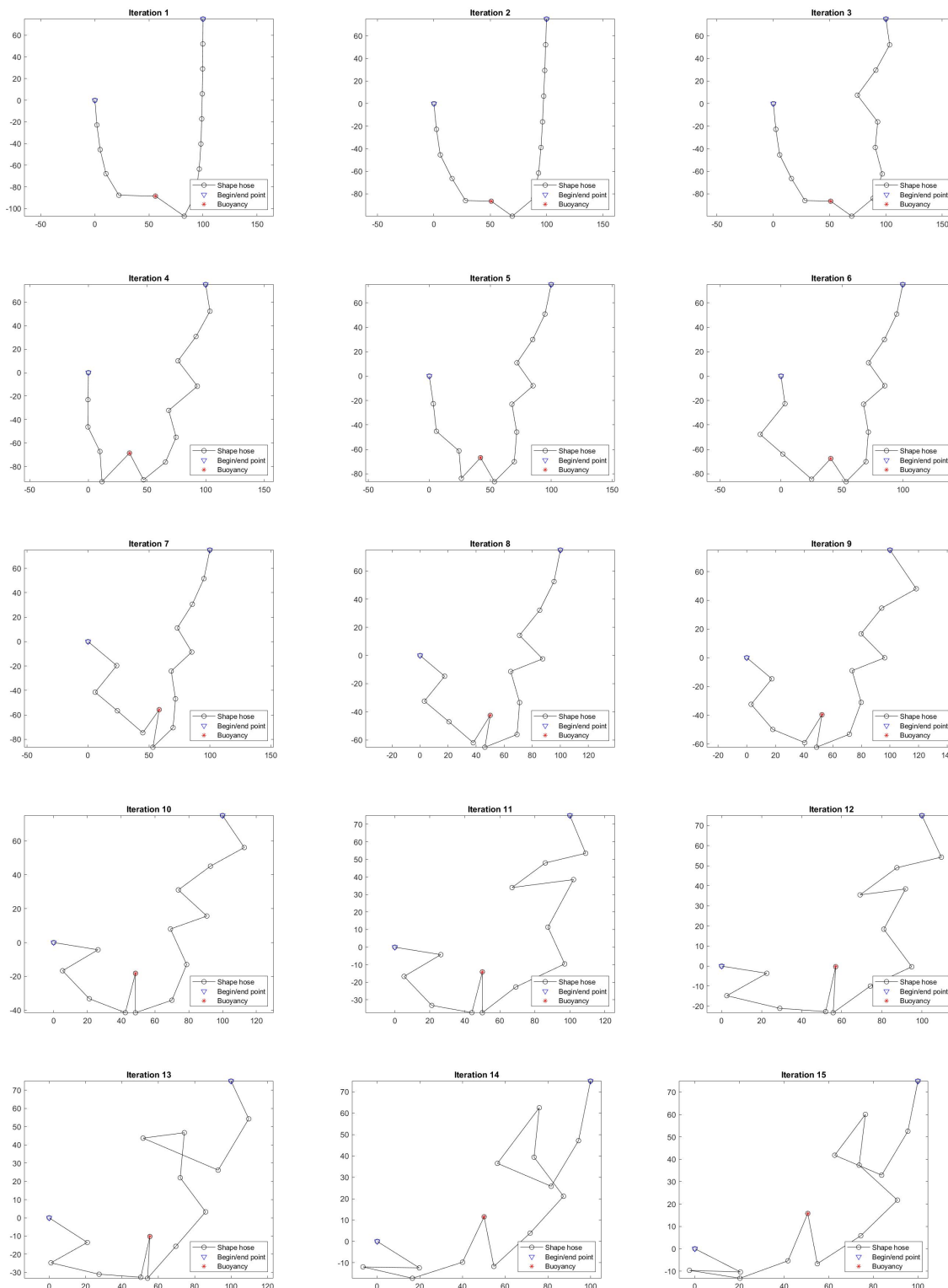


Figure A.3: Iterations 1 Node, number 1 to 15

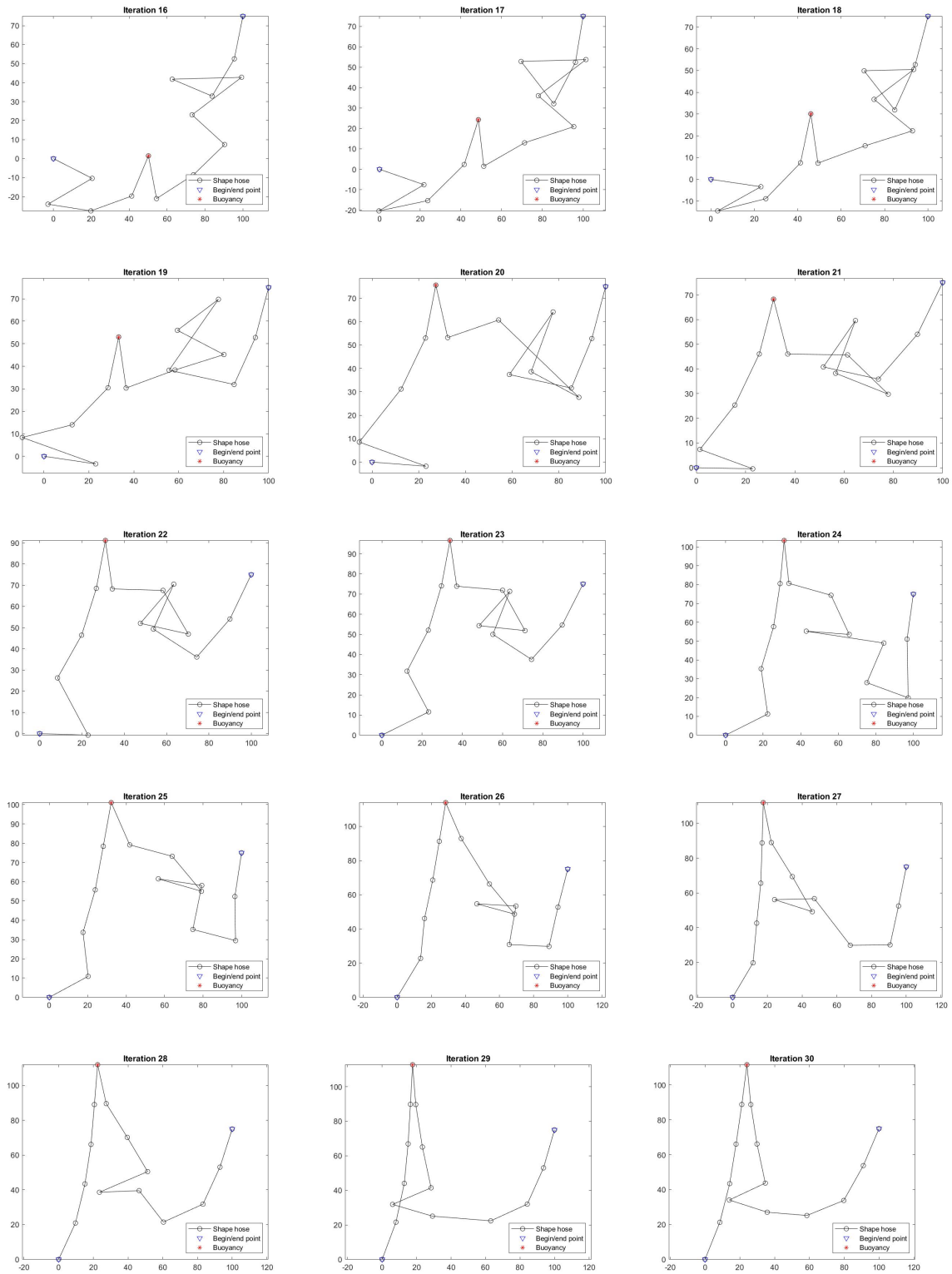


Figure A.4: Iterations 1 Node, number 16 to 30

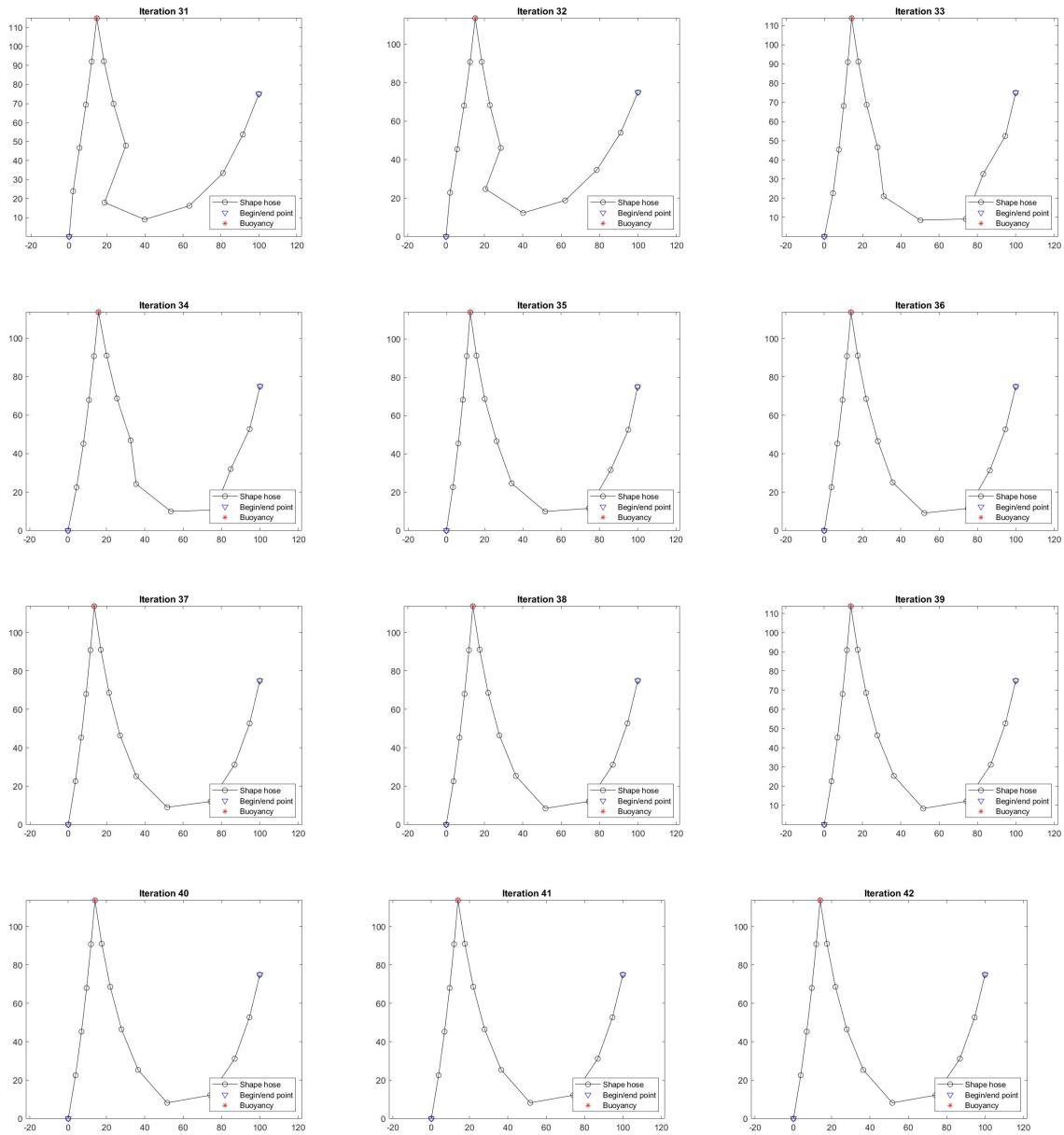


Figure A.5: Iterations 1 Node, number 31 to 42

### A.3. Matlab iteration steps, S-shape

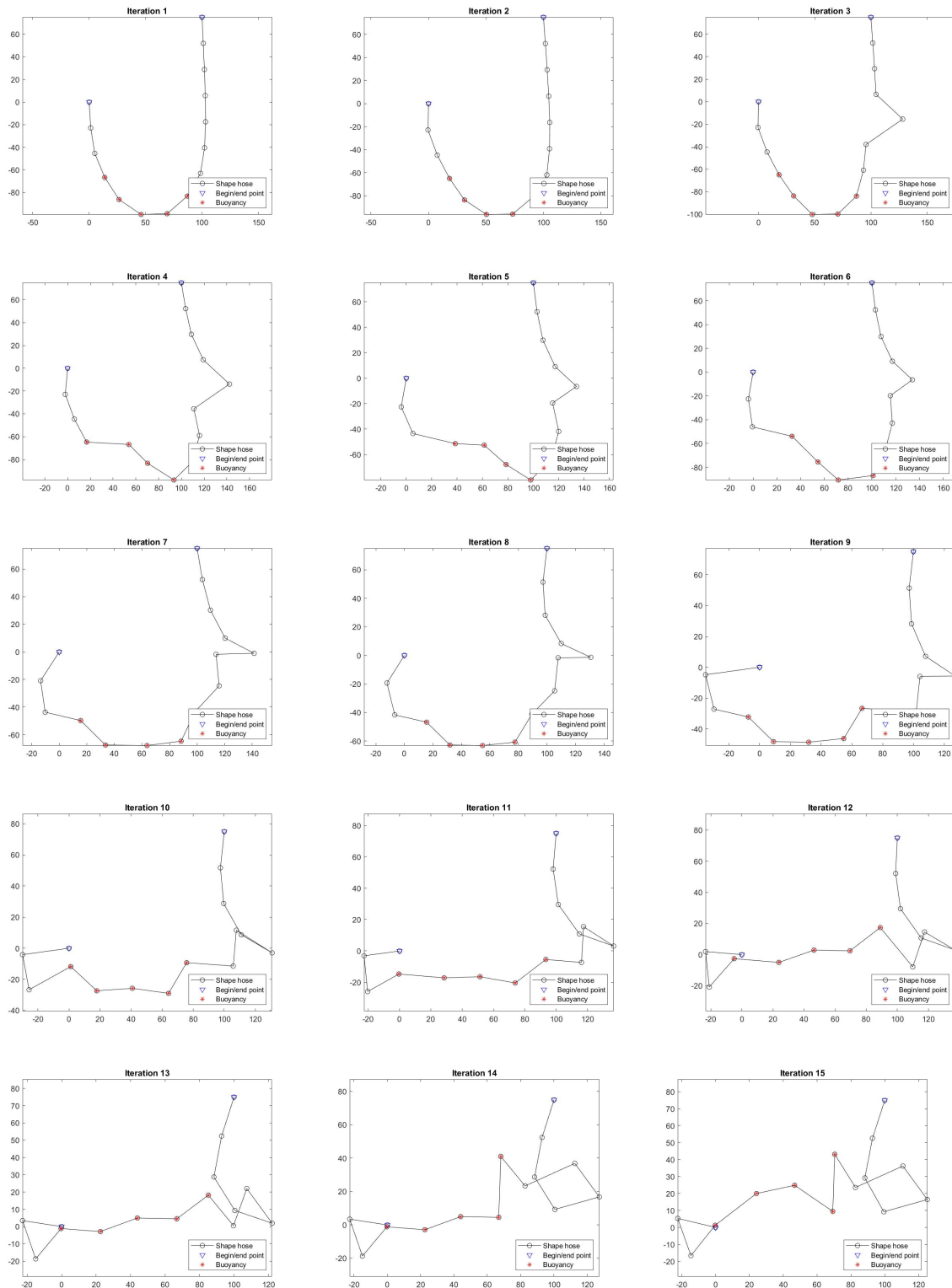
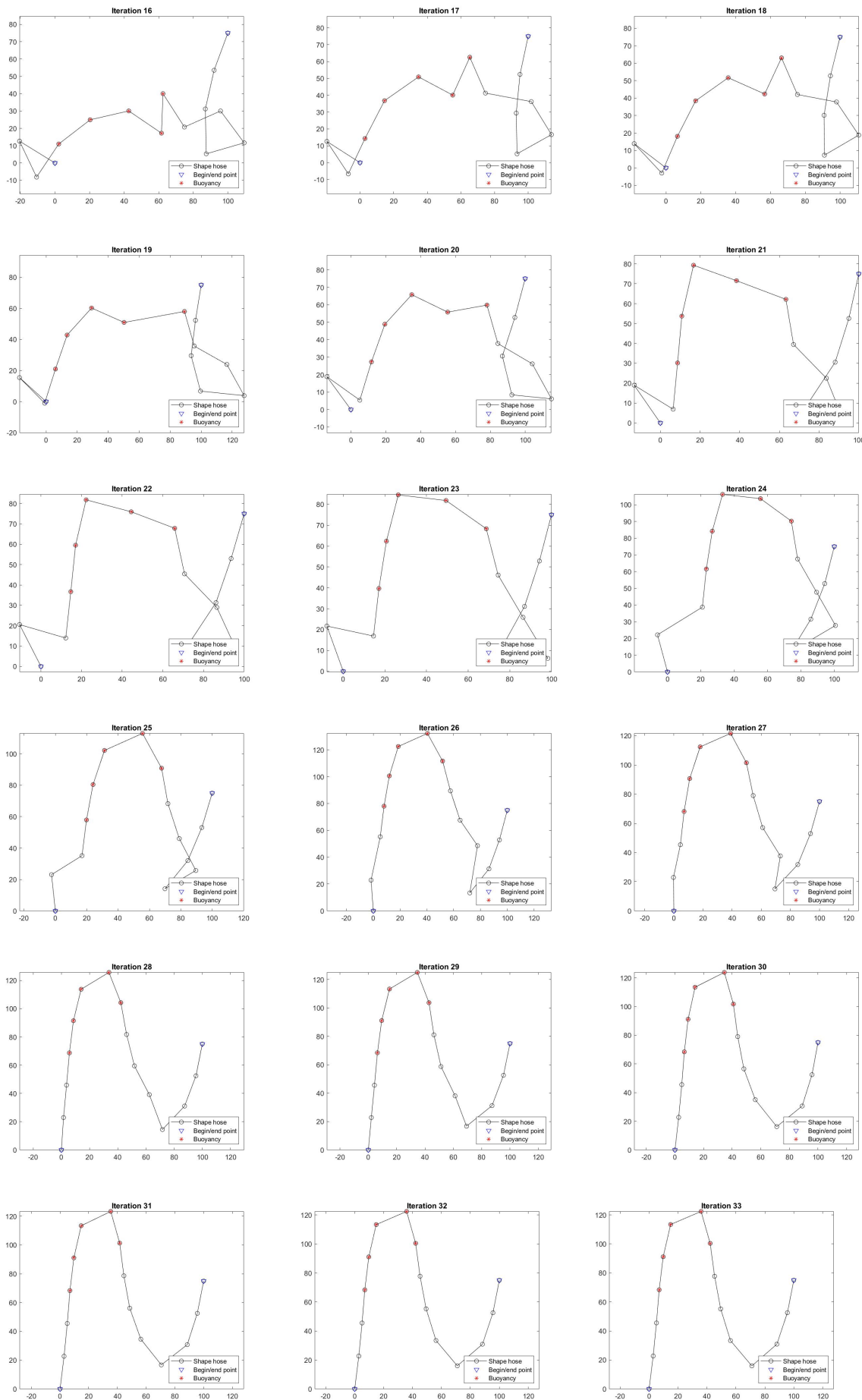


Figure A.6: Iterations s-shape, number 1 to 15





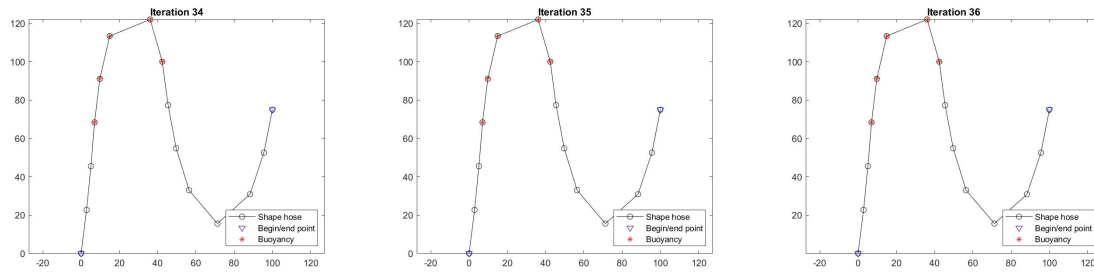


Figure A.8: Iterations s-shape, number 34 to 36

# A.4. Matlab element variation plots

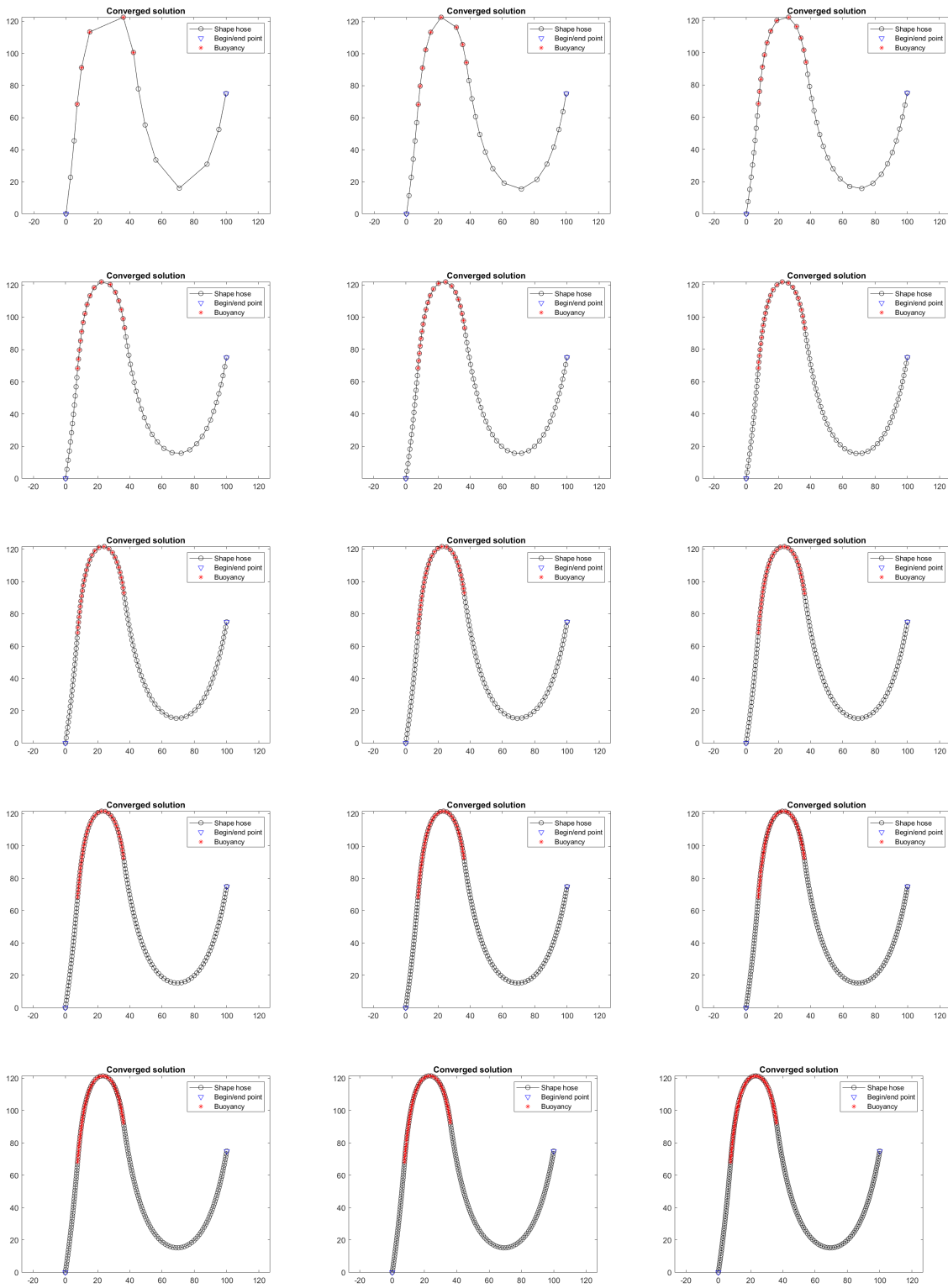


Figure A.9: Plots number of elements per segment 1 to 15

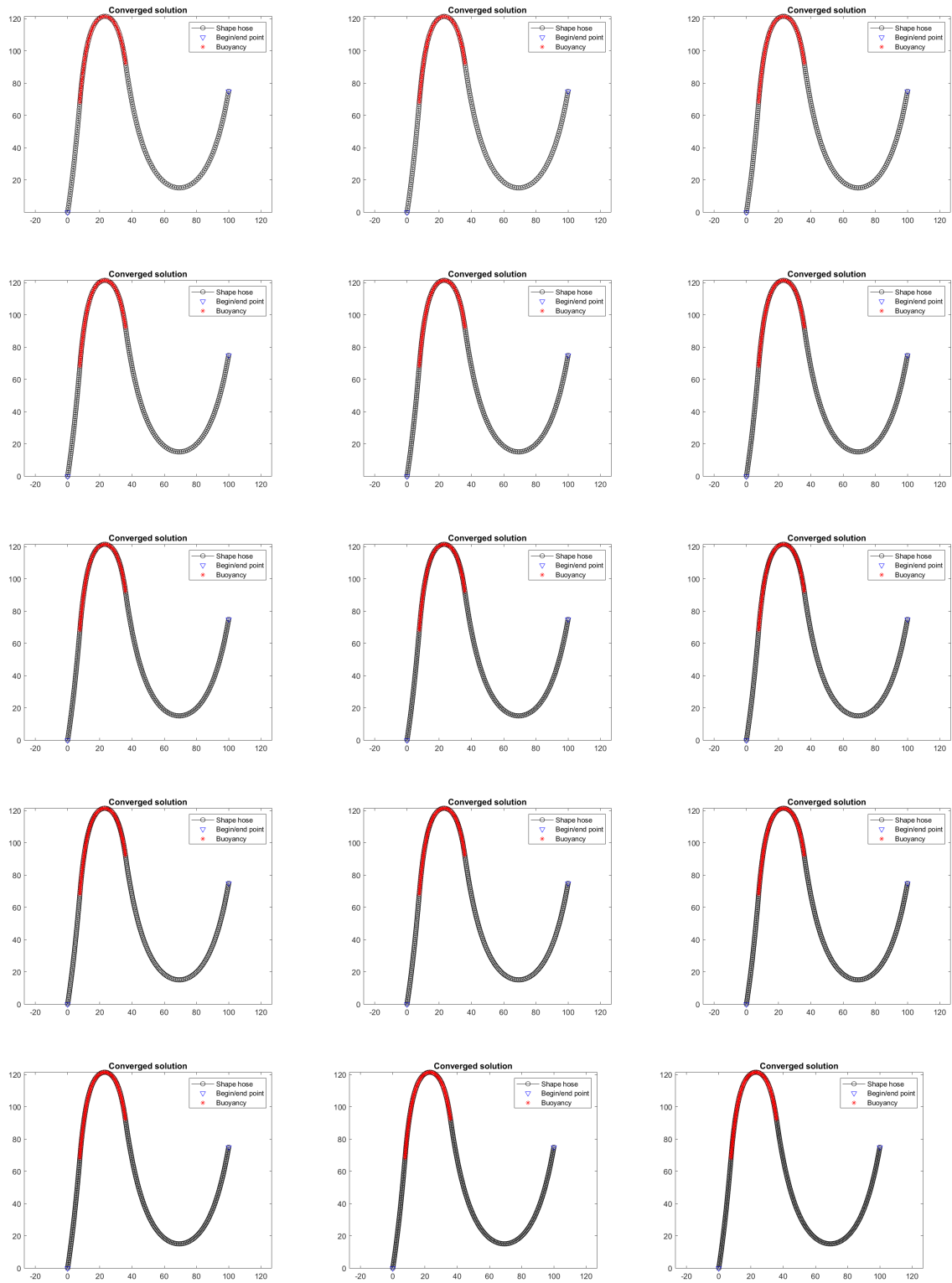


Figure A.10: Plots number of elements per segment 16 to 30

# B

## Jumper hose data

The following table summarizes the initial design parameters specified on the crawler and the umbilical:

Parameter	Value	Unit
Maximum forward velocity		[m/s]
Estimated nodule production (wet density)		[kg/s]

Table B.1: Specific collector initial design parameters

Parameter	Value	Unit
Estimated umbilical submerged mass		[N/m]
Estimated minimum bending radius		[m]
Estimated diameter		[mm]

Table B.2: Specific Umbilical Design Parameters

Parameter	Value	Unit
Estimated internal diameter		[mm]
Estimated slurry velocity		[m/s]
Estimated slurry density		[kg/m <sup>3</sup> ]
Estimated maximum internal over-pressure		[bar]
Estimated maximum internal under-pressure		[bar]
Estimated hose length		[m]
Estimated Single hose length		[m]
Axial tension with crawler in air		[kN]
Axial tension with crawler in water		[kN]

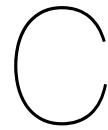
Table B.3: Specific Jumper Assembly Design Parameters

Line part	ID <sup>1</sup> [m]	OD <sup>2</sup> [m]	Length [m]	Weight in air [kg/m]	Weight in seawater filled with seawater [kg/m]	MBR <sup>3</sup> [m]	Axial load ability [kN]	Bending stiffness [kNm]
Hose body standard								
Hose body neck Reinforcement								
Coupling incl. rubber								

Table B.4: Hose parameters in initial design

Parameter	Value	Unit
ID (Inner Diameter)		[mm]
OD (Outer Diameter)		[mm]
Weight in air, hose empty		[kg/m]
Weight in air, hose full of seawater		[kg/m]
Weight in seawater, hose empty		[kg/m]
Weight in seawater, hose full of seawater		kg/m]
Water pressure		[MPa]
Operational MBR (Minimum Bending Radius)		[m]
Bending stiffness		[kNm <sup>2</sup> ]
Axial load ability		[kN]

Table B.5: Parameters Jumper hose design



# MatLab code

## C.1. Base model Matlab Static solution

```
% statische iteratie voor parameters deel 1
% segments hose
clear; clc; close all; dbstop if error
TENSION_ONLY = 1;
%in this document the segments of the hose has x amount of points
%--> example: on hosesegment of 22.86 m, 3 iteration points
%--> for now used 48 modules from document 2.7

%% 1. Define parameters hose given
g = 9.80665; % [m/s^2]
lElement = 22.86 ; % [m] Blue nodules
mass_slurry_kg = 1200; % [kg/m^3]
mass_water_kg = 1025; % [kg/m^3]
mass_hose_kg = 137; % [kg/m]
mass_hose_water_kg = 28.9; % [kg/m]
ID = 300; % [mm]
OD = 473.3; % [mm]
lift_buoyancy = 427; % [kg] max 8 per hose of 22.86 m
EA = 2.0408e7; % [N] uit tabel 6.1

area_hose = pi*((ID*10^-3)/2)^2; % [m^2]
%mass = mass_hose_kg+(mass_slurry_kg-mass_water_kg)*area_hose; % [kg/m] Blue nodules mass h
mass water
%E = EA/(area_hose*10^6); % [MPa=N/mm2] --> *10^6 for N/m^2

% 1.1 Define parameters hose choosen
L_1 = 300; % [m] GUESSED
H = 75; % [m] GUESSED
number_of_modules = 32; % [st]
number_of_modules_per_hose = 8; % [st]
start_segment = 3;
W = 100;

pieces = L_1/lElement; % [st]
nElement = ceil(pieces); % [st] amount of elements needed to construct hose
L = lElement*nElement; % [m]

% 1.2 Create hosesegments, discribed by x points
point_hosesegment = 1;
nElem = nElement*point_hosesegment; % every hose piese is described with 3 points
lElem = L/nElem; % Actual tensionless element size
%m_length = mass*lElem; % [kg] weight hose from point to point
%m = mass; % [kg/m] from point to point
```

```

nNode = nElem + 1; % Number of nodes...
NodeCoord = zeros(nNode,2); % Array that will receive initial position of nodes (te
Element = zeros(nElem, 5); % Array with element properties [NodeLeft NodeRight m EA
% 10 is the tensionless length of the element

% 1.3 Berekening krachten voor massa
volume_hose_OD = lElem*pi()*((OD/1000)/2)^2; %[m3]
volume_hose_ID = lElem*pi()*((ID/1000)/2)^2; %[m3]
buoyancy_force_node = mass_water_kg*volume_hose_OD*g; %[N]
gravity_force_hose = mass_hose_kg*lElem*g; %[N]
gravity_force_slurry = mass_slurry_kg*volume_hose_ID*g;
force_per_node = gravity_force_hose + gravity_force_slurry - buoyancy_force_node; %[N]
m = force_per_node/lElem/g;
mass = m;

%% 2. Loop on elements to create them (and nodes as well)
NodeRow = zeros(1,nElem);
for iElem = 1: 1: nElem
    NodeLeft = iElem;
    NodeRight = NodeLeft + 1;
    NodeCoord(NodeRight,:) = NodeCoord(NodeLeft,:) + [lElem 0]; % make it horizontal
    Element(iElem,:) = [NodeLeft NodeRight m EA lElem]; % assing properties to ele
    NodeRow(1,iElem) = iElem;
end

NodeCoord(:,2) = 1; % Move the nodes horizontally
% (this is just for plotting, not necessary otherwise)

% Plot undeformed wire and position of supports
fig_1 = figure; hold on
f = zeros(2,1);
for iElem = 1: 1: nElem
    Nodes = Element(iElem, [1 2]);
    f(1) = plot(NodeCoord(Nodes,1),NodeCoord(Nodes,2),'-+');
    hold on
end
% Plot also the supports
f(2) = plot([0 W], [0 H], 'vb');
hold off
axis equal
legend([f(1),f(2)], {'construction of string with small string', 'Begin/end point'})

filename_1= strcat('FIG1_SSP1_OVERVIEW_',num2str(number_of_modules),'_BM_',num2str(nu
saveas(fig_1,filename_1,'png')
close

%% 3. Guess initial deformation
nDof = 2*nNode; % Each node can move horizontally and vertically
FreeDof = 1:1:nDof;
FixedDof = [1 2 nDof-1 nDof]; % First node and last node are fixed
FreeDof(FixedDof) = []; % Free dofs (those that are not fixed)

% For the initial configuration, let us assume a parabola of type:
% [s is the coordiante along the undeformed position of the wire]
SAG = 50; % Let us assume a big sag - this will assure that all elements
% are under tension, which may be necessary for the convergence of
% the solver

```



```

% Determine the shape of the parabola
A = [0 0]; B = [H/2 -SAG]; C = [W H]; % Bx was chosen 'random'
Matrix = [A(1) A(2) 1; B(1) B(2) 1; C(1) C(2) 1];
Vector = [A(1)^2; B(1)^2; C(1)^2];
pqr = Matrix\Vector;
p = pqr(1); q = pqr(2); r = pqr(3);
a = 1/q; b = -p*a; c = -r*a;

s = NodeCoord(:,1);
x = W*(s/L);
y = a.*x.^2 + b.*x + c;

u = zeros(nDof,1); % displacement
u(1:2:end) = x-NodeCoord(:,1); % The displacement of the node corresponds to
u(2:2:end) = y-NodeCoord(:,2); % the actual position minus the initial position

% Safe original guess displacement
u_org = zeros(nDof,1);
u_org(1:2:end) = x-NodeCoord(:,1);
u_org(2:2:end) = y-NodeCoord(:,2);

% Plot initial guess
fig_2 = figure;
for iElem = 1: 1: nElem
    Nodes = Element(iElem, [1 2]);
    DofsX = 2*Nodes-1;
    DofsY = 2*Nodes;
    hold on
    plot(NodeCoord(Nodes,1)+u(DofsX),NodeCoord(Nodes,2)+u(DofsY), '-ok');
    hold off
    legend('Initial guess','Location','northwest')
end
filename_2= strcat('FIG2_SSP1_INITIAL_GUESS_',num2str(number_of_modules), '_BM_',num2str(n
saveas(fig_2,filename_2,'png')
close

%% 4. Iteration until convergence
% 4.1. Assemble external load vector - self weight
Pext_g = zeros(nDof,1);
% k_soil = k_soil*lElem;
for iElem = 1: 1: nElem
    Nodes = Element(iElem, [1 2]);
    DofsY = 2*Nodes; % only vertical degrees of freedom matter for weight
    l0 = Element(iElem, 5);
    m = Element(iElem, 3);
    Pext_g(DofsY) = Pext_g(DofsY) - g*m*l0/2; % Half wight to each node
end

% 4.2. Iterate
CONV = 0;
kIter = 0;
nMaxIter = 100;
fig_3 = figure;
while CONV == 0
    kIter = kIter + 1; % update the iteration counter
    fprintf(1, 'Iteration %d ...\n', kIter);
    % Check stabilty - define a number maximum of iterations. If solution

```

```

% hasn't converged, check what is going wrong (if something).
if kIter > nMaxIter, break; end

% 4.2.1. Assemble vector with internal forces and stiffness matrix
K = zeros(nDof,nDof);
Fi = zeros(nDof,1);
Pext = zeros(nDof,1);
for iElem = 1: 1: nElem
    Nodes = Element(iElem, [1 2]); % Nodes of the element
    NodePos = NodeCoord(Nodes,:) + [u(2*Nodes-1) u(2*Nodes)]; % Position of nodes
    Dofs = [2*Nodes(1)-1; 2*Nodes(1); 2*Nodes(2)-1; 2*Nodes(2)];
    [Fi_elem,K_elem,Tension(iElem),WARN] = StringForcesAndStiffness(NodePos,EA,lElem)
    if NodePos(1,2) < 0
        K_elem(2,2) = K_elem(2,2);% + k_soil/2;
    end
    if NodePos(2,2) < 0
        K_elem(4,4) = K_elem(4,4);% + k_soil/2;
        Pext(Dofs(4)) = Pext_g(Dofs(4));% - k_soil*NodePos(2,2);
    else
        Pext(Dofs(4)) = Pext_g(Dofs(4));
    end
    Fi(Dofs) = Fi(Dofs) + Fi_elem;
    K(Dofs,Dofs) = K(Dofs,Dofs) + K_elem;
end

% 4.2.2. Calculate "residual forces"
R = Pext - Fi;

% 4.2.3. Check for convergence
if norm(R(FreeDof))/norm(Pext(FreeDof)) < 1e-4
    CONV = 1;
end

% 4.2.4. Calculate increment of displacements
du = zeros(nDof,1);
du(FreeDof) = K(FreeDof,FreeDof)\R(FreeDof);
if isnan(norm(du))
    du = rand(size(du))/lElem/100;
end

% 4.2.5. Apply arclength to help with convergence
Scale = min([1 lElem/max(abs(du))]);
du = du*Scale; % Enforce that each node does not displace
                % more (at each iteration) than the length
                % of the elements

% 4.2.6. Update displacement of nodes
u = u + du;
% 4.2.7. Plot updated configuration
if 1 %kIter/10 == round(kIter/10)
    plot(NodeCoord(:,1) + u(1:2:end-1), NodeCoord(:,2) + u(2:2:end), '-
ok')
    hold on

    plot([A(1) C(1)], [A(2) C(2)], 'vr');
    axis equal
    title(['Iteration ' num2str(kIter)]);
    drawnow;
    hold off;

```

```

    % Display convergence
    Rmax = max(abs(R(FreeDof)));
    disp(Rmax)
    disp(norm(R(FreeDof))/norm(Pext(FreeDof)))
    drawnow;
    hold off;
    legend('Iterated solution','Location','northwest')

    %% save iterations
    filename_2= strcat('FIG_CAT_IT',num2str(kIter));
    saveas(gcf,filename_2,'jpeg')
end
end
if CONV
    title('Converged solution');
else
    title('Solution did not converge.');
end
filename_3= strcat('FIG3_SSP1_CATENARY_',num2str(number_of_modules), '_BM_',num2str(number_of_modules));
saveas(fig_3,filename_3,'png')
close

% 4.3 Table formulation, table bij figure
coordinate_org = zeros(nDof,1);
coordinate_org(1:2:end)= NodeCoord(:,1) + u_org(1:2:end-1);
coordinate_org(2:2:end)= NodeCoord(:,2) + u_org(2:2:end);
Pext_eigengewicht = Pext_g;

coordinate_cat = zeros(nDof,1);
coordinate_cat(1:2:end)= NodeCoord(:,1) + u(1:2:end-1);
coordinate_cat(2:2:end)= NodeCoord(:,2) + u(2:2:end);

%% 5. plot 1st and second iteration
fig_4 = figure;
for iElem = 1: 1: nElem
    Nodes = Element(iElem, [1 2]);
    DofsX = 2*Nodes-1;
    DofsY = 2*Nodes;
    hold on
    plot(NodeCoord(Nodes,1)+u_org(DofsX),NodeCoord(Nodes,2)+u_org(DofsY), '-.b');
    hold on
    plot(NodeCoord(:,1) + u(1:2:end-1), NodeCoord(:,2) + u(2:2:end), '-.k')
    hold off
    legend({'Initial guess','Iterated solution'},'Location','northwest')
end
filename_4= strcat('FIG4_SSP1_GUESS_VS_CATENARY_',num2str(number_of_modules), '_BM_',num2str(number_of_modules));
saveas(fig_4,filename_4,'png')
close

% Plot Tension parabol
fig_7 = figure;
plot(Tension, '-*')
grid on
grid minor

```

```

title(['Tension Catenary System']);
xlabel('Node number')
ylabel('Tension [N]')
filename_7= strcat('FIG1_TENSION_max_it_Parabol_',num2str(point_hosesegment),'_points
%saveas(fig_7,filename_7,'png')

Tension_catenary = Tension;
R_catenary = R;

%new u
u_it = u;
close

%% 6. Loads on system
% 6.1 Assemble external load vector
% self weight & buoyancy, overschrijf de Pext_g
Pext_g = zeros(nDof,1);
for iElem = 1: 1: nElem
    Nodes = Element(iElem, [1 2]);
    DofsY = 2*Nodes; % only vertical degrees of freedom matter for weight
    l0 = Element(iElem, 5);
    m = Element(iElem, 3);
    Pext_g(DofsY) = Pext_g(DofsY) - g*l0*m/2; % Half weight to each node m/s2*m*kg/m= N
end

% Buoyancy modules
number_of_buoyant_hoses = (number_of_modules/number_of_modules_per_hose); %[st]
number_of_buoyant_hoses_element = number_of_buoyant_hoses*point_hosesegment; %number_of

% Force due to buoyancy
buoyancy_force_per_module = lift_buoyancy*g; %[N]
buoyancy_force_total = (lift_buoyancy*g)*number_of_modules; %[N]

% location buoyancy on even number (y coordinate)
buoyancy_coordinate_begin = 2 + start_segment*point_hosesegment*2; %round_even(nDof/6)
buoyancy_plot_begin = buoyancy_coordinate_begin/2; %plot location of plot_node_y

if mod(point_hosesegment,2) == 1 %odd
    if mod(number_of_buoyant_hoses,1)==0.5
        buoyancy_nodes_element = floor(number_of_buoyant_hoses_element)+1;
        buoyancy_nodes_element_add = floor(number_of_buoyant_hoses_element)*2; %y coordinate
        buoyancy_nodes_element_end = buoyancy_coordinate_begin + buoyancy_nodes_element_add;
    else
        buoyancy_nodes_element = round(number_of_buoyant_hoses_element)+1; % +1 voor nodes in
        buoyancy_nodes_element_add = round(number_of_buoyant_hoses_element)*2; %y coordinate
        buoyancy_nodes_element_end = buoyancy_coordinate_begin + buoyancy_nodes_element_add;
    end
elseif mod(point_hosesegment,2) == 0 % even segmenten per hose
    if mod(number_of_buoyant_hoses,1)==0.5
        buoyancy_nodes_element = floor(number_of_buoyant_hoses_element)+1;
        buoyancy_nodes_element_add = floor(number_of_buoyant_hoses_element)*2; %y coordinate
        buoyancy_nodes_element_end = buoyancy_coordinate_begin + buoyancy_nodes_element_add;
    else
        buoyancy_nodes_element = round(number_of_buoyant_hoses_element)+1; % +1 voor nodes in
        buoyancy_nodes_element_add = round(number_of_buoyant_hoses_element)*2; %y coordinate
        buoyancy_nodes_element_end = buoyancy_coordinate_begin + buoyancy_nodes_element_add;
    end
end

```

```

end
buoyancy_plot_end = buoyancy_nodes_element_end/2; %plot location of plot_node_y
buoyancy_nodes_midden = (round(buoyancy_nodes_element/2) + start_segment)*2;

% buoyancy force - selfweight hose
%distributed_buoyancy_force = buoyancy_force_total/buoyancy_nodes_element;
Pext_g(buoyancy_nodes_midden) = buoyancy_force_total - g*10*m; % maximaal belade gedeelte incl
Pext_g_buoy = zeros(nDof,1);
Pext_g_buoy(buoyancy_nodes_midden) = buoyancy_force_total; % maximaal belade gedeelte exclus

%% 7. Iteration until convergence buoyancy
CONV = 0;
kIter = 0;
nMaxIter = 100000;
fig_5 = figure;
while CONV == 0
    kIter = kIter + 1; % update the iteration counter
    fprintf(1, 'Iteration %d ...\n', kIter);
    % Check stability - define a number maximum of iterations. If solution
    % hasn't converged, check what is going wrong (if something).
    if kIter > nMaxIter, break; end

    % 7.1.1. Assemble vector with internal forces and stiffness matrix
    K = zeros(nDof,nDof);
    Fi = zeros(nDof,1);
    Pext = zeros(nDof,1);
    for iElem = 1: 1: nElem
        Nodes = Element(iElem, [1 2]); % Nodes of the element
        NodePos = NodeCoord(Nodes,:) + [u_it(2*Nodes-1) u_it(2*Nodes)]; % Position of nodes
        Dofs = [2*Nodes(1)-1; 2*Nodes(1); 2*Nodes(2)-1; 2*Nodes(2)];
        [Fi_elem,K_elem,Tension(iElem),WARN] = StringForcesAndStiffness(NodePos,EA,lElem,TENS
        if NodePos(1,2) < 0
            K_elem(2,2) = K_elem(2,2);% + k_soil/2;
        end
        if NodePos(2,2) < 0
            K_elem(4,4) = K_elem(4,4);% + k_soil/2;
            Pext(Dofs(4)) = Pext_g(Dofs(4));% - k_soil*NodePos(2,2);
        else
            Pext(Dofs(4)) = Pext_g(Dofs(4));
        end
        Fi(Dofs) = Fi(Dofs) + Fi_elem;
        K(Dofs,Dofs) = K(Dofs,Dofs) + K_elem;
    end

    % 7.1.2. Calculate "residual forces"
    R = Pext - Fi;

    % 7.1.3. Check for convergence
    if norm(R(FreeDof))/norm(Pext(FreeDof)) < 1e-4
        CONV = 1;
    end

    % 7.1.4. Calculate increment of displacements
    du = zeros(nDof,1);
    du(FreeDof) = K(FreeDof,FreeDof)\R(FreeDof);
    if isnan(norm(du))
        du = rand(size(du))/lElem/100;
    end
end

```

```

% 7.1.5. Apply arclength to help with convergence
Scale = min([1 lElem/max(abs(du))]);
du = du*Scale; % Enforce that each node does not displace
                % more (at each iteration) than the lenght
                % of the elements

% 7.1.6. Update displacement of nodes
u_it = u_it + du;
plot_node_x = (NodeCoord(:,1) + u_it(1:2:end-1));
plot_node_y = (NodeCoord(:,2) + u_it(2:2:end));
% 7.1.7. Plot updated configuration
if 1 %kIter/10 == round(kIter/10)
    plot(NodeCoord(:,1) + u_it(1:2:end-1), NodeCoord(:,2) + u_it(2:2:end), '-
ok')
        hold on

        plot([A(1) C(1)], [A(2) C(2)], 'vb');
        % plot buoyancy locations
        plot(plot_node_x(buoyancy_nodes_midden/2), plot_node_y(buoyancy_nodes_midden/2),

        axis equal
        title(['Iteration ' num2str(kIter)]);
        drawnow;
        hold off;
        legend({'Shape hose', 'Begin/end point', 'Buoyancy'}, 'Location', 'southeast')

        % % save iterations
        filename_2a= strcat('FIG_SSHAPE_1NODE_IT', num2str(kIter));
        saveas(gcf, filename_2a, 'jpeg')

        % Display convergence
        Rmax = max(abs(R(FreeDof)));
        disp(Rmax);
        disp(norm(R(FreeDof))/norm(Pext(FreeDof)))
        drawnow;
        hold off;
    end
end
if CONV
    title('Converged solution');
else
    title('Solution did not converge. ');
end

filename_5= strcat('FIG5_SSP1_BUOYANCY_S_SHAPE_', num2str(number_of_modules), '_BM_', num2str(kIter));
saveas(fig_5, filename_5, 'png')
close

%% Plot Tension parabol
Tension_sshape = Tension;

fig_8 = figure;
plot(Tension_catenary, '-*')
hold on
plot(Tension_sshape, '-*')
hold on
plot(NodeRow(buoyancy_nodes_midden/2), Tension_sshape(buoyancy_nodes_midden/2), 'ok');

```

```

hold off

grid on
grid minor
title('Tension Catenary and Buoyant System')
xlabel('Node number')
ylabel('Tension [N]')
legend('Tension Catenary System', 'Tension Buoyant System', 'Buoyant Nodes')

filename_8= strcat('FIG3_TENSION_max_it_SSHAPE_', num2str(point_hosesegment), '_points_per_1
saveas(fig_8,filename_8,'png')

% 7.2. Table formulation, table bij figure
coordinate_sshape = zeros(nDof,1);
coordinate_sshape(1:2:end)= NodeCoord(:,1) + u_it(1:2:end-1);
coordinate_sshape(2:2:end)= NodeCoord(:,2) + u_it(2:2:end);

%% 8. Calculation angle between points
%plot_node_x and plot_node_y are coordinates points
position_vector = [plot_node_x,plot_node_y];
position_x = zeros(nNode,1);
position_y = zeros(nNode,1);
alpha_e = zeros(nNode,1);
alpha_e_1 = zeros(nNode,1);
alpha_ee = zeros(nNode,1);
alpha_ee_1 = zeros(nNode,1);
alpha_check = zeros(nNode,1);

% 8.1 Calculate angles between hose piece and earth
for i= 2:1:nNode
position_x(i)= plot_node_x(i)-plot_node_x(i-1); % same as coordinate_sshape
position_y(i)= plot_node_y(i)-plot_node_y(i-1); % same as coordinate_sshape
end

for i=1:1:nNode-1
alpha_e_1(i) = position_y(i+1)/position_x(i+1);
alpha_e(i) = atand(alpha_e_1(i));
alpha_ee_1(i) = position_x(i+1)/position_y(i+1);
alpha_ee(i) = atand(alpha_ee_1(i));
alpha_check(i) = alpha_e(i)+alpha_ee(i);
end

% 8.2 Calculate angle between hose i and i+1, sign iff statement
beta_under = zeros(nNode,1);
for i = 2:1:nNode
    beta_under(1) = alpha_e(1) +90 ;
    if    alpha_ee(i-1)>=0& alpha_e(i)>=0;
        beta_under(i)= alpha_ee(i-1) + 90 + alpha_e(i);
    elseif    alpha_ee(i-1)>=0& alpha_e(i)<0;
        beta_under(i)= alpha_ee(i-1) + 90 + alpha_e(i);
    elseif    alpha_ee(i-1)<0& alpha_e(i)<0;
        beta_under(i)= alpha_ee(i-1) - 90 + alpha_e(i);
    elseif    alpha_ee(i-1)<0& alpha_e(i)>0;
        beta_under(i)=360+(alpha_ee(i-1)-90 + alpha_e(i));
    end
    beta_under(nNode) = alpha_ee(nNode-1);
end

```

```

beta_upside = zeros(nNode,1);
for i = 2:1:nNode
    beta_upside(1) = alpha_ee(1);
    if alpha_e(i-1)>=0 & alpha_ee(i)>=0;
        beta_upside(i) = alpha_e(i-1) + 90 + alpha_ee(i);
    elseif alpha_e(i-1)>=0 & alpha_ee(i)<0;
        beta_upside(i) = alpha_e(i-1) + 180 + (90 + alpha_ee(i));
    elseif alpha_e(i-1)<0 & alpha_ee(i)<0;
        beta_upside(i) = alpha_e(i-1) - 90 + alpha_ee(i);
    elseif alpha_e(i-1)<0 & alpha_ee(i)>0;
        beta_upside(i) = alpha_e(i-1) +90 + alpha_ee(i);
    end
    beta_upside(nNode) = alpha_e(nNode-1)+90;
end
beta = beta_under + beta_upside;

close all

```

## C.2. Matlab code parameter study

```

% statische iteratie voor parameters deel 2
clear; clc; close all; dbstop if error
TENSION_ONLY = 1;
%in this document the segments of the hose has x amount of points
%--> example: on hosesegment of 22.86 m, 3 iteration points
%--> for now used 48 modules from document 2.7

```

```

tic
% 1. Define parameters hose given
g = 9.80665; % [m/s^2]
lElement = 22.86 ; % [m] SEGMENT
mass_slurry_kg = 1200; % [kg/m]
mass_water_kg = 1025; % [kg/m]
mass_hose_kg = 137; % [kg/m]
mass_hose_water_kg = 28.9; % [kg/m]
ID = 300; % [mm]
OD = 473.3; % [mm]
lift_buoyancy = 427; % [kg] max 8 per hose of 22.86 m
EA = 2.0408e7; % [N] uit tabel 6.1
point_hosesegment = 5;

```

```

area_hose = pi*((ID*10^-3)/2)^2; % [m^2]
area_hose_out = pi*((OD*10^-3)/2)^2; % [m^2]
E = EA/(area_hose*10^6); % [MPa=N/mm2] --> *10^6 for N/m^2

```

```

% 1.4 Force in air

```

```

lElem = lElement/point_hosesegment;
volume_hose_OD = lElem*pi()*((OD/1000)/2)^2; % [m3]
volume_hose_ID = lElem*pi()*((ID/1000)/2)^2; % [m3]
buoyancy_force_node = mass_water_kg*volume_hose_OD*g; % [N]
gravity_force_hose = mass_hose_kg*lElem*g; % [N]
gravity_force_slurry = mass_slurry_kg*volume_hose_ID*g;
gravity_force_water = mass_water_kg*volume_hose_ID*g;
force_slurry_per_node = gravity_force_hose + gravity_force_slurry - buoyancy_force_node;
force_water_per_node = gravity_force_hose + gravity_force_water - buoyancy_force_node;
mass_slurry = force_slurry_per_node/lElem/g;
mass_water = force_water_per_node/lElem/g;

```



```

% 1.0.1 Variation matrix
L_11 = 300; L_12 = 350; L_13 = 400; L_14 = 450;
L_1_var = [L_11; L_12; L_13; L_14];
H_var = [75; 100; 125];
num_mod_300 = [20:1:31]; num_mod_350 = [23:1:36];
num_mod_400 = [26:1:40]; num_mod_450 = [29:1:44];
number_of_modules_per_hose_var = [3; 4; 5; 6; 7; 8];
start_seg_300 = [1:1:3]; start_seg_350 = [1:1:3];
start_seg_400 = [1:1:3]; start_seg_450 = [1:1:4];
W_var = [50; 100; 150; 200];
mass_var = [mass_slurry ;mass_water];

for n = 1:1:length(L_1_var)
    save_pieces = L_1_var(n)/lElement;           %[st]
    save_nElement(n) = ceil(save_pieces);        %[st] amount of elements needed to construct hose
    save_L(n) = lElement*save_nElement(n);      %[m]
    save_nElem(n) = save_nElement(n)*point_hosesegment;
    save_L_node(n) = lElement/point_hosesegment;
end

% Opbouwen combinatie vector
[aa, ab, ac, ad, ae, af, ag] = ndgrid(W_var, mass_var, start_seg_300, number_of_modules_per_hose_var);
comb_matrix1 = [aa(:), ab(:), ac(:), ad(:), ae(:), af(:), ag(:)];
[ba, bb, bc, bd, be, bf, bg] = ndgrid(W_var, mass_var, start_seg_350, number_of_modules_per_hose_var);
comb_matrix2 = [ba(:), bb(:), bc(:), bd(:), be(:), bf(:), bg(:)];
[ca, cb, cc, cd, ce, cf, cg] = ndgrid(W_var, mass_var, start_seg_400, number_of_modules_per_hose_var);
comb_matrix3 = [ca(:), cb(:), cc(:), cd(:), ce(:), cf(:), cg(:)];
[da, db, dc, dd, de, df, dg] = ndgrid(W_var, mass_var, start_seg_450, number_of_modules_per_hose_var);
comb_matrix4 = [da(:), db(:), dc(:), dd(:), de(:), df(:), dg(:)];
comb_matrix = [comb_matrix1; comb_matrix2; comb_matrix3; comb_matrix4];

% opslaan basis info
table_0(1,1) = g; %[m/s2]
table_0(2,1) = mass_slurry_kg; table_0(2,2) = mass_water_kg; table_0(2,3) = mass_hose_kg;
table_0(3,1) = lElement; %[m]
table_0(4,1) = ID; table_0(4,2) = OD; %[mm]
table_0(5,1) = area_hose; table_0(5,2) = area_hose_out; %[m^2]
table_0(6,1) = volume_hose_ID; table_0(6,2) = volume_hose_OD;
table_0(7,1) = lift_buoyancy;%[kg]
table_0(8,1) = EA; table_0(8,2) = E; %[N] & [MPa=N/mm2]
table_0(9,1) = buoyancy_force_node; table_0(9,2) = gravity_force_hose;
table_0(9,3) = gravity_force_slurry ; table_0(9,4) = gravity_force_water;
table_0(10,1) = force_slurry_per_node; table_0(10,2) = force_water_per_node;
table_0(11,1) = mass_slurry; table_0(11,1) = mass_water;

% opslaan parameter variaties
table_00(1,1:length(L_1_var)) = L_1_var;           %[m] gekozen lengte
table_00(2,1:length(save_L)) = save_L;           %[m] afgeronde lengte
table_00(3,1:length(H_var)) = H_var;             %[m] hoogte VTS aansluiting
table_00(4,1:length(num_mod_300)) = num_mod_300; %[st] total number modules
table_00(5,1:length(num_mod_350)) = num_mod_350; %[st] total number modules
table_00(6,1:length(num_mod_400)) = num_mod_400; %[st] total number modules
table_00(7,1:length(num_mod_450)) = num_mod_450; %[st] total number modules
table_00(8,1:length(number_of_modules_per_hose_var)) = number_of_modules_per_hose_var; %[st]
table_00(9,1:length(start_seg_300)) = start_seg_300; %[No] start hose buoyancy
table_00(10,1:length(start_seg_350)) = start_seg_350; %[No] start hose buoyancy

```

```

table_00(11,1:length(start_seg_400)) = start_seg_400; %[No] start hose buoyancy
table_00(12,1:length(start_seg_450)) = start_seg_450; %[No] start hose buoyancy
table_00(13,1:length(W_var)) = W_var; %[m] crawler tot VTS
table_00(14,1:length(mass_var)) = mass_var; %[kg/m] slurry of water

table_000(1,1:length(L_1_var)) = L_1_var; %[m] gekozen length
table_000(2,1:length(save_nElement)) = save_nElement; % number of hoses per length
table_000(3,1:length(save_nElem)) = save_nElem; % number of elements total
table_000(4,1:length(save_L_node)) = save_L_node; % length element

% Combinatie matrix
table_0000 = comb_matrix;

%% 1.1 Varing parameters
for k = 1:1:length(comb_matrix)
    % variabele
    L_1 = comb_matrix(k,7);
    H = comb_matrix(k,6);
    number_of_modules = comb_matrix(k,5);
    number_of_modules_per_hose = comb_matrix(k,4);
    start_segment = comb_matrix(k,3);
    mass = comb_matrix(k,2);
    W = comb_matrix(k,1);

    if mass == mass_water
        mass_name = mass_water_kg;
    else
        mass_name = mass_slurry_kg;
    end
end

%% 1.2 Define parameters hose, for choosen variables

pieces = L_1/lElement; %[st]
nElement = ceil(pieces); %[st] amount of elements needed to construct hose
L = lElement*nElement; %[m]

% 1.3 Opbouw hose
nElem = nElement*point_hosesegment; % every hose piese is described with 3 points
lElem = L/nElem; % ELEMENT % Actual tensionless element size -
-> not 22.86 m
% m_length = mass*lElem; %[kg] from point to point
m = mass; %[kg/m]
nNode = nElem + 1; % Number of nodes...
NodeCoord = zeros(nNode,2); % Array that will receive initial position of nodes (
Element = zeros(nElem, 5); % Array with element properties [NodeLeft NodeRight m EA lElem]
% 10 is the tensionless length of the element

%% 2. Loop on elements to create them (and nodes as well)
NodeRow = zeros(1,nElem);
for iElem = 1: 1: nElem
    NodeLeft = iElem;
    NodeRight = NodeLeft + 1;
    NodeCoord(NodeRight,:) = NodeCoord(NodeLeft,:) + [lElem 0]; % make it horizontal
    Element(iElem,:) = [NodeLeft NodeRight m EA lElem]; % assing properties to ele
    NodeRow(1,iElem) = iElem;
end

```

```

NodeCoord(:,2) = 1;           % Move the nodes horizontally
                             % (this is just for plotting, not necessary otherwise)

%% 3. Guess initial deformation
nDof = 2*nNode;              % Each node can move horizontally and vertically
FreeDof = 1:1:nDof;
FixedDof = [1 2 nDof-1 nDof]; % First node and last node are fixed
FreeDof(FixedDof) = [];      % Free dofs (those that are not fixed)
% check what happens if free, check if R can be used as force on crawler

% For the initial configuration, let us assume a parabola of type:
% [s is the coordinate along the undeformed position of the wire]
SAG = 75; % Let us assume a big sag - this will assure that all elements
          % are under tension, which may be necessary for the convergence of
          % the solver

% Determine the shape of the parabola
A = [0 0]; B = [30 -SAG]; C = [W H]; % Bx was chosen 'random'
Matrix = [A(1) A(2) 1; B(1) B(2) 1; C(1) C(2) 1];
Vector = [A(1)^2; B(1)^2; C(1)^2];
pqr = Matrix\Vector;
p = pqr(1); q = pqr(2); r = pqr(3);
a = 1/q; b = -p*a; c = -r*a;

s = NodeCoord(:,1);
x = W*(s/L);
y = a.*x.^2 + b.*x + c;

u = zeros(nDof,1);          % displacement
u(1:2:end) = x-NodeCoord(:,1); % The displacement of the node corresponds to
u(2:2:end) = y-NodeCoord(:,2); % the actual position minus the initial position

%% 4. Iteration until convergence
% 4.1. Assemble external load vector - self weight
Pext_g = zeros(nDof,1);
% k_soil = k_soil*lElem;
for iElem = 1: 1: nElem
    Nodes = Element(iElem, [1 2]);
    DofsY = 2*Nodes; % only vertical degrees of freedom matter for weight
    l0 = Element(iElem, 5);
    m = Element(iElem, 3);
    Pext_g(DofsY) = Pext_g(DofsY) - g*l0*m/2; % Half weight to each node
end

% 4.2. Iterate
CONV = 0;
kIter = 0;
nMaxIter = 1000;
fig_1 = figure;
while CONV == 0
    kIter = kIter + 1; % update the iteration counter
    fprintf(1, 'Iteration %d ...\n', kIter);
    % Check stability - define a number maximum of iterations. If solution
    % hasn't converged, check what is going wrong (if something).
    if kIter > nMaxIter, break; end

    % 4.2.1. Assemble vector with internal forces and stiffness matrix
    K = zeros(nDof,nDof);

```

```

Fi = zeros(nDof,1);
Pext = zeros(nDof,1);
for iElem = 1: 1: nElem
    Nodes = Element(iElem, [1 2]); % Nodes of the element
    NodePos = NodeCoord(Nodes, :) + [u(2*Nodes-1) u(2*Nodes)]; % Position of nodes
    Dofs = [2*Nodes(1)-1; 2*Nodes(1); 2*Nodes(2)-1; 2*Nodes(2)];
    [Fi_elem, K_elem, Tension(iElem), WARN] = StringForcesAndStiffness(NodePos, EA, lElem);
    if NodePos(1,2) < 0
        K_elem(2,2) = K_elem(2,2); % + k_soil/2;
    end
    if NodePos(2,2) < 0
        K_elem(4,4) = K_elem(4,4); % + k_soil/2;
        Pext(Dofs(4)) = Pext_g(Dofs(4)); % - k_soil*NodePos(2,2);
    else
        Pext(Dofs(4)) = Pext_g(Dofs(4));
    end
    Fi(Dofs) = Fi(Dofs) + Fi_elem;
    K(Dofs,Dofs) = K(Dofs,Dofs) + K_elem;
end

% 4.2.2. Calculate "residual forces"
R = Pext - Fi;

% 4.2.3. Check for convergence
if norm(R(FreeDof))/norm(Pext(FreeDof)) < 1e-4
    CONV = 1;
end

% 4.2.4. Calculate increment of displacements
du = zeros(nDof,1);
du(FreeDof) = K(FreeDof,FreeDof)\R(FreeDof);
if isnan(norm(du))
    du = rand(size(du))/lElem/100;
end

% 4.2.5. Apply arclength to help with convergence
Scale = min([1 lElem/max(abs(du))]);
du = du*Scale; % Enforce that each node does not displace
                % more (at each iteration) than the lenght
                % of the elements

% 4.2.6. Update displacement of nodes
u = u + du;
plot_node_x_cat = (NodeCoord(:,1) + u(1:2:end-1));
plot_node_y_cat = (NodeCoord(:,2) + u(2:2:end));
% 4.2.7. Plot updated configuration
% if 1 %kIter/10 == round(kIter/10)
% plot(NodeCoord(:,1) + u(1:2:end-1), NodeCoord(:,2) + u(2:2:end), '-
ok')
% hold on
%
% plot([A(1) C(1)], [A(2) C(2)], 'vr');
% axis equal
% title(['Iteration ' num2str(kIter)]);
% drawnow;
% hold off;
%
% % Display convergence
% Rmax = max(abs(R(FreeDof)));

```

```

% disp(Rmax)
% disp(norm(R(FreeDof))/norm(Pext(FreeDof)))
% drawnow;
% hold off;
% legend('Iterated solution','Location','northwest')
% end
end
% if CONV
% title('Converged solution');
% else
% title('Solution did not converge. ');
% end

%% 5. Plot Tension parabol, catenary system
% fig_2 = figure;
% plot(Tension, '-*')
% grid on
% grid minor

% title(['Tension Catenary System']);
% xlabel('Node number')
% ylabel('Tension [N]')

% save eigengewicht
Pext_eigengewicht = Pext_g;
Tension_catenary = Tension;
R_catenary = R;

% save coordinates catenary
coordinate_cat_1 = [plot_node_x_cat(1:1:nNode/2),plot_node_y_cat(1:1:nNode/2)];
coordinate_cat_min_y_1 = min(coordinate_cat_1(:,2));
coordinate_cat_y_1b = find(plot_node_y_cat == coordinate_cat_min_y_1);
coordinate_cat_x_1b = plot_node_x_cat(coordinate_cat_y_1b);

% new u
u_it = u;

%% 6. Loads on system
% 6.1 Assemble external load vector
% self weight & buoyancy, overschrijf de Pext_g
Pext_g = zeros(nDof,1);
for iElem = 1: 1: nElem
    Nodes = Element(iElem, [1 2]);
    DofsY = 2*Nodes; % only vertical degrees of freedom matter for weight
    l0 = Element(iElem, 5);
    m = Element(iElem, 3);
    Pext_g(DofsY) = Pext_g(DofsY) - g*l0*m/2; % Half wight to each node
end

% Buoyancy modules
number_of_buoyant_hoses = (number_of_modules/number_of_modules_per_hose); %[st]
number_of_buoyant_hoses_element = number_of_buoyant_hoses*point_hosesegment; %number_of_mo

% Force due to buoyancy
buoyancy_force_per_module = lift_buoyancy*g; %[N]
buoyancy_force_total = (lift_buoyancy*g)*number_of_modules; %[N]

```

```

% location buoyancy on even number (y coordinate)
buoyancy_coordinate_begin = 2 + start_segment*point_hoseselement*2; %round_even(nDof/6)
buoyancy_plot_begin = buoyancy_coordinate_begin/2; %plot location of plot_node_y

if mod(point_hoseselement,2) == 1 %odd
    if mod(number_of_buoyant_hoses,1)==0.5
        buoyancy_nodes_element = floor(number_of_buoyant_hoses_element)+1;
        buoyancy_nodes_element_add = floor(number_of_buoyant_hoses_element)*2; %y coordinate
        buoyancy_end_node_element = buoyancy_coordinate_begin + buoyancy_nodes_element_add;
    else
        buoyancy_nodes_element = round(number_of_buoyant_hoses_element)+1; % +1 voor nodes in
        buoyancy_nodes_element_add = round(number_of_buoyant_hoses_element)*2; %y coordinate
        buoyancy_end_node_element = buoyancy_coordinate_begin + buoyancy_nodes_element_add;
    end
elseif mod(point_hoseselement,2) == 0 % even segmenten per hose
    if mod(number_of_buoyant_hoses,1)==0.5
        buoyancy_nodes_element = floor(number_of_buoyant_hoses_element)+1;
        buoyancy_nodes_element_add = floor(number_of_buoyant_hoses_element)*2; %y coordinate
        buoyancy_end_node_element = buoyancy_coordinate_begin + buoyancy_nodes_element_add;
    else
        buoyancy_nodes_element = round(number_of_buoyant_hoses_element)+1; % +1 voor nodes in
        buoyancy_nodes_element_add = round(number_of_buoyant_hoses_element)*2; %y coordinate
        buoyancy_end_node_element = buoyancy_coordinate_begin + buoyancy_nodes_element_add;
    end
end

buoyancy_plot_end = buoyancy_end_node_element/2; %plot location of plot_node_y

% buoyancy force - selfweight hose
distributed_buoyancy_force = buoyancy_force_total/buoyancy_nodes_element;
Pext_g((buoyancy_coordinate_begin):2:(buoyancy_end_node_element)) = distributed_buoyancy_force*10*m; % maximaal belade gedeelte inclusief eigen gewicht, in iteratie r284 r 286
Pext_g_buoy = zeros(nDof,1);
Pext_g_buoy((buoyancy_coordinate_begin):2:(buoyancy_end_node_element)) = distributed_buoyancy_force*10*m;

%% 7 Iteration until convergence
CONV = 0;
kIter = 0;
nMaxIter = 1000;
fig_3 = figure;
while CONV == 0
    kIter = kIter + 1; % update the iteration counter
    fprintf(1, 'Iteration %d ...\n', kIter);
    % Check stability - define a number maximum of iterations. If solution
    % hasn't converged, check what is going wrong (if something).
    if kIter > nMaxIter, break; end

    % 7.1.1. Assemble vector with internal forces and stiffness matrix
    K = zeros(nDof,nDof);
    Fi = zeros(nDof,1);
    Pext = zeros(nDof,1);
    for iElem = 1: 1: nElem
        Nodes = Element(iElem, [1 2]); % Nodes of the element
        NodePos = NodeCoord(Nodes,:) + [u_it(2*Nodes-1) u_it(2*Nodes)]; % Position of nodes
        Dofs = [2*Nodes(1)-1; 2*Nodes(1); 2*Nodes(2)-1; 2*Nodes(2)];
        [Fi_elem,K_elem,Tension(iElem),WARN] = StringForcesAndStiffness(NodePos,EA,1Elem

```

```

    if NodePos(1,2) < 0 %x coordinaat?
        K_elem(2,2) = K_elem(2,2);% + k_soil/2;
    end
    if NodePos(2,2) < 0 %y coordinaat?
        K_elem(4,4) = K_elem(4,4);% + k_soil/2;
        Pext(Dofs(4)) = Pext_g(Dofs(4));% - k_soil*NodePos(2,2);
    else
        Pext(Dofs(4)) = Pext_g(Dofs(4));
    end
    Fi(Dofs) = Fi(Dofs) + Fi_elem;
    K(Dofs,Dofs) = K(Dofs,Dofs) + K_elem;
end

% 7.1.2. Calculate "residual forces"
R = Pext - Fi;

% 7.1.3. Check for convergence
if norm(R(FreeDof))/norm(Pext(FreeDof)) < 1e-4
    CONV = 1;
end
% 7.1.4. Calculate increment of displacements
du = zeros(nDof,1);
du(FreeDof) = K(FreeDof,FreeDof)\R(FreeDof);
if isnan(norm(du))
    du = rand(size(du))/lElem/100;
end
% 7.1.5. Apply arclength to help with convergence
Scale = min([1 lElem/max(abs(du))]);
du = du*Scale; % Enforce that each node does not displace
               % more (at each iteration) than the lenght
               % of the elements

% 7.1.6. Update displacement of nodes
u_it = u_it + du;
plot_node_x = (NodeCoord(:,1) + u_it(1:2:end-1));
plot_node_y = (NodeCoord(:,2) + u_it(2:2:end));
% 7.1.7. Plot updated configuration
% if 1 %kIter/10 == round(kIter/10)
%     plot(NodeCoord(:,1) + u_it(1:2:end-1), NodeCoord(:,2) + u_it(2:2:end), '-
ok')
%         hold on
%
%         plot([A(1) C(1)], [A(2) C(2)], 'vb');
%         % plot buoyancy locations
%         plot(plot_node_x(buoyancy_plot_begin:1:buoyancy_plot_end), plot_node_y(buoyancy_plo
%
%         axis equal
%         title(['Iteration ' num2str(kIter)]);
%         drawnow;
%         hold off;
%         legend({'Shape hose', 'Begin/end point', 'Buoyancy'}, 'Location', 'southeast')
%
%         % Display convergence
%         Rmax = max(abs(R(FreeDof)));
%         disp(Rmax);
%         disp(norm(R(FreeDof))/norm(Pext(FreeDof)))

```

```

%         drawnow;
%         hold off;
%     end
end
%     if CONV %eindplot
%     plot(NodeCoord(:,1) + u_it(1:2:end-1), NodeCoord(:,2) + u_it(2:2:end), '-
ok')
%         hold on
%
%         plot([A(1) C(1)], [A(2) C(2)], 'vb');
%         % plot buoyancy locations
%         plot(plot_node_x(buoyancy_plot_begin:1:buoyancy_plot_end), plot_node_y(buoyancy_
%
%
%         axis equal
%         title(['Iteration ' num2str(kIter)]);
%         drawnow;
%         hold off;
%         legend({'Shape hose', 'Begin/end point', 'Buoyancy'}, 'Location', 'southeast')
%
%     % Display convergence
%     Rmax = max(abs(R(FreeDof)));
%     disp(Rmax);
%     disp(norm(R(FreeDof))/norm(Pext(FreeDof)))
%     drawnow;
%     hold off;
%     title('Converged solution');
% else
%     title('Solution did not converge.');
```



```

% xlabel('Node number')
% ylabel('Tension [N]')
% legend('Tension Catenary System', 'Tension Buoyant System', 'Buoyant Nodes')

% filename_5= strcat('FIG2_TENSION_SSHAPE_L',num2str(L_1),'_W',num2str(W),'_H',num2str(H),
% saveas(fig_5,filename_5,'png')

%% 8 Convergentie check
% first half of hose for max
coordinate_sshape_1 = [plot_node_x(1:1:nNode/2),plot_node_y(1:1:nNode/2)];
coordinate_max_y_1 = max(coordinate_sshape_1(:,2));
coordinate_y_1 = find(plot_node_y == coordinate_max_y_1);
coordinate_x_1 = plot_node_x(coordinate_y_1);
coordinate_min_y_1 = min(coordinate_sshape_1(:,2));
coordinate_y_1b = find(plot_node_y == coordinate_min_y_1);
coordinate_x_1b = plot_node_x(coordinate_y_1b);
% second part of hose for minimum
coordinate_sshape_2 = [plot_node_x(coordinate_y_1+1:nNode),plot_node_y(coordinate_y_1+1:nNode)];
coordinate_min_y_2 = min(coordinate_sshape_2(:,2));
coordinate_y_2 = find(plot_node_y == coordinate_min_y_2);
coordinate_x_2 = plot_node_x(coordinate_y_2);
% coordinate check bend restrictor
coordinate_bendrestrictor = [plot_node_x(2:1:point_hosesegment),plot_node_y(2:1:point_hosesegment)];
coordinate_br_x = coordinate_bendrestrictor(1:1:end,1);
coordinate_br_y = coordinate_bendrestrictor(1:1:end,2);
% Tension Catenary gegevens
plot_tension_cat = Tension_catenary';
Tension_cat_begin = Tension_catenary(1,1); % [N]
Tension_cat_min = min(Tension_catenary); % [N]
Tension_cat_min_node = find( plot_tension_cat == Tension_cat_min);
Tension_cat_max = max(Tension_catenary); % [N]
Tension_cat_max_node = find( plot_tension_cat == Tension_cat_max);
Tension_cat_end = Tension_catenary(1,nElem); % [N]
% Tension Buoyancy gegevens
plot_tension = Tension';
Tension_buoy_begin = Tension(1,1); % [N]
Tension_buoy_max = max(Tension); % [N]
Tension_buoy_max_elem = find( plot_tension == Tension_buoy_max);
Tension_buoy_max_start = Tension(1,((point_hosesegment*start_segment)-1));
Tension_buoy_max_start_elem = point_hosesegment*start_segment; % element voor buoyancy force
Tension_buoy_min = min(Tension); % [N]
Tension_buoy_min_node = find( plot_tension == Tension_buoy_min);
Tension_buoy_end = Tension(1,nElem); % [N]

% Table information saving
% Matrix Nodes max en min
table_1(k,1) = L_1; % [m]
table_1(k,2) = H; % [m]
table_1(k,3) = number_of_modules; % [st]
table_1(k,4) = number_of_modules_per_hose; % [st]
table_1(k,5) = start_segment; % [No]
table_1(k,6) = mass; % [kg/m]
table_1(k,7) = W; % [m]
table_1(k,8) = coordinate_x_1b; % [m]
table_1(k,9) = coordinate_min_y_1; % [m]
table_1(k,10) = coordinate_x_1; % [m]
table_1(k,11) = coordinate_max_y_1; % [m]

```

```

table_1(k,12) = coordinate_x_2;           % [m]
table_1(k,13) = coordinate_min_y_2;     % [m]

% Matrix Nodes, save sshape for figure plot
kk = k*2; kkk = round_odd_down(kk);
table_11(1,kk) = L_1;                   % [m]
table_11(2,kk) = H;                     % [m]
table_11(3,kk) = number_of_modules;     % [st]
table_11(4,kk) = number_of_modules_per_hose; % [st]
table_11(5,kk) = start_segment;         % [No]
table_11(6,kk) = mass;                  % [kg/m]
table_11(7,kk) = W;                     % [m]
table_11(8,kk) = buoyancy_plot_begin;   % [No]
table_11(9,kk) = buoyancy_plot_end;     % [No]
table_11(10,kk) = nDof;
table_11(11,kk) = nNode;
table_11(11+(1:numel(plot_node_y)),kk) = plot_node_y; % [m]
table_11(11+(1:numel(plot_node_x)),kkk) = plot_node_x; % [m]

% % Matrix Nodes max en min
% table_111(k,1) = L_1;                 % [m]
% table_111(k,2) = H;                   % [m]
% table_111(k,3) = number_of_modules;   % [st]
% table_111(k,4) = number_of_modules_per_hose; % [st]
% table_111(k,5) = start_segment;       % [No]
% table_111(k,6) = mass;                % [kg/m]
% table_111(k,7) = W;                   % [m]
% table_111(k,8) = coordinate_cat_x_1b;  % [m]
% table_111(k,9) = coordinate_cat_min_y_1; % [m]
% table_111(k,10) = coordinate_cat_y_1b; % [no y]

% Matrix Force per iteration
table_2(k,1) = L_1;                     % [m]
table_2(k,2) = H;                       % [m]
table_2(k,3) = number_of_modules;       % [st]
table_2(k,4) = number_of_modules_per_hose; % [st]
table_2(k,5) = start_segment;           % [No]
table_2(k,6) = mass;                    % [kg/m]
table_2(k,7) = W;                       % [m]
table_2(k,8) = buoyancy_force_total;    % [N]
table_2(k,9) = sum(Pext_g_buoy);        % [N] check up above
table_2(k,10) = 10;                     % [m] wat misschien het verschil veroorzaakt
table_2(k,11) = (g*10*m);               % [N] eigengewicht per node
table_2(k,12) = (g*10*m)*(nNode-1);     % [N] begin en end node maar halve hose aan gewicht
table_2(k,13) = sum(Pext_eigengewicht); % [N] check up Verdeling eigengewicht
table_2(k,14) = (distributed_buoyancy_force-g*10*m); % [N] Force Buoyant node
table_2(k,15) = sum(Pext_g);            % [N] Total force on hose, load vector
table_2(k,16) = (sum(Pext_g_buoy)+sum(Pext_eigengewicht)); % [N] verschil tussen buoyancy and weight

% Matrix Force 2 Pext_g total
table_22(1,k) = L_1;                    % [m]
table_22(2,k) = H;                      % [m]
table_22(3,k) = number_of_modules;      % [st]
table_22(4,k) = number_of_modules_per_hose; % [st]
table_22(5,k) = start_segment;          % [No]
table_22(6,k) = mass;                   % [kg/m]
table_22(7,k) = W;                      % [m]

```

```

table_22(8,k) = sum(Pext_g_buoy); % [N] total force
table_22(9,k) = distributed_buoyancy_force; % [N] force per node
table_22(10,k) = number_of_buoyant_hoses_element; % [st] exact nodes buoyant
table_22(11,k) = buoyancy_nodes_element; % [st] afgerond plotted nodes buoyant
table_22(12,k) = buoyancy_coordinate_begin; % [No]
table_22(13,k) = buoyancy_nodes_element_add; % [st] afgerond
table_22(14,k) = buoyancy_end_node_element; % [No]
table_22((14+(1:numel(Pext_g))),k) = Pext_g/1000; % [kN]

% Matrix Tension cat info
table_3(k,1) = L_1; % [m]
table_3(k,2) = H; % [m]
table_3(k,3) = number_of_modules; % [st]
table_3(k,4) = number_of_modules_per_hose; % [st]
table_3(k,5) = start_segment; % [No]
table_3(k,6) = mass; % [kg/m]
table_3(k,7) = W; % [m]
table_3(k,8) = Tension_cat_begin; % [N]
table_3(k,9) = Tension_cat_min; % [N]
table_3(k,10) = Tension_cat_min_node(1,1); % [No]
table_3(k,11) = Tension_cat_max; % [N]
table_3(k,12) = Tension_cat_max_node; % [No]
table_3(k,13) = Tension_cat_end; % [N]
table_3(k,14) = R_catenary(1,1); % [N] Resudual force fixed node X Crawler
table_3(k,15) = R_catenary(2,1); % [N] Resudual force fixed node Y Crawler
table_3(k,16) = R_catenary(nDof-1,1); % [N] Resudual force fixed node X
table_3(k,17) = R_catenary(nDof,1); % [N] Resudual force fixed node X

% Martix Tension sshape info
table_33(k,1) = L_1; % [m]
table_33(k,2) = H; % [m]
table_33(k,3) = number_of_modules; % [st]
table_33(k,4) = number_of_modules_per_hose; % [st]
table_33(k,5) = start_segment; % [No]
table_33(k,6) = mass; % [kg/m]
table_33(k,7) = W; % [m]
table_33(k,8) = Tension_buoy_begin; % [N]
table_33(k,9) = Tension_buoy_max ; % [N]
table_33(k,10) = Tension_buoy_max_elem; % [No]
table_33(k,11) = Tension_buoy_max_start; % [N]
table_33(k,12) = Tension_buoy_max_start_elem; % [No]
table_33(k,13) = Tension_buoy_min; % [N]
table_33(k,14) = Tension_buoy_min_node(1,1); % [No]
table_33(k,15) = Tension_buoy_end; % [N]
table_33(k,16) = R(1,1); % [N] Resudual force fixed node X Crawler
table_33(k,17) = R(2,1); % [N] Resudual force fixed node Y Crawler
table_33(k,18) = R(nDof-1,1); % [N] Resudual force fixed node X
table_33(k,19) = R(nDof,1); % [N] Resudual force fixed node X

table_333(1,k) = L_1; % [m]
table_333(2,k) = H; % [m]
table_333(3,k) = number_of_modules; % [st]
table_333(4,k) = number_of_modules_per_hose; % [st]
table_333(5,k) = start_segment; % [No]
table_333(6,k) = mass; % [kg/m]
table_333(7,k) = W; % [m]
table_333((7+(1:numel(Tension'))),k) = Tension'; % [kN]

```

```

%Tabel parameters fail or not to fail
table_4_all_comb(k,1) = L_1; % [m]
table_4_all_comb(k,2) = L; % [m]
table_4_all_comb(k,3) = H; % [m]
table_4_all_comb(k,4) = number_of_modules; % [st]
table_4_all_comb(k,5) = number_of_modules_per_hose; % [st]
table_4_all_comb(k,6) = start_segment; % [No]
table_4_all_comb(k,7) = mass; % [kg/m]
table_4_all_comb(k,8) = W; % [m]
table_4_all_comb(k,9) = Tension_buoy_begin/1000; % [kN]
table_4_all_comb(k,10) = Tension_buoy_max/1000; % [kN]
table_4_all_comb(k,11) = coordinate_min_y_1; % [m]
table_4_all_comb(k,12) = coordinate_min_y_2; % [m]
table_4_all_comb(k,12+(1:numel(coordinate_br_y))) = coordinate_br_y; % [m]

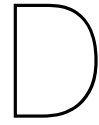
close all
end
%% Opslaan Totale Tabellen
% ssp2 = static solution parameters
% pv = parameter variation
save((strcat('Matrix_SSP2_PV_Basisinfo_1_', num2str(point_hoseselement), '.mat')), 'table_0')
save((strcat('Matrix_SSP2_PV_Basisinfo_2_', num2str(point_hoseselement), '.mat')), 'table_00')
save((strcat('Matrix_SSP2_PV_Basisinfo_3_', num2str(point_hoseselement), '.mat')), 'table_0000')
save((strcat('Matrix_SSP2_PV_Basisinfo_4_', num2str(point_hoseselement), '.mat')), 'table_000')
save((strcat('Matrix_SSP2_PV_Nodes_1_', num2str(point_hoseselement), '.mat')), 'table_1')
save((strcat('Matrix_SSP2_PV_Nodes_2_', num2str(point_hoseselement), '.mat')), 'table_11')
% save((strcat('Matrix_SSP2_PV_Nodes_3_', num2str(point_hoseselement), '.mat')), 'table_111')
save((strcat('Matrix_SSP2_PV_Force_1_', num2str(point_hoseselement), '.mat')), 'table_2')
save((strcat('Matrix_SSP2_PV_Force_2_', num2str(point_hoseselement), '.mat')), 'table_22')
save((strcat('Matrix_SSP2_PV_Tension_Cat_', num2str(point_hoseselement), '.mat')), 'table_3')
save((strcat('Matrix_SSP2_PV_Tension_Sshape_', num2str(point_hoseselement), '.mat')), 'table_33')
save((strcat('Matrix_SSP2_PV_Tension_Sshape_full_', num2str(point_hoseselement), '.mat')), 'table_333')
save((strcat('Matrix_SSP2_PV_Matrix_all_combinations_', num2str(point_hoseselement), '.mat')), 'table_4_all_comb')

%% CSV
% Construeren CSV files
format bank % bepaalde afronding voor dollars officieel --> afronding naar 1 cent
writematrix(table_0, strcat('SSP2_PV_Basisinfo_1_csv_', num2str(point_hoseselement), '.csv'), 'table_0')
writematrix(table_00, strcat('SSP2_PV_Basisinfo_2_csv_', num2str(point_hoseselement), '.csv'), 'table_00')
writematrix(table_0000, strcat('SSP2_PV_Basisinfo_3_csv_', num2str(point_hoseselement), '.csv'), 'table_0000')
writematrix(table_000, strcat('SSP2_PV_Basisinfo_4_csv_', num2str(point_hoseselement), '.csv'), 'table_000')
writematrix(table_1, strcat('SSP2_PV_Nodes_1_csv_', num2str(point_hoseselement), '.csv'), 'table_1')
writematrix(table_11, strcat('SSP2_PV_Nodes_2_csv_', num2str(point_hoseselement), '.csv'), 'table_11')
% writematrix(table_111, strcat('SSP2_PV_Nodes_3_csv_', num2str(point_hoseselement), '.csv'), 'table_111')
writematrix(table_2, strcat('SSP2_PV_Force_1_csv_', num2str(point_hoseselement), '.csv'), 'table_2')
writematrix(table_22, strcat('SSP2_PV_Force_2_csv_', num2str(point_hoseselement), '.csv'), 'table_22')
writematrix(table_3, strcat('SSP2_PV_Tensioncat_csv_', num2str(point_hoseselement), '.csv'), 'table_3')
writematrix(table_33, strcat('SSP2_PV_Tensionsshape_csv_', num2str(point_hoseselement), '.csv'), 'table_33')
writematrix(table_333, strcat('SSP2_PV_Tensionsshape_full_csv_', num2str(point_hoseselement), '.csv'), 'table_333')
writematrix(table_4_all_comb, strcat('SSP2_PV_Matrix_all_combinations_csv_', num2str(point_hoseselement), '.csv'), 'table_4_all_comb')

format short % terugzetten naar normale afronding

toc

```



# Theoretical background OrcaFlex

## D.1. Calculation stages

5 calculation stages of the forces and moment of the segments of a line. **For this chapter, the documentation for OrcaFlex is used, [18].**

### D.1.1. Tension force

The tension in the axial spring-damper at the centre of the segment is the effective tension. This force vector points in the direction of the segment,  $s + n$  and given by:

$$T_e = T_w + (p_o a_o - p_i a_i) \quad (D.1)$$

Where:

$T_w$  = wall tension

$p_i$  = internal pressure, calculated from the contents pressure.

$p_o$  = external pressure. Calculated by allowing the static pressure head due to the instantaneous height difference between the point and mean water level.

$a_i$  = internal cross section areas of stress annulus,  $= \frac{\pi}{4} ID_{stress}^2$

$a_o$  = external cross sectional areas of stress annulus  $= \frac{\pi}{4} OD_{stress}^2$

The axial stiffness can be linear or non linear. For linear axial stiffness the wall tension is defined as:

$$T_w = EA\epsilon - 2\nu(p_o a_o - p_i a_i) + k_{tt} \frac{\tau}{l_0} + EAc \frac{dl}{dt} \frac{1}{l_0} \quad (D.2)$$

Where,

$EA$  = axial stiffness of the line

$\epsilon$  = total mean axial strain

$l$  = instantaneous length of segment

$\lambda$  = expansion factor of segment

$l_0$  = unstretched length of segment

$\nu$  = Poisson ratio

$k_{tt}$  = tension/torque coupling

$\tau$  = segment twist angle

$c$  = damping coefficient

$$\frac{dl}{dt} = \text{rate of increase of length}$$

For nonlinear stiffness, axial stiffness is calculated by  $T_w^{air}(\epsilon)$  and is represented by a variable data source.

#### D.1.1.1. Damping coefficient

The damping coefficient represents the numerical damping of the line calculated by:

$$c = \frac{\lambda_a}{100} c_{crit} \quad (D.3)$$

Where,

$\lambda_a$  = target tension damping, specified as a % of the critical damping level

$$c_{crit} = \sqrt{\frac{2ml_0}{EA}}$$

$m$  = segment mass

This numerical damping term is only included when using the explicit integration scheme. The implicit integration scheme includes in-built numerical damping.

#### D.1.2. Bend moments

The rotational spring-dampers induce a bending moment to the node. There is a linear bending stiffness and nonlinear bending stiffness. The linear bending stiffness can be isotropic or non-isotropic. Isotropic means that the bending stiffness is the same in the x and y-direction. Resulting in a bending moment of magnitude:

$$|\mathbf{m}_2| = EI|c| + d \frac{d|c|}{dt} \quad (D.4)$$

Where,

$EI$  = Bending stiffness

$d$  = bending damping value =  $\frac{\lambda_b}{100} d_{crit}$

$\lambda_b$  = target bending damping

$d_{crit}$  = bending critical damping value for segment =  $l_0 \sqrt{mEI l_0}$

$m$  = segments mass

Bend moment is generated in the binormal direction  $\mathbf{b}_2$

$\mathbf{b}_2$  = unit vector of direction ( $\mathbf{s}_z \times \mathbf{n}_z$ )

For linear non-isotropic bending stiffness the components of the bend moment are given separately as:

$$\begin{aligned} \mathbf{m}_2 \text{ component for } \mathbf{S}_{x2} \text{ direction} &= EI_x c_x + d_x \frac{dc_x}{dt} \\ \mathbf{m}_2 \text{ component for } \mathbf{S}_{y2} \text{ direction} &= EI_y c_y + d_y \frac{dc_y}{dt} \end{aligned} \quad (D.5)$$

Where,

$EI_x, EI_y$  = bending stiffness of segment in specified direction

$c_x, c_y$  = components of the curvature vector  $c$  in the  $\mathbf{S}_{x2}$  and  $\mathbf{S}_{y2}$  direction

$$d_x = \frac{\lambda_b}{100} l_0 \sqrt{mEI_x l_0}$$

$$d_y = \frac{\lambda_b}{100} l_0 \sqrt{mEI_y l_0}$$

In case of nonlinear isotropic bending stiffness, the magnitude of the bending moment is given by:

$$|\mathbf{m}_2| = BM(|c|) + d' \frac{d|c|}{dt} \quad (D.6)$$

Where,

$BM(|c|)$  = the function of relating curvature to bend moment

$$d' = \frac{\lambda_b}{100} d'_{crit}$$

$$d'_{crit} = \text{bending critical damping value for a segment} = l_0 \sqrt{mEI_{nom}l_0}$$

$EI_{nom}$  = nominal bending stiffness, defined to be the bending stiffness at zero curvature

A hysteretically or non-hysteretically interpreted curvature-moment data can be selected for nonlinear bend stiffness. Non-hysteretic means that the data are applied using a simple elastic model. For a hysteretic model, hysteresis effects are applied depending on the curvature's history.

### D.1.3. Shear Forces

The shear forces follows from the bend moment calculations. Each line segment is a straight stiff rod in which the bend moment vector varies from  $\mathbf{m}_1$  at the end closed to end A of the line and  $\mathbf{m}_2$  at the other end of the segment. Because the segment is stiff in bending, the bending moment varies linearly along with it; the shear force in the segment is then the constant vector representing the rate of change of bend moment along the length. The shear force  $\mathbf{f}_s$  is therefore given by:

$$\mathbf{f}_s = \mathbf{s}_z \times \frac{1}{l} (\mathbf{m}_2 - \mathbf{m}_1) \quad (D.7)$$

### D.1.4. Torsion moments

If torsion is included the torque moment is calculated. First for each segment the directions  $\mathbf{s}_{x1}$ ,  $\mathbf{s}_{y1}$ ,  $\mathbf{s}_{x2}$  and  $\mathbf{s}_{y2}$  are determined, see figure 5.2. These directions are needed to calculate the twist angle  $\tau$ , the angle between directions  $\mathbf{s}_{x1}$  and  $\mathbf{s}_{x2}$ .

In case of linear torsional stiffness, the torque generated by the torsion spring damper is a moment vector  $\mathbf{m}$  about the segment axial direction  $\mathbf{s}_z$ , with magnitude

$$|\mathbf{m}| = k \frac{\tau}{l_0} + k_{tt} \epsilon + c \frac{d\tau}{dt} \quad (D.8)$$

Where,

$k$  = torsional stiffness

$\tau$  = segment twist angle

$l_0$  = unstretched length of segment

$\frac{d\tau}{dt}$  = rate of twist

$k_{tt}$  = tension/torque coupling

$\epsilon$  = total mean axial strain

$c$  = torsional damping coefficient of the line

#### D.1.4.1. Damping coefficient

The damping coefficient represents the numerical damping of the line calculated by:

$$c = \frac{\lambda_t}{100} c_{crit} \quad (D.9)$$

Where,

$\lambda_t$  = target torsion damping, specified as a % of the critical damping level

$$c_{crit} = \sqrt{\frac{2I_z k}{l_0}}$$

$I_z$  = rotational moment of inertia of the segment

This numerical damping term is only included when using the explicit integration scheme. The implicit integration scheme includes in-built numerical damping.

## D.2. Content flow types

For uniform content, each section of the line is assumed to be full of contents of the given density. The flow rate and velocity are calculated by the following simple formula's:

$$\begin{aligned} \text{Volume flow rate} &= \text{Mass flow rate} / \rho \\ \text{Flow velocity} &= \text{Volume flow rate} / A \end{aligned} \quad (\text{D.10})$$

Where,

$\rho$  = the specified content density

$A$  = internal cross-sectional area

For slug flows, the content density varies along the arc length of the line and can vary over time. It results in a variation of mass, weight and centrifugal and Coriolis forces.

### D.2.1. Content flow effects

For the contents flow effect, the centrifugal and Coriolis force are used. The centrifugal force is seen as a force on a node due to the flow through a node. The contents flow into a node has a velocity of  $s_i \mathbf{u}_i$ , resulting in input of momentum of  $\rho a_i s_i^2 \mathbf{u}_i$ . With a corresponding output momentum of  $\rho a_o s_o^2 \mathbf{u}_o$ . The force on the contents that are required to achieve the change in the flow direction, from  $\mathbf{u}_i$  to  $\mathbf{u}_o$ , must therefore be the net change in the rate of momentum:

$$\rho a_o s_o^2 \mathbf{u}_o - \rho a_i s_i^2 \mathbf{u}_i \quad (\text{D.11})$$

Resulting in a centrifugal force,  $\mathbf{f}_{ce}$  on the node, must be equal and opposite to

$$\mathbf{f}_{ce} = \rho(a_i s_i^2 \mathbf{u}_i - a_o s_o^2 \mathbf{u}_o) = \rho a s^2 (\mathbf{u}_i - \mathbf{u}_o) \quad (\text{D.12})$$

*The Coriolis force due to movement of a segment.*

The Coriolis force causes objects to deflect to the right on their intended path in the northern hemisphere and to the left in the southern hemisphere, affecting the direction. The strength of the deflection is proportionate to the speed. Considering a segment between two nodes  $\mathbf{n}_i$  and  $\mathbf{n}_{i+1}$ , with a fixed global frame and a local moving frame. The moving frame has an origin which moves with node  $\mathbf{n}_i$ , and its z-axis always points in direction  $\mathbf{u}$  from  $\mathbf{n}_i$  to  $\mathbf{n}_{i+1}$ . Consider the contents of a segment. Its velocity relative to the local moving axes is,

$$\mathbf{p}' = \frac{r}{a\rho} \mathbf{u} \quad (\text{D.13})$$

Velocity relative to the global axis is,

$$\mathbf{v}_i + \frac{d\mathbf{p}}{dt} = \mathbf{v}_i + \mathbf{p}' + \boldsymbol{\omega} \times \mathbf{p} \quad (\text{D.14})$$

The acceleration relative to global axis is,

$$\begin{aligned} \frac{d}{dt}(\mathbf{v}_i + \mathbf{p}' + \boldsymbol{\omega} \times \mathbf{p}) &= (\mathbf{v}_i + \mathbf{p}' + \boldsymbol{\omega} \times \mathbf{p})' + \boldsymbol{\omega} \times (\mathbf{v}_i + \mathbf{p}' + \boldsymbol{\omega} \times \mathbf{p}) \\ &= 0 + 0 + \boldsymbol{\omega}' \times \mathbf{p} + \boldsymbol{\omega} \times \mathbf{p}' + \boldsymbol{\omega} \times \mathbf{v}_i + \boldsymbol{\omega} \times \mathbf{p}' + \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \mathbf{p}) \\ &= \boldsymbol{\omega}' \times \mathbf{p} + 2\boldsymbol{\omega} \times \mathbf{p}' + \boldsymbol{\omega} \times \mathbf{v}_i + \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \mathbf{p}) \end{aligned} \quad (\text{D.15})$$

Resulting in a Coriolis force on the segment of

$$f_{co} = 2lr(\boldsymbol{\omega} \times \mathbf{u}) \quad (\text{D.16})$$

With,

$$\boldsymbol{\omega} = \frac{1}{l} \mathbf{u} \times (\mathbf{v}_{i+1} - \mathbf{v}_i) \quad (\text{D.17})$$

The Coriolis force becomes,

$$\begin{aligned} \mathbf{f}_{co} &= 2r[\mathbf{u}(\mathbf{v}_{i+1} - \mathbf{v}_i)] \times \mathbf{u} \\ &= 2r[(\mathbf{u}\cdot\mathbf{u})(\mathbf{v}_{i+1} - \mathbf{v}_i) - (\mathbf{u}\cdot(\mathbf{v}_{i+1} - \mathbf{v}_i))\mathbf{u}] \\ &= 2r[(\mathbf{v}_{i+1} - \mathbf{v}_i) - (\mathbf{u} \text{ direction component of } (\mathbf{v}_{i+1} - \mathbf{v}_i))] \end{aligned} \quad (\text{D.18})$$



We then apportion this total Coriolis force on the segment into two equal parts, a force of

$$r(\text{component of } (v_i + 1-v_i) \text{ normal to } u) \quad (\text{D.19})$$

on each of the two nodes at the ends of the segment.

Where,

$\rho$  = contents density

$A$  = internal cross-sectional area

$s$  = contents flow speed

$r$  = mass flow rate =  $\rho as$

$l$  = segment length

$\mathbf{p}$  = position of the node relative to global axes

$\mathbf{v}$  = velocity of the node relative to global axes

$\mathbf{u}$  = unit vector in the contents flow direction

$\boldsymbol{\omega}$  = angular velocity of the local (moving) frame relative to global frame GXYZ

$\frac{dx}{dt}$  = rate of change of any quantity  $x$  relative to global axes

$x'$  = rate of change of any quantity  $x$  relative to local axes

## D.3. Morison equation

The Morison equation was originally for calculating wave loads on a fixed vertical cylinder, and the force has two components. The water particle acceleration (fluid inertia) and the particle velocity (drag force).

$$f = C_m \Delta a_f + \frac{1}{2} \rho C_d A |v_f| v_f \quad (\text{D.20})$$

Where,

$f$  = the fluid force (per unit length) on the body

$C_m$  = the inertia coefficient for the body

$\Delta$  = the mass of fluid displaced by the body

$A_f$  = the fluid acceleration relative to the earth

$\rho$  = the density of water

$C_d$  = the drag coefficient for the body

$A$  = the drag area

$v_f$  = the fluid velocity relative to earth.

Orcaflex uses an extended form of the Morison equation, which incorporates a moving body's force.

$$f = (C_m \Delta a_f + C_a \Delta a_b) + \frac{1}{2} \rho C_d A |v_r| v_r \quad (\text{D.21})$$

where,

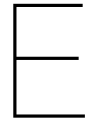
$C_a$  = the added mass coefficient for the body

$A_b$  = the body acceleration relative to the earth

$v_r$  = the fluid velocity relative to the body.

The inertia force consists of two parts, one proportional to fluid acceleration relative to earth (Froude-Krylov component) and one proportional to fluid acceleration relative to the body (Added mass component).





# OrcaFlex models

## E.1. Catenary check

- Model 1 test above water catenary same height
  - To verify the MatLab catenary calculations and learn to work with OrcaFlex
- Model 1 test underwater catenary same height
  - To verify the MatLab catenary calculations and learn to work with OrcaFlex
- Model 1 test underwater catenary height difference
  - To verify the MatLab catenary calculations and learn to work with OrcaFlex

## E.2. Old models

- Catenary 1
  - Learning to work with OrcaFlex
- Jumper hose
  - Learning to work with OrcaFlex
- Line object
  - Learning to work with OrcaFlex
- Vessel object
  - Learning to work with OrcaFlex

## E.3. Matlab check-up

- Model 1 buoyancy 5 segments per section forcing by hand
  - The MatLab model build in Orcaflex

## E.4. Parameter update check-up

- Model 5 segments
  - Starting point for updating OrcaFlex parameter
- Model 5 segments buoyancy
  - Implementing buoyancy in Matlab model by modules instead of hand forcing
- Model update 5 segments
  - Implementing a different number of segments
- Model update hose type 5 segments a

- Implementing parameters like bending stiffness, material parameters for 200 m distance between crawler and clump weight
- Model update hose type 5 segments b
  - Implementing parameters like bending stiffness, material parameters for 150 m distance between crawler and clump weight
- Model update hose type 5 segments c
  - Implementing parameters like bending stiffness, material parameters for 100 m distance between crawler and clump weight
- Model update hose type 5 segments d
  - Implementing parameters like bending stiffness, material parameters for 50 m distance between crawler and clump weight
- Model update hose structure 5 segments
  - Implementing hose structure of blue nodules design, coupling and reinforcement
- Model update hose structure new parameters 5 segments
  - New parameters due to failure of initial design parameters, forcing by hand to compare to MatLab.
- Model update hose structure buoyancy check
  - Implementing buoyancy in the new parameter model
- OrcaFlex model 32 modules 4 per hose
  - The final design for static analysis

## E.5. The dynamic study, x motion

These Orcaflex models study the dynamic motion of the jumper hose induced by the crawler in in line direction. The models were a built op fase to implement displacements in OrcaFlex. It also established the range of the crawler.

- OrcaFlex model 32 modules 4 per hose, 50 meters in the x-direction
- OrcaFlex model 32 modules 4 per hose, 100 meters in the x-direction
- OrcaFlex model 32 modules 4 per hose, 150 meters in the x-direction
- OrcaFlex model 32 modules 4 per hose, 50 meters in the x-direction, accelerating and decelerating

## E.6. The dynamic study, y-motion

These Orcaflex models study the dynamic motion of the jumper hose induced by the crawler in a perpendicular direction. It also established the range of the crawler.

- OrcaFlex model 32 modules 4 per hose, 50 meters in x-direction, range 150 meter in y-direction
  - Update for y-direction to 225 meter, 250 meter, 275 meter 300 meter, 280 meter
- OrcaFlex model 32 modules 4 per hose, 100 meters in x-direction, range 65 meter in y-direction
  - Update for y-direction to 144 meter, 215 meter, 240 meter 265 meter, 270 meter
- OrcaFlex model 32 modules 4 per hose, 150 meters in x-direction, range 47 meter in y-direction
  - Update for y-direction to 132 meter, 198 meter, 223 meter 248 meter, 250 meter, 260 meter

- OrcaFlex model 32 modules 4 per hose, 200 meters in x-direction, range 150 meter in y-direction
  - Update for y-direction to 200 meter, 210 meter, 225 meter
- OrcaFlex model 32 modules 4 per hose, 225 meters in x-direction, range 150 meter in y-direction
  - Update for y-direction to 175 meter, 185 meter
- OrcaFlex model 32 modules 4 per hose, 250 meters in x-direction, range 150 meter in y-direction
  - Update for y-direction to 155 meter
- OrcaFlex model 32 modules 4 per hose, 275 meters in x-direction, range 100 meter in y-direction
  - Update for y-direction to 75 meter
- OrcaFlex model 32 modules 4 per hose, 280 meters in x-direction, range 50 meter in y-direction
- OrcaFlex model 32 modules 4 per hose, 290 meters in x-direction, range 25 meter in y-direction
  - Update for y-direction to 30 meter
- OrcaFlex model 32 modules 4 per hose, 300 meters in x-direction, range 10 meter in y-direction
  - Update for y-direction to 25 meter

## E.7. VTS motions

These Orcaflex models study the dynamic motion of the jumper hose.

- OrcaFlex model 32 modules 4 per hose, heave motion
- OrcaFlex model 32 modules 4 per hose, heave motion, x-direction 150 meter
- OrcaFlex model 32 modules 4 per hose, heave motion, 100 m in x-direction, limit range in y 240 meter
- OrcaFlex model 32 modules 4 per hose, bocht, distance 200 meter
- OrcaFlex model 32 modules 4 per hose, increase the speed of crawler
- OrcaFlex model 32 modules 4 per hose, speed crawler and VTS same

## E.8. Range study zigzag pattern crawler and VTS motion

labelapp:orca mat range vts These Orcaflex models study the dynamic motion of the jumper hose and establish the range of the zigzag pattern with a constant VTS motion

- OrcaFlex model 32 modules 4 per hose, range crawler perpendicular 50 meters, range start location x-direction
  - Range update, 200, 250, 275 meters
- OrcaFlex model 32 modules 4 per hose, range crawler perpendicular 60 meters, range start location x-direction
  - Range update, 200, 250, 260 meters
- OrcaFlex model 32 modules 4 per hose, range crawler perpendicular 70 meters, range start location x-direction
  - Range update, 200, 225, 250, 255
- OrcaFlex model 32 modules 4 per hose, range crawler perpendicular 80 meters, range start location x-direction

- Range update 200, 240, 250, 255
- OrcaFlex model 32 modules 4 per hose, range crawler perpendicular 90 meters, range start location x-direction
  - Range update 200, 250
- OrcaFlex model 32 modules 4 per hose, range crawler perpendicular 100 meters, range start location x-direction
  - Range update 200, 240, 245
- OrcaFlex model 32 modules 4 per hose, range crawler perpendicular 120 meters, range start location x-direction
  - Range update 200, 240, 235
- OrcaFlex model 32 modules 4 per hose, range crawler perpendicular 140 meters, range start location x-direction
  - Range update 200, 220
- OrcaFlex model 32 modules 4 per hose, range crawler perpendicular 160 meters, range start location x-direction
  - Range update 200, 205
- OrcaFlex model 32 modules 4 per hose, range crawler perpendicular 180 meters, range start location x-direction
  - Range update 180, 185
- OrcaFlex model 32 modules 4 per hose, range crawler perpendicular 200 meters, range start location x-direction
  - Range update 150, 165
- OrcaFlex model 32 modules 4 per hose, range crawler perpendicular 220 meters, range start location x-direction
  - Range update 125, 135
- OrcaFlex model 32 modules 4 per hose, range crawler perpendicular 240 meters, range start location x-direction
  - Range update 100
- OrcaFlex model 32 modules 4 per hose, range crawler perpendicular 250 meters, range start location x-direction
  - Range update 50, 55

# Bibliography

- [1] Climate agreement 2015. 2015. URL: [http://unfccc.int/files/meetings/paris\\_nov\\_2015/application/pdf/paris\\_agreement\\_english\\_.pdf](http://unfccc.int/files/meetings/paris_nov_2015/application/pdf/paris_agreement_english_.pdf).
- [2] Amir Blanken. "Blue Nodules Deliverable report". In: *D2.10 Report dynamical analysis* (2020).
- [3] CORDIS. *CORDIS Results Pack on mineral extraction*. 2020. URL: [https://blue-nodules.eu/download/project\\_flyer\\_newsletters/articles/Cordis-Results-Pack-on-Mineral-Extraction.pdf](https://blue-nodules.eu/download/project_flyer_newsletters/articles/Cordis-Results-Pack-on-Mineral-Extraction.pdf).
- [4] Angela M. Cowan. *Resource library, Plate Tectonics, national geographic*. 2013. URL: [https://www.nationalgeographic.org/media/plate-tectonics/?utm\\_source=BibbliorCM\\_Row](https://www.nationalgeographic.org/media/plate-tectonics/?utm_source=BibbliorCM_Row) (visited on 10/24/2021).
- [5] Lev Egorov et al. *Sustainable seabed mining: guidelines and a new concept for Atlantis II Deep*. Vol. 4. University of Southampton, 2012.
- [6] The Editors of Encyclopaedia Britannica. *Archimedes principle*. 2020. URL: <https://www.britannica.com/science/Archimedes-principle> (visited on 12/10/2021).
- [7] GRID-Arendal. *Formation environment for manganese nodules*. 2014. URL: <https://www.grida.no/resources/7364> (visited on 10/25/2021).
- [8] GRID-Arendal. *Formation of Fe-Mn crusts*. 2013. URL: <https://www.grida.no/resources/8000> (visited on 10/25/2021).
- [9] Sighard F Hoerner. "Fluid Dynamic Drag, published by the author". In: *Midland Park, NJ* (1965), pp. 16–35.
- [10] Royal IHC. 2019. URL: <https://www.royalihc.com/en/news/european-consortium-launches-blue-nodules-project>.
- [11] Metin Inanc. *Compressing terrain elevation datasets*. Rensselaer Polytechnic Institute, 2008.
- [12] IPBES. *Policy support tool*. 2019. URL: <https://ipbes.net/policy-support/tools-instruments/multi-criteria-analysis-mca?page=1> (visited on 12/10/2021).
- [13] JAMSTEC. *Cobalt-rich Ferromanganese Crust Extended beyond 5,500m Depth in Ocean*. 2016. URL: [https://www.jamstec.go.jp/e/about/press\\_release/20160209/](https://www.jamstec.go.jp/e/about/press_release/20160209/) (visited on 10/25/2021).
- [14] Joao Barbosa. *W5: geometrically non-linear systems (axial deforming strings)*. Introduction to computational dynamics, Tu Delft. URL: <https://brightspace.tudelft.nl/d21/le/content/221620/viewContent/1556656/View> (visited on 11/20/2021).
- [15] Zsolt Kavacs. "Blue Nodules Deliverable report". In: *D2.7 Detailed design of jumper hose* (2020).
- [16] Jaco van der Hoeven Laurens de Jonge. "Blue Nodules Deliverable report". In: *D1.1 Terms of Reference* (2017).
- [17] Jaco van der Hoeven Laurens de Jonge. "Blue Nodules Deliverable report". In: *D2.4. Initial Design of Vehicle Propulsion and Propulsion Test Performance* (2017).
- [18] Orcina Ltd. *Orcaflex Manual, online version*. <https://www.orcina.com/webhelp/OrcaFlex/>. Orcina. Daltongate, Ulverston, Cumbria LA12 7AJ, UK, 2022.
- [19] Orcina Ltd. *Orcaflex Manual, online version, ch. Theory, sec. Dynamic theory*. <https://www.orcina.com/webhelp/OrcaFlex/>. Orcina. Daltongate, Ulverston, Cumbria LA12 7AJ, UK, 2022.
- [20] Orcina Ltd. *Orcaflex Manual, online version, ch. Theory, sec. Environment theory*. <https://www.orcina.com/webhelp/OrcaFlex/>. Orcina. Daltongate, Ulverston, Cumbria LA12 7AJ, UK, 2022.
- [21] Orcina Ltd. *Orcaflex Manual, online version, ch. Theory, sec. Line theory*. <https://www.orcina.com/webhelp/OrcaFlex/>. Orcina. Daltongate, Ulverston, Cumbria LA12 7AJ, UK, 2022.
- [22] Orcina Ltd. *Orcaflex Manual, online version, ch. Theory, sec. Static theory*. <https://www.orcina.com/webhelp/OrcaFlex/>. Orcina. Daltongate, Ulverston, Cumbria LA12 7AJ, UK, 2022.

- [23] Vincent Ellis McKelvey. *Subsea mineral resources*. 1689. US Government Printing Office, 1986.
- [24] A.V. Metrikine. *Dynamics, Slender Structures and an Introduction to Continuum Mechanics*. Brightspace, structural dynamics CIE4140. An optional note. Tu Delft.
- [25] NOAA. *What is a hydrothermal vent?* 2021. URL: <https://oceanservice.noaa.gov/facts/vents.html> (visited on 10/25/2021).
- [26] NOAA. *What's In a Nodule?* 2021. URL: <https://oceanexplorer.noaa.gov/oceanos/explorations/ex2104/features/nodule/welcome.html> (visited on 10/25/2021).
- [27] D. Papini. "On shape control of cables under vertical static load". <https://www.maths.lth.se/na/courses/Documents/papini.pdf>. MA thesis. Lund, Sweden: Lund University, Sept. 2010.
- [28] Arjan Schipaanboord. "Blue Nodules Deliverable report". In: *D2.1 Design Requirement Report for Subsea Harvesting Equipment* (2016).
- [29] Alex Svirin. *Differential Equations, Second order equations, Equation of catenary*. 2021. URL: <https://math24.net/equation-catenary.html> (visited on 11/18/2021).
- [30] CRISTIAN NEIPP TARSICIO BELEÂ NDEZ and AUGUSTO BELEÂ NDEZ. "Cables Under Concentrated Loads: A Laboratory Project for an Engineering Mechanics Course". In: *Int. J. Engng Ed* 1.19 (Feb. 2003), pp. 272–281.
- [31] Massachusetts Institute of Technology (MIT). 2019. URL: <https://www.youtube.com/watch?v=MWvCtFlitQM>.
- [32] Unknown. *Biology, plate tectonics*. 2013. URL: [https://www.mun.ca/biolofgy/scarr/Plate\\_Tectonics.html](https://www.mun.ca/biolofgy/scarr/Plate_Tectonics.html) (visited on 10/24/2021).
- [33] Paul Webb. *Introduction to oceanography chapter 1 part 2*. 2021. URL: <https://rwu.pressbooks.pub/webboceanography/chapter/1-2-continental-margins/> (visited on 10/24/2021).
- [34] Paul Webb. *Introduction to oceanography chapter 1 part 3*. 2021. URL: <https://rwu.pressbooks.pub/webboceanography/chapter/1-3-marine-provinces/> (visited on 10/24/2021).