

# SIMLA Version 3.24.0

## User Manual

Svein Sævik, Ole David Økland,  
Gro Sagli Baarholm and Janne K.Ø. Gjøsteen

June 15, 2025

## Contents

Contents	1
List of Figures	8
List of Tables	11
<b>1 Changelists</b>	<b>12</b>
1.1 Version numbering . . . . .	12
1.2 Changes in version 3.24 . . . . .	12
1.3 Changes in version 3.22 . . . . .	14
1.4 Changes in version 3.20 . . . . .	15
1.5 Changes in version 3.18 . . . . .	17
1.6 Changes in version 3.16 . . . . .	19
1.7 Changes in version 3.15 . . . . .	21
1.8 Changes in version 3.14 . . . . .	22
1.9 Changes in version 3.13 . . . . .	23
1.10 Changes in version 3.12 . . . . .	24
1.11 Changes in version 3.11 . . . . .	25
1.12 Changes in version 3.10 . . . . .	26
1.13 Changes in version 3.00 . . . . .	28
<b>2 Introduction</b>	<b>29</b>
2.1 Purpose of document . . . . .	29
2.2 Program basis . . . . .	29
2.3 SIMLA program files . . . . .	29
2.4 Basic concepts . . . . .	30
2.5 How to run SIMLA . . . . .	31
2.6 How to make SIMLA wait for a license . . . . .	33
<b>3 SIMLA input guide</b>	<b>34</b>
3.1 General . . . . .	34
3.2 BONCON - Boundary Conditions . . . . .	39
3.3 CLOAD - Concentrated loads . . . . .	40
3.4 CONSTR - Constraints . . . . .	41
3.4.1 PDISP . . . . .	41

LOCAL	42
GLOBAL	42
WAVE	43
RAO	43
SPECIAL	45
3.4.2 CONEQ	46
LOCAL	46
LOCSL	47
GLOBAL	47
3.4.3 HISTCONEQ	48
3.4.4 FEEDCONEQ	49
3.5 CONTINT - contact interfaces	50
3.6 CONTROL - CONTROL parameters	52
3.6.1 STRESSFREE	53
3.6.2 RESTART	53
3.6.3 EIGEN	53
3.6.4 FILEINIT	54
3.6.5 TIMEINIT	54
3.6.6 INIJLAY	55
3.6.7 INISLAY	56
3.6.8 TIMEINITSDD	56
3.6.9 AUTOSTART	57
3.6.10 SDDSTART	60
3.7 COSUPR Contact surface material properties	62
3.8 COSURFPR - Contact surface properties	62
3.8.1 Standard route file	64
3.8.2 Survey route file	64
3.8.3 Modelling of initial soil embedment	67
3.9 CROSSGEOM - Cross sectional geometry	68
3.9.1 TUBE	69
3.9.2 CYLR	69
3.9.3 CORR	69
3.9.4 BOX	70
3.9.5 RECT	70
3.9.6 GENERAL	72
3.10 CURLOAD - current loading	74
3.11 DYNCONT - Control parameters for dynamic analysis	74
3.12 DYNRES_ - dynamic results	75
3.12.1 DYNRES_N	75
3.12.2 DYNRES_E	76

3.12.3 DYNRES_I . . . . .	78
3.13 DROPS_GRID - Grid for hydrodynamic loading . . . . .	80
3.14 DROPS_HCOEF - Hydrodynamic coefficients . . . . .	83
3.15 DROPS_LOAD - Hydrodynamic loading . . . . .	86
3.15.1 Irregular wave loading with current . . . . .	87
Irregular wave loading with standard current profile . . . . .	88
Irregular wave loading with DNV-RP-F109 current profile . . . . .	89
3.15.2 Second order wave loading with current . . . . .	90
3.15.3 Wave kinematics read from file with current . . . . .	90
3.15.4 General comments . . . . .	91
3.16 ELCON - Element connectivity and properties . . . . .	94
3.16.1 PIPE . . . . .	97
3.16.2 COMPIPE . . . . .	97
3.16.3 CABLE . . . . .	97
3.16.4 CONT . . . . .	98
3.16.5 SPRING . . . . .	100
3.16.6 SEA . . . . .	100
3.16.7 HSHEAR . . . . .	100
3.16.8 BODY . . . . .	101
3.17 ELDAMP - Element damping properties . . . . .	102
3.17.1 BEAM . . . . .	103
3.17.2 CONTACT . . . . .	103
3.17.3 RALEIGH . . . . .	104
3.18 ELECC - Element eccentricity . . . . .	104
3.18.1 BEAM . . . . .	105
3.18.2 STINGER . . . . .	105
3.18.3 RADIUS . . . . .	108
3.19 ELHIST - element time histories . . . . .	111
3.20 ELLOAD - Element loads . . . . .	113
3.21 ELMASS - Element mass properties . . . . .	114
3.22 ELORIENT - Orientation of elements . . . . .	115
3.23 ELPROP - Element properties . . . . .	118
3.23.1 PIPE . . . . .	119
3.23.2 COMPIPE . . . . .	120
3.23.3 BELLMOUTH . . . . .	122
3.23.4 CONTBODY . . . . .	122
3.23.5 BODY . . . . .	124
3.23.6 CABLE . . . . .	125
3.23.7 USERCONTACT . . . . .	126
3.23.8 COAT . . . . .	127

3.23.9 ROLLER . . . . .	127
3.23.10 GENSPRING . . . . .	129
3.23.11 SOILCONTACT . . . . .	132
3.24 ENVRES_ - Envelope results . . . . .	134
3.24.1 ENVRES_N . . . . .	134
3.24.2 ENVRES_E . . . . .	134
3.24.3 ENVRES_I . . . . .	135
3.25 FATPROP - FATigue properties . . . . .	136
3.25.1 Fatigue file format . . . . .	136
3.26 FEED - lay simulation scenario definition (feed elements from lay vessel)	137
3.27 FLOWLOAD - Slug flow load model . . . . .	140
3.28 GEOM - geometry of body . . . . .	142
3.29 HEAD - HEADING . . . . .	148
3.30 HLA - lay simulation scenario definition . . . . .	148
3.31 HLAPLOT - HLA time plot results . . . . .	149
3.32 HLAVIS - General HLA output . . . . .	152
3.33 HYDROPRO - Hydrodynamic properties . . . . .	153
3.33.1 TRAWLBOARD . . . . .	154
3.33.2 Flat plate rotational drag model . . . . .	156
3.33.3 Interpolation of hydrodynamic mass . . . . .	156
3.33.4 Interpolation of drag and lift loads . . . . .	158
3.33.5 File input format for hydrodynamic coefficients . . . . .	160
3.34 INISTR - initial strain loading . . . . .	164
3.35 JOINTPR_APPLY - Apply pipeline joint properties . . . . .	165
3.36 JOINTPR_DEFINE - Define pipeline joint properties . . . . .	166
3.37 MATERIAL - MATERIAL properties . . . . .	168
3.37.1 LINEAR (PIPE31, PIPE34) . . . . .	170
3.37.2 ELASTOPLASTIC (PIPE33, PIPE39) . . . . .	171
3.37.3 EPCURVE . . . . .	172
3.37.4 HYCURVE . . . . .	174
3.37.5 ISOHYCURVE . . . . .	175
3.37.6 SEA . . . . .	175
3.37.7 RESULTANT (COMPIPE42) . . . . .	176
3.37.8 CONTACT (CONT124-126, 130, 152, 164, 170) . . . . .	178
3.37.9 ISOCONTACT (CONT130, CONT152, CONT164) . . . . .	182
3.37.10 ISOKXYCONTACT (CONT153, CONT164) . . . . .	183
3.37.11 R_CONTACT (CONT126) . . . . .	183
3.37.12 CLAY_Y (CONT126) . . . . .	188
3.37.13 SAND_Y (CONT126) . . . . .	190
3.37.14 BREAKOUT_Y (CONT126) . . . . .	191

Undrained soil . . . . .	192
Drained soil . . . . .	195
Userdefined model . . . . .	197
3.37.15 UNDRAIN_Z (CONT126) . . . . .	198
3.37.16 DRAIN_Z (CONT126) . . . . .	200
3.37.17 HYPERELASTIC . . . . .	200
3.37.18 LINSPRING (SPRING136) . . . . .	201
3.37.19 SPRING (CONT128) . . . . .	201
3.37.20 GENSPRING (SPRING137) . . . . .	202
3.37.21 CONCRETE (HSHEAR342) . . . . .	202
3.38 MOVE_GROUP - move group during autostart . . . . .	203
3.39 NOCOOR - Nodal coordinates of model . . . . .	204
3.40 NODPROP - node interpolated element properties . . . . .	207
3.41 NOORIENT - orientation of nodes . . . . .	208
3.42 PELOAD - external pressure and gravity loading . . . . .	209
3.43 PILOAD - internal pressure load . . . . .	210
3.44 RAOPROP - RAO properties . . . . .	211
3.45 READTRF - READ TRAnsfer function data . . . . .	214
3.46 REEL - reeling and straightening simulation scenario definition . . . . .	216
3.47 SEALO - Sea load specification . . . . .	218
3.48 SIMLA - lay simulation scenario definition . . . . .	219
3.49 TABLE - Table data . . . . .	222
3.50 THIST - time histories . . . . .	223
3.50.1 THIST_F - history from file . . . . .	224
3.50.2 THIST_H - harmonic load . . . . .	224
3.50.3 THIST_R - ramping loads smoothly on or off . . . . .	224
3.51 TIMECO - time control data . . . . .	226
3.52 TLOAD - temperature loading . . . . .	228
3.53 UNITS - SIMVIS unit conversion factors . . . . .	228
3.54 VISRES - visual results . . . . .	229
3.55 WAVELO - Regular and irregular wave loading . . . . .	230
3.55.1 General . . . . .	231
3.55.2 Regular wave loading, WAVETYPE=REGULAR . . . . .	232
3.55.3 Irregular wave loading, WAVETYPE=IRREGULAR . . . . .	232
<b>4 SIMPOST Report Generator</b>	<b>235</b>
4.1 General . . . . .	235
4.2 Input data . . . . .	235
4.3 NOPLOT - NOdal history PLOTs . . . . .	236
4.4 IPPLLOT - Integration Point history PLOTs . . . . .	237

4.5	ELPLOT - EElement history PLOTs . . . . .	238
4.6	GNPLOT - Global Nodal PLOTs . . . . .	240
4.7	GLPLOT - Global EElement PLOTs . . . . .	242
4.8	FAPLOT- Fatigue calculation . . . . .	244
4.9	NRPLOT- Nodal Reaction PLOTs . . . . .	245
4.10	RS PLOT - ReStart PLOT . . . . .	247
4.11	ENPLOT- ENvelope PLOTs . . . . .	247
4.12	VIVFAT - Vortex Induced Vibration FATigue . . . . .	248
5	DYNPOST Report Generator . . . . .	250
5.1	General . . . . .	250
5.2	Input data to DYNPOST . . . . .	250
5.2.1	MXPLOT - MaXima/MiNima postprocessing . . . . .	251
5.2.2	DYNPLOT - Dynamic plot postprocessing . . . . .	252
6	Searching for Errors . . . . .	253
7	Third party licenses . . . . .	254
8	Examples . . . . .	255
8.1	General remarks . . . . .	255
8.2	Dynamic simulation of pipeline installation by S-lay . . . . .	255
8.2.1	Modelling . . . . .	255
8.2.2	Example of results . . . . .	257
8.3	The cable kinking problem . . . . .	258
8.3.1	Objectives . . . . .	258
8.3.2	Modelling . . . . .	260
8.3.3	Validation of finite elements . . . . .	262
8.3.4	Case study - elastic catenary model . . . . .	262
8.3.5	Case Study - nonlinear moment-curvature model . . . . .	267
8.3.6	Dynamic criterion for evaluation of kink formation . . . . .	268
8.4	J-tube pull-in . . . . .	274
8.4.1	Objectives . . . . .	274
8.4.2	Input data . . . . .	274
8.4.3	Modelling . . . . .	274
8.4.4	Results . . . . .	276
8.5	Pipeline walking . . . . .	277
8.5.1	Objectives . . . . .	277
8.5.2	Input data . . . . .	277
8.5.3	Modelling . . . . .	278
8.5.4	Results . . . . .	278

8.6 Lateral buckling versus alternative soil models . . . . .	279
8.6.1 Review of lateral soil models . . . . .	279
8.6.2 Objectives . . . . .	281
8.6.3 Input data and modelling . . . . .	281
8.6.4 Results . . . . .	281
8.7 Pipeline trawl-gear interaction loads . . . . .	283
8.7.1 General remarks . . . . .	283
8.7.2 Objectives . . . . .	286
8.7.3 Input data . . . . .	286
8.7.4 Modelling . . . . .	286
8.7.5 Results . . . . .	288
Bibliography	289
Index	292

## List of Figures

2.1	Program system structure and analysis workflow with pre- and post-processing. . . . .	32
2.2	Contour plot of structure detail. . . . .	32
3.1	Coordinate system. . . . .	37
3.2	Node coordinate system. . . . .	37
3.3	Position of pipe in free span analysis ( <b>ICATEN = 0</b> ). . . . .	59
3.4	Vessel definition. . . . .	60
3.5	Input format for a route file with 3 lines. . . . .	64
3.6	Illustration of line definition for a route file with 3 lines. . . . .	65
3.7	Definition of 5 survey points, the $s$ -axis and the $x_1$ - and $y_1$ -coordinates. . . . .	66
3.8	Definition of $s$ - and $z$ -coordinates for 5 survey points and penetration $\bar{u}_{z0}$ . . . . .	66
3.9	Input format of 5 point survey file. . . . .	67
3.10	<b>CROSSGEOM</b> of type <b>TUBE</b> . . . . .	71
3.11	<b>CROSSGEOM</b> of type <b>CORR</b> . . . . .	71
3.12	<b>CROSSGEOM</b> of type <b>BOX</b> . . . . .	71
3.13	<b>CROSSGEOM</b> of type <b>GENERAL</b> . . . . .	73
3.14	<b>CROSSGEOM pipegen</b> of type <b>GENERAL</b> from example. . . . .	73
3.15	Element Gaussian station numbers s1 to s3 along the element axis. Integration points P1 and P2. . . . .	79
3.16	Definition of hydrodynamic grid . . . . .	81
3.17	Definition of wave spectrum for regular wave . . . . .	95
3.18	Rigid triangle with contact domain in green color. . . . .	99
3.19	No failure because contact points are separated by at least 2 edges located on distinct triangular geometry elements. . . . .	99
3.20	Failure in which contact is detected only at one contact point. . . . .	100
3.21	Element coordinates ( $x, y, z$ ), roller coordinates ( $x_1, y_1, z_1$ ), orientation angle $\varphi$ of the roller coordinate system and incremental change of arch length $ds$ and orientation angle $d\varphi$ between rollers. . . . .	108
3.22	Modelling of spiral with linearly changing radius of curvature. . . . .	109
3.23	Rotation-symmetric geometry for pipe-pipe external contact interaction modelled by the <b>RADIUS</b> option. . . . .	111
3.24	Rotation-symmetric geometry modelled by the <b>RADIUS</b> option. . . . .	112
3.25	<b>ELORIENT</b> command for PIPE and HSHEAR elements. . . . .	116

3.26	<b>ELORIENT</b> command for the CONT, BODY and SPRING elements. . . . .	116
3.27	$\partial\bar{C}$ and $\partial\bar{S}$ illustrates <b>IFULLEDGE</b> set to 0 and 1, respectively. . . . .	123
3.28	$C^0$ -continuous contact geometry for <b>IFULLEDGE</b> set to 0. . . . .	124
3.29	Defining maximum allowable incremental pipe contact point displacement by <b>CONTPAR2=VALUE</b> . . . . .	129
3.30	Analysis where <b>CONTPAR1=1</b> and <b>CONTPAR2=0</b> or <b>1</b> is recommended . . . . .	130
3.31	Standard FEED analysis. . . . .	138
3.32	The file 'flow.txt' defines the three fluid segments with lengths $L_i$ in the first column and densities $\rho_i$ in the second column. End 2 of element <b>IELREF</b> defines the origin of the curvilinear coordinate $s$ where the reference static internal pressure <b>PREF</b> is prescribed. The static pressure elsewhere is calculated based on the vertical height difference $h$ and by considering the fluid densities $\rho_i$ along the slug flow. . . . .	144
3.33	Illustration of material hardening effect. . . . .	172
3.34	Material curve combination. . . . .	182
3.35	Illustration of symmetric and anti-symmetric softening options. Here, the break-out displacement $y_{brk}$ has been exceeded in the positive displacement direction. This results in softening of the positive red yield force curve. For the symmetric softening option, the blue negative yield force curve for the opposite displacement direction will undergo the same softening. In contrast, if anti-symmetric softening is selected, there will be no softening for the opposite displacement direction. . . . .	194
3.36	Polar coordinate definition. . . . .	206
3.37	<b>NOORIENT</b> command. . . . .	209
3.38	Vessel RAO coordinate system in relation to global system. . . . .	212
3.39	Reeling. . . . .	216
3.40	A cylinder segment with local coordinate system and velocity vectors. . . . .	231
8.1	S-lay model . . . . .	256
8.2	Vessel dynamic displacement history in surge node ID 3001. . . . .	258
8.3	Vessel dynamic displacement history in heave node ID 3001. . . . .	258
8.4	Vessel dynamic displacement history in pitch node ID 3001. . . . .	259
8.5	Dynamic displacement in surge at pipe tensioner node ID 441. . . . .	259
8.6	Dynamic displacement in heave at pipe tensioner node ID 441. . . . .	260
8.7	Dynamic axial force in element ID 440. . . . .	260
8.8	Contact force plot at stinger tip. . . . .	261
8.9	Dynamic bending moment plot at stinger tip. . . . .	261
8.10	Straight beam model . . . . .	262
8.11	Torsion buckling of the straight beam model with 10 cm mesh size . . . . .	263

8.12	Study cases . . . . .	264
8.13	Torsion behavior for Case A . . . . .	264
8.14	Deformation pattern for Case A . . . . .	265
8.15	Torsion behavior for the case studies B and C . . . . .	265
8.16	Deformation patterns for the case study B (left) and the case study C (right) . . . . .	265
8.17	Bending behavior for Case C in the activated region . . . . .	266
8.18	Influence of tension variation on critical torsion moment . . . . .	266
8.19	Material models for parametric study . . . . .	267
8.20	Critical torsion moment variations along the length of the non-linear model . . . . .	267
8.21	Torsion buckling results for COMPIPE42 model . . . . .	268
8.22	Development of the maximum value of the total curvature for elastic model . . . . .	269
8.23	Time history - maximum total curvature - $\frac{M_x}{M_x^{crit}} = 0.25$ . . . . .	269
8.24	Time history - maximum total curvature - $\frac{M_x}{M_x^{crit}} = 0.5$ . . . . .	270
8.25	Time history - maximum total curvature - $\frac{M_x}{M_x^{crit}} = 0.8$ . . . . .	270
8.26	Maximum total curvature versus time . . . . .	271
8.27	Maximum total curvature versus time . . . . .	271
8.28	Axial force distribution for Heave amplitude $a_0 = 4$ m, $\frac{M_x}{M_x^{crit}} = 0.27$ . . . . .	272
8.29	J-tube pull-in model . . . . .	274
8.30	Pull-in force versus length of pull . . . . .	276
8.31	Pull force versus different stages of pull-in process . . . . .	277
8.32	Non-dimensional transient temperature profiles . . . . .	278
8.33	Axial force and associated displacements at different instants in time during start-up . . . . .	278
8.34	Explanation of walking physical process . . . . .	279
8.35	Displament at cool end as function of time . . . . .	279
8.36	Model for pipeline lateral buckling . . . . .	282
8.37	Comparison of lateral snaking results obtained for different soil models	283
8.38	Typical trawl gear set-up (DNV, 2014) . . . . .	284
8.39	Model used to simulate clump-weight/pipeline interaction . . . . .	286
8.40	Pipeline configuration during different pull-over phases . . . . .	288
8.41	Results in terms of warp-line force and pipe stresses . . . . .	288

## List of Tables

3.1	Overview of SIMLA element library.	96
3.2	Units for <b>EPS</b> and <b>SIGMA</b> .	173
8.1	STUDY INPUT DATA	263
8.2	INPUT PARAMETERS - SIMULATION OF J-TUBE PULL	275
8.3	INPUT PARAMETERS - SIMULATION OF FLOWLINE WALKING	277
8.4	INPUT PARAMETERS - SIMULATION OF GLOBAL BUCKLING	282
8.5	INPUT PARAMETERS - SIMULATION OF CLUMP-WEIGHT	287

# 1 Changelists

## 1.1 Version numbering

The version number consists of three numbers separated by periods, e.g. 3.18.0. The two first numbers refer to the version and the third number is updated for subsequent bugfix releases.

As of version 3.16, even numbered versions are reserved for official versions, e.g. 3.16 and 3.18, while odd numbered versions like 3.17 and 3.19 are reserved for development versions.

## 1.2 Changes in version 3.24

### General changes:

- A new lateral pipe-soil interaction model for modelling of the break-out response in undrained and drained soils has been implemented, see Section [3.37.14](#).
- New pipe-soil interaction models for prediction of initial embedment in undrained and drained soils have been implemented, see Sections [3.37.15](#) and [3.37.16](#).
- A logarithmic current profile which accounts for current-wave interaction has been implemented for DROPS analysis. See Section [3.15.1](#).
- Stokes second order wave kinematics for DROPS analysis has been implemented. See Section [3.15.2](#).
- A new feature for reading wave kinematics including current from file has been implemented for DROPS analysis. See Section [3.15.3](#).
- Extended user-control for print to screen and log file. Avoiding unnecessary printing reduces the CPU time. See Section [3.6](#).
- Time-scaling of the elastic stiffness for the PIPE31 element type has been implemented. See Section [3.19](#).
- An optional command line argument for setting the number of load steps needed for storage of DYNRES results was implemented, see Section [2.5](#).

### Input changes:

**DROPS\_LOAD - CHANGED** - The input data format was changed because of the new RPF109 current profile in Section 3.15.1. Unfortunately, the change makes old input files incompatible with the current SIMLA version.

**DROPS\_LOAD - CHANGED** - The new wave type SECONDORDER generates wave kinematics based on Stokes second order wave theory.

**DROPS\_LOAD - CHANGED** - The new wave type FROMFILE enables to read wave kinematics including current from file.

**CONTROL - CHANGED** - Four more options for the IPRINT-parameter were included to reduce printing to screen and log file.

**MATERIAL - CHANGED** - The optional parameter DTREVERS for the SAND\_Y model was removed.

**MATERIAL - CHANGED** - The optional parameter DTREVERS for the CLAY\_Y model was removed.

**MATERIAL - CHANGED** - The parameter GAMD was renamed to GAMSAT for the SAND\_Y model. This is because the parameter represents the soil saturated weight per volume, not the soil dry weight per volume.

**MATERIAL - NEW** - BREAKOUT\_Y for modelling of break-out response in both undrained and drained soils.

**MATERIAL - NEW** - UNDRAIN\_Z for prediction of initial embedment in undrained soil.

**MATERIAL - NEW** - DRAIN\_Z for prediction of initial embedment in drained soil.

**ELHIST - NEW** - For time-scaling of the elastic stiffness for PIPE31.

### 1.3 Changes in version 3.22

#### General changes:

- SIMPOST memory scaling based on command line arguments, see Chapter 4.
- Improvements and minor bugfixes for new features that were implemented in version 3.18 and 3.20: The DROPS features, the CONT153 element and the eigenvalue analysis type.
- Option for re-setting the accumulated plastic strain at reversed loading for the elasto-plastic material model, see Sec. 3.37.2.
- The element types CONT140 and CONT142 are no longer maintained and their input description has been removed from the user manual. It is recommended to use element type CONT130 instead.

#### Input changes:

No input cards have been changed.

## 1.4 Changes in version 3.20

This version was released only to participants in the DROPS JIP Phase I.

### General changes:

- The sandy soil model developed by Verley and Sotberg and the clay soil model developed by Verley and Lund have been implemented, see Section 3.37.12 and 3.37.13.
- The PONDUS hydrodynamic load model for on-bottom stability analysis has been implemented, see Sections 3.13, 3.14 and 3.15.
- Time-controlled hyper-elastic soil springs providing enhanced modelling capabilities of the analysis start-up phase, see SOILCONTACT element property type in Section 3.23.
- Penetration-dependent soil resistance curves in the local  $x$ -direction has been implemented for the CONT126 element type, see Sections 3.37.8 and 3.37.11.

### Input changes:

**DYNRES\_E - CHANGED** - DROPS hydrodynamic loads and wave kinematics for the element types PIPE31, PIPE33 and COMPIPE42.

**DYNRES\_I - CHANGED** - New result types for the CLAY\_Y and SAND\_Y soil models.

**MATERIAL - CHANGED** - The CONTACT and R\_CONTACT material types now allow for modelling of penetration-dependent soil resistance curves in the  $x$ -direction. Unfortunately, the new input format for the optional parameters is not compatible with input files for older SIMLA versions, see Sections 3.37.8 and 3.37.11.

**COSURFPR - CHANGED** - Improved description of the survey file format and modelling of initial soil embedment.

**DROPS\_HCOEF - NEW** - For specification of hydrodynamic coefficients for on-bottom stability analysis.

**DROPS\_GRID - NEW** - For defining grids where time series of wave kinematics and fixed pipe hydrodynamic forces are pre-generated for on-bottom stability analysis.

**DROPS\_LOAD - NEW** - For defining wave and current loading for on-bottom stability analysis.

**MATERIAL - NEW** - CLAY\_Y for modelling of lateral soil-pipe interaction in clay.

**MATERIAL - NEW** - SAND\_Y for modelling of lateral soil-pipe interaction in sand.

**ELPROP - NEW** - SOILCONTACT for modelling of hyper-elastic time-controlled soil springs.

**TABLE - NEW** - For defining data arrays. A table is required when defining DROPS hydrodynamic properties for pipe elements.

## 1.5 Changes in version 3.18

### General changes:

- New option for selecting initial or updated stiffness matrix for Rayleigh damping, see Section [3.17](#).
- A new slug flow model including transient contributions from static pressure, gravity, Coriolis forces and centrifugal forces has been implemented. See Section [3.27](#).
- The element type PIPE34 has been modified to include 6 internal ovalization degrees of freedom. This feature allows for analysis of the reduced bending stiffness effect of bends.
- The HYDROPRO type for modelling of trawl boards has been improved with respect to the angles applied for parametrizing the hydrodynamic mass and the drag and lift loads.
- A new contact element CONT153 that handles frictional contact between 3D bodies and pipelines has been implemented. See Section [3.16](#) and [3.23.4](#).
- All 2-dimensional elements have been removed.
- Eigenvalue analysis has been implemented, see Section [3.6.3](#).
- DYNPOST memory scaling based on command line arguments, see Chapter [5](#).

### Input changes:

**ELDAMP - CHANGED** - RALEIGH has been extended with one optional input parameter **UPDATE=VALUE** for selecting initial or updated stiffness matrix in the Rayleigh damping model.

**DYNRES\_E - CHANGED** - Force components for CONT153 and CONT164 can now be referred to global coordinates by means of the new optional input parameter **REFSYS**.

**DYNRES\_E - CHANGED** - HYDROPRO result types for the BODY502 element.

**ELPROP - CHANGED** - PIPE allows for including the **ELTIME** parameter for the PIPE34 element type. This is for activating the reduced bending stiffness effect at the time the full initial curvature of the bend has been introduced.

**ELPROP - CHANGED** - BODY has been extended with optional input data for including hydrodynamic Coriolis-Centripetal loads and a user-defined location of the hydrodynamic center.

**HYDROPRO** - **CHANGED** - TRAWLBOARD has been extended with optional input data for modelling of rotational drag loads.

**CONTROL** - **CHANGED** - The 2-dimensional analysis option has been removed.

**CONTROL** - **CHANGED** - Has been extended with the new analysis type EIGEN for performing Eigenvalue analysis.

**FLOWLOAD** - **NEW** - For simulation of transient slug loads.

**ELPROP** - **NEW** - CONTBODY for contact between pipelines and 3D bodies.

**MATERIAL** - **REMOVED** - The PONDUS soil model is not available in the present version.

## 1.6 Changes in version 3.16

### General changes:

- SIMLA must now be executed through the script SIMLA.BAT instead of SIMLA.EXE on Windows machines. See further instructions in Section 2.5.
- Increased flexibility and efficiency for modelling of roller-based contact geometries, see Sections 3.18.2 and 3.18.3.
- Enhanced contact algorithm for CONT164 with respect to contact search control parameters and constant stick stiffness, see Sections 3.23.9 and 3.37.10.
- Activation times for elasto-plastic material behavior implemented for PIPE33 and COMPIPE42.

### Input changes:

**ELECC - CHANGED** - STINGER allows now for efficient modelling of roller-based stinger contact geometries with linear variation of the curvature radius.

**ELPROP - CHANGED** - ROLLER can be restricted to only having one contact element with each PIPE element by means of the optional input parameter **CONT-PAR1=VALUE**.

**ELPROP - CHANGED** - ROLLER has been extended with one optional input parameter **CONT-PAR2=VALUE** for controlling the maximum allowable contact point displacements per time step.

**ELPROP - CHANGED** - PIPE has been extended with two optional input parameters for controlling the activation time for elasto-plastic material behavior. See input parameters **ELTIME=VALUE** and **AUTOPLAST=VALUE**.

**ELPROP - CHANGED** - COMPIPE has been extended with three optional input parameters for controlling the activation time for elasto-plastic material behavior. See input parameters **ELTIME=VALUE**, **AUTOPLAST=VALUE** and **TCURV=VALUE**.

**ELECC - NEW** - RADIUS for modelling of rotation-symmetric contact geometries consisting of several rollers.

**MATERIAL - NEW** - ISOKXYCONTACT allows for use of a constant stick stiffness for CONT164 that may improve the numerical performance in certain situations.

**NO PLOT - NEW** - This result option has been re-introduced in SIMPOST for printing of nodal translation and rotation components, see Sec. 4.3. Note that the **NO PLOT** option was employed for printing of stress and strain histories for SIMLA versions older than version 3.14.0.

**HYDROPRO - REMOVED** - RISER is not available in the present version.

## 1.7 Changes in version 3.15

### General changes:

- Arbitrary geometries may be included in visualization in XPost. See section [3.28](#).

### Input changes:

**GEOM - CHANGED** - Arbitrary geometry can be read from file.

## 1.8 Changes in version 3.14

### General changes:

- File name and memory extension available through options -n and -m on the command line. See section 2.5.
- The user can now make SIMLA wait for a license. See section 2.6.
- The stiffness of CONT130 and CONT152 now takes element length into account.
- More advanced hydrodynamic properties are available for PIPE and BODY elements through the **HYDROPRO** card.
- Option available for ROLLER making it possible to restrict the number of active contact elements for each pipe segment to 1.
- **NODAL** option in the **VISRES** card is obsolete. Thus the **NOPLLOT** is also obsolete. The same results can be retrieved by the use of **IPPLOT**.

### Input changes:

**ELPROP - CHANGED** - ROLLER can be restricted to only having one contact element with each PIPE segment.

**VISRES - CHANGED** - **NODAL** is obsolete.

**HYDROPRO - NEW** - Allowing more hydrodynamical properties for BODY and PIPE elements.

**NOPLLOT - REMOVED** - Not needed anymore.

## 1.9 Changes in version 3.13

### General changes:

- Time available on the X-axis in SimPost time history plots.
- Improvements related to the visualization of the seabed contact element in XPost.
- Ability to select 2D isotropic friction contact for elements CONT130, CONT124 and CONT164.
- Ability to select isotropic friction for seabed contact element CONT126.
- Ability to read a 5 point file format from pipeline survey data to build the seabed contact surface
- Ability to use a KP table based input (**COSUPR**) for the rock dumping contact element CONT128.
- Material associated with CONT128 is modified to include torsion.

### Input changes:Cards that have been changed:

**MATERIAL - CHANGED** - Isotropic friction included for CONT126, CONT130, CONT124 and CONT164. Torsional curve included for CONT128.

**COSURFPR - CHANGED** - Negative number of lines indicates the use of a five point file as input to analysis.

**ELPROP - CHANGED** - The rock burial element (CONT128) can be linked to a KP table (**COSUPR**)

## 1.10 Changes in version 3.12

### General changes:

- Including materialtype EPCURVE and HYCURVE for cable element.
- Including possibility to model horizontal break out resistance by means of two contributions:
  - 1) One elastoplastic spring that governs the pure friction Coulumb effect.
  - 2) One hyperelastic spring that is added to the above to describe the break-out peak resistance.

### Input changes:

**THIST** - **CHANGED** - Include harmonic and smooth ramping load history.

**MATERIAL** - **CHANGED** - New material type CONTACT\_R has been included to enable no coupling between torsion skin friction and passive earth pressure transverse soil resistance between pipe and soil. This is to allow simulation of rotation effects during laying.

**FEED** - **CHANGED** - Include optional parameters to model rotation in tensioner.

**HLAPLOT** - **CHANGED** - Include timeseries plot for towerdist, torque moment and rotation in tensioner.

**SIMLA** - **CHANGED** - Allow negative **NELPST** for retrieval analysis.

**MOVE\_GROUP** - **NEW** - Move element groups during autostart

## 1.11 Changes in version 3.11

### General changes:

- Including the cable/roller element CONT164 and cable/pipe element CONT170. This is to allow abandonment/recovery analysis using a combination of pipe and cable elements.

### Input changes:

**THIST - CHANGED** - Include harmonic and smooth ramping load history.

**MATERIAL - CHANGED** - New material type CONTACT\_R has been included to enable no coupling between torsion skin friction and passive earth pressure transverse soil resistance between pipe and soil. This is to allow simulation of rotation effects during laying.

**FEED - CHANGED** - Include optional parameters to model rotation in tensioner.

**HLAPLOT - CHANGED** - Include timeseries plot for towerdist, torque moment and rotation in tensioner.

**SIMLA - CHANGED** - Allow negative **NELPST** for retrieval analysis.

## 1.12 Changes in version 3.10

### General changes:

- More advanced coordinate, element and constraint equation definitions including recursive **REPEAT** command.
- The **ORIENT** concept is based on elements instead of nodes enabling consistent handling of many elements sharing the same node.
- More advanced **VISRES** command allowing user selected results to be shown in XPOST and post processed in SIMPOST.
- New element type HSHEAR342, enabling modeling of concrete coating.
- New element type CONT142, two node pipe in pipe contact element.
- New start algorithms related to Subsea Depressor Device (SDD).
- New element type CONT152, two node pipe in body contact element.
- New element type BODY502, one node element that can have a geometry given by the **GEOM** card.

### Input changes:

**CONSTR REPEAT - CHANGED** - Repeat can now be nested.

**CONTROL - CHANGED** - Option INITIATE has been split, and must now be defined as TIMEINIT or FILEINIT. New options SDDSTART and TIMEINITSDD.

**CONTINT - CHANGED** - New option for **IGAP** for CONT130 and CONT142.

**COSURFPR - CHANGED** - New parameter given after file name, describing the number of lines which are defined in the file. If several lines are given, default behavior is to interpolate between the lines. New optional parameter: if a line number is given, this line is used as it was the only line.

**CURLOAD - CHANGED** - Added parameter LOCAL/GLOBAL to allow current to be defined relative to vessel heading.

**ELCON REPEAT - CHANGED** - Repeat can now be nested. Extra parameter for element increment is included in the repeat sequence.

**ELPROP - CHANGED** - Two new optional parameters to give buoyancy and dry mass history for a specific group. This will overrule the general settings from the **PELOAD** card.

**FEED - CHANGED** - Restructuring of card, new options included.

**HLAVIS - CHANGED** - New format.

**HLA - CHANGED** - Parameter **VESSEL** is removed.

**HLAPLOT - CHANGED** - New plot types available. Plot type **SAGUTIL** has changed syntax.

**NOCOOR - CHANGED** - Including extra parameters to choose between COORDINATE or POLAR description of the node coordinates. For option POLAR, new parameters are included.

**SIMLA - CHANGED** - Order of input has been restructured, obsolete parameters removed and new ones included.

**THIST - CHANGED** - New optional parameters to allow for harmonic load history.

**TIMECO - CHANGED** - New types of analysis added: STATIC-FEED and STATIC-SDD. New optional parameters for iteration control.

**VISRES - CHANGED** - Including extra parameters to choose which visual results to be included and in which mode these are to be visualised. The mode options are **NODAL** and **INTEGRATION**.

**WAVELOAD - CHANGED** - For option **IRREGULAR** an extra parameter is included to give a seed for the wave generation.

**ORIENT - REMOVED** - Card is replaced with the two new cards **NOORIENT** and **ELORIENT**. It is no longer required to orient the nodes, but to orient the elements.

**ELORIENT - NEW** - Orients the elements. All elements except the sea elements must be oriented.

**NOORIENT - NEW** - Orients the nodes. This is not required, but must be done to be able to describe local boundary conditions etc.

**GEOM - NEW** - Describe geometry of a body element.

## 1.13 Changes in version 3.00

### General changes:

- Removing negative reaction forces at TDP.
- Improved vessel control algorithm.
- Including S-lay with roller distance control algorithm in SIMLA.
- Enable J-lay in force control mode to improve uphill and downhill laying.
- Including irregular wave dynamic analysis.
- Controlling the contact elements using time instead of step number.
- **PELOAD** definition is updated associating the histories to masses rather than forces.
- Initiation of dynamic analysis with pipe-configuration from given time in static laying analysis.
- HLAPLOT's for plot of support forces and layback.
- General improvements of HLA-plotting including stability plots for alternative parallel routes.
- New post processing programs DYNRES and DYNPOST for the **DYNRES** results, enabling effective result scanning to decide upon which sea states that are governing.

### Input changes:

**UNIT - NEW** - Card for specification of analysis units to obtain preferred units in HLA-plots.

**CONTINT - CHANGED** - Including new gap control parameters i.e. the two last parameters are no more dummy parameters and need to be defined for all CONT elements.

**CONTROL - CHANGED** - Including one extra parameter for the autostart option, the node ID number for the vessel COG. This enables the sea elements to be automatically centered about the vessel COG and the associated wave to be generated from the COG.

**COSURFPR - CHANGED** - Change in input to describe material properties on KP basis, and to allow for more than one line in a contact surface.

## 2 Introduction

### 2.1 Purpose of document

The purpose of this document is to describe the input to version 3.24.0 of the SIMLA program system. For details regarding verification testing of SIMLA and theoretical background, see the test and theory manuals, ([Sævik, 2007](#)) and ([Sævik, 2017](#)).

### 2.2 Program basis

SIMLA is a computer program for simulation of subsea pipelines. The development was started based on a request from Norsk Hydro in September 2000 related to simulating pipeline installation of the Ormen Lange Pipelines. Since then a lot of features have been implemented such as new element types and non-linear time domain dynamics. The computer program has been developed from scratch with respect to data structure, and flexibility has been provided for with respect to allowing easy implementation of future element types. The framework developed by Prof. Kolbein Bell and his colleagues related to the SAM library for formulation of FEM equations has been extensively used, making life easier for the FEM programmer. With respect to implementing time domain dynamics, benefit has been taken from the USFOS code. Thanks to former colleague Tore Holmås for valuable support related to utilizing the libraries developed at MARINTEK Structural Engineering. With respect to mathematical formulations, many references are relevant, see the theory manual ([Sævik, 2007](#)). Some keywords important to the development are summarized below:

- Co-rotated "Ghost reference" formulation for describing beam kinematics, same as RIFLEX, FENRIX.
- Contact formulations taken from previous developments BFLEX, PFLEX and BOUNDARY.
- Special seabed contact elements, springs acting relative to arbitrary user defined curve in space (route description).
- Material models based on the same principles as published by Sævik and Levold, ([Sævik and Levold, 1995](#)).

### 2.3 SIMLA program files

Version 3.24.0 involves the following files:

**prefix.sif** The SIMLA Input File.

**prefix.sof** The SIMLA Output File which contains model information.

**prefix.slf** The SIMLA Log File where warnings and error messages are written.

**prefix.raf** The SIMLA result database where model data and element/nodal results required to perform restart, visual presentation and post-processing are stored.

**prefix.dyn** The SIMLA result database where element/nodal results for all time steps are stored. Only printed if **DYNRES** cards are activated.

**prefix.1dat** A log data file which contains visualization and input data for the SIMVIS program. This file is not printed by default.

## 2.4 Basic concepts

SIMLA allows for both nonlinear static and dynamic analysis. In both cases the time domain is used to describe the load histories and the analysis sequence. The sequence of analysis is controlled by the **TIMECO** card, see Section 3.51, which defines a set of time intervals where different properties may apply with respect to step length, time interval for restart info and result storage, type of analysis in terms of static or dynamic and HLA result exchange with SIMVIS. An overview of the program system structure and the analysis workflow with pre- and post-processing is shown in Fig. 2.1.

The input data is given on ASCII files which are text string based. Depending on the time sequence defined, results may be stored and exchanged in different ways. For an analysis run with pre- and post-processing the procedure will be as follows:

1. Use FLEXEDIT to create a .sif input file with an analysis sequence defined by the **TIMECO** card, see Section 3.51. Typically static analysis is used in the first sequence and dynamic in the second sequence. The time interval for result and restart info storage to the .raf file contained in the result database in Fig. 2.1 is also defined by the **TIMECO** card. Note that only numerical data is stored on the .raf file if no time intervals for storage are given, and that the .raf file may become very large if results are stored for all load steps during time domain analysis. It is therefore recommended to select a set of results to be stored at each time step by applying the **DYNRES** card, see Section 3.12. Using the **DYNRES** card, results are stored on a separate .dyn file which utilizes a buffering technique to save computation time.
2. Run SIMLA using either FLEXEDIT or the command line. See Section 2.5 for further instructions. Results are printed to the data base file .raf and also to the file .dyn if the **DYNRES** card is activated.

3. SIMPOST is applied for processing of results stored on the .raf binary file, see Chapter 4. SIMPOST enables printing of fairly general user defined plots to .mpf ASCII-files. Further, the .mpf files can be imported into the MATRIXPLOT program for plotting, see ([Giertsen, 1998](#)).
4. The DYNPOST program is applied for processing of **DYNRES** results stored on the binary file .dyn, see Chapter 5. As opposed to SIMPOST, the results are available for all time steps. DYNPOST prints plot data to .mpf ASCII-files which can be imported in the MATRIXPLOT program for plotting, see ([Giertsen, 1998](#)).
5. XPOST is used for visualization and presentation of results. The card **VISRES** must be included in the .sif input file to allow for post-processing with XPOST, see Section 3.54. Contour plots as shown in Fig. 2.2 are defined by the **VISRES** card. XPOST also allows for exportation of animations to video files. The **DYNRES** time history plots are available in XPOST, in which the numerical values can be extracted by the cut and paste functionality in Windows (CTRL-C/CTRL-V).
6. SIMVIS is used for visualization and presentation of selected results. Typical applications are seabed intervention tasks and detailed visualization. HLA must be activated in the .sif input file to allow for post-processing with SIMVIS. For the time being, the user can utilize SIMVIS for two scenarios:
  - During static pipeline analysis along a predefined route, via the data cards **SIMLA** and **HLA** defined in Sections 3.48 and 3.30.
  - During arbitrary static and dynamic analysis by the **HLAVIS** data card see Section 3.32.

## 2.5 How to run SIMLA

SIMLA must be executed through the script SIMLA.BAT both when using the command line and FLEXEDIT on Windows machines. This script is available in the installation package and sets the required environment variables for Java and HLA.

A description on how to setup and run SIMLA from FLEXEDIT is available in the *SIMLA Quick Start User Guide*, see ([Giertsen et al., 2017](#)).

When running SIMLA from the command line on Windows machines, the basis is to write the name referring to the SIMLA.BAT script in the prompt. In addition, the following optional command line arguments can be applied:

- Adding -m and a float number representing the multiplication factor for scaling of the default work memory size used by SIMLA.

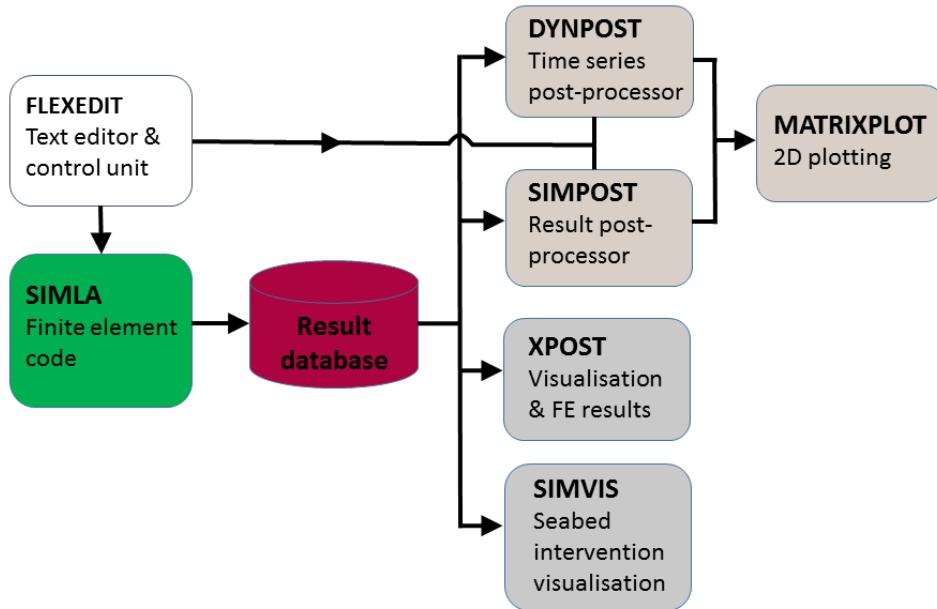


Figure 2.1: Program system structure and analysis workflow with pre- and post-processing.

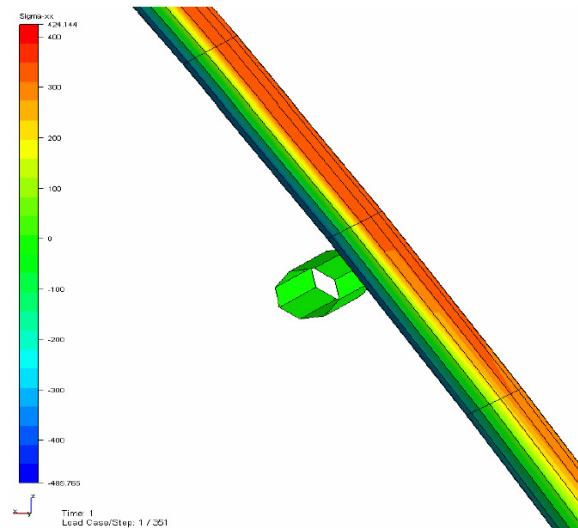


Figure 2.2: Contour plot of structure detail.

- Adding `-n` and the input file name prefix makes SIMLA execute the analysis without further requesting the user to specify the file name.
- Adding `-s` and a float number (larger than 0.001) multiplies the default maximum number (5000) of load steps that can be stored on the `*.raf` file. Increasing the number of steps will result in longer simulation time and a larger `*.raf` file.
- Adding `-d` and a float number representing the multiplication factor for scaling of the default memory size for **DYNRES** results used by SIMLA.

- Adding -s2 and an integer number representing the number of load steps needed for storage of dynres results. Only relevant when **DYNRES** card(s) are defined.

The optional arguments above may be specified in an arbitrary order.

In an analysis restart, SIMLA must be executed with the same -m, -s and -d multiplication factors as applied in the previous analysis, and also the same -s2 integer number.

The -d option is typically required when having a large number of time steps that are subjected to analysis restart, otherwise the **DYNRES** results may be cut off when the number of data points exceeds a certain amount of the allocated memory for **DYNRES** results.

EXAMPLE:

---

```
command line: SIMLA -m 1.56 -n prefixname -d 17.5 -s 10
```

---

## 2.6 How to make SIMLA wait for a license

To make SIMLA wait for a licence when running, set the environment variable MARIN-TEK\_LICENSE\_WAIT to yes.

## 3 SIMLA input guide

### 3.1 General

All input data is described in ASCII file format. Comment lines are defined by a "#" in the first column, i.e. at the start of the line. Empty lines are ignored. The input is organized in different data groups with individual identifiers, as listed below. Data belonging to one identifier will be read until next identifier or end of file is reached, i.e. input can be broken over lines. The data groups can be supplied in arbitrary order. The maximum line length is 136 characters and the maximum number of letters in one single text string is 32.

The input may be given by an arbitrary set of units if not otherwise noted. The user must ensure that a consistent unit set is applied.

Example input files for SIMLA test examples included in ([Sævik, 2007](#)) are supplied separately.

During a restart analysis, some input data such as boundary conditions and loading may be changed by the user. Data identifiers that can be changed during restart are marked with *ITALIC* font. The following identifiers define the different data groups:

<i>BONCON</i>	Boundary conditions
<i>CLOAD</i>	Concentrated loads
<i>CONSTR</i>	Constraints between nodal DOFs
<i>CONTINT</i>	Contact interfaces
<i>CONTROL</i>	Control parameters
<i>COSUPR</i>	Contact surface properties (soil description)
<i>COSURFPR</i>	Contact surface properties (route file)
<i>CROSSGEOM</i>	Cross sectional geometry
<i>CURLOAD</i>	Current load definition
<i>DROPS_GRID</i> <sup>(5)</sup>	Drops grid for hydrodynamic loading
<i>DROPS_HCOEF</i> <sup>(5)</sup>	Drops hydrodynamic coefficients
<i>DROPS_LOAD</i> <sup>(5)</sup>	Drops hydrodynamic loading
<i>DYNCONT</i>	Control parameters for dynamic analysis
<i>DYNRES_</i>	Dynamic result presentation
<i>ELCON</i>	Element connectivity and properties
<i>ELDAMP</i> <sup>(1)</sup>	Element damping properties

<b>ELECC</b>	Element eccentricities
<i>ELHIST</i>	Element time histories
<i>ELLOAD</i>	Element loads
<i>ELMASS</i>	Element mass properties
<b>ELORIENT</b>	Orientation of elements
<i>ELPROP</i>	Element properties
<b>ENVRES_E</b>	Envelope element results
<b>ENVRES_N</b>	Envelope node results
<b>ENVRES_I</b>	Envelope integration station results
<b>FATPROP</b>	Fatigue properties
<b>FEED</b>	Lay simulation scenario definition (feed elements from lay-vessel)
<b>FLOWLOAD</b>	Slug flow model
<b>GEOM</b>	Geometry of special element
<b>HEAD</b>	Heading describing the model
<b>HLA</b>	HLA simulation definition
<b>HLAVIS</b>	General HLA output
<b>HLAPLOT</b>	Plots for HLA
<b>HYDROPRO</b>	Hydrodynamic properties
<b>INISTR</b>	Initial strains, displacements or rotations
<b>JOINTPR_APPLY</b>	Pipeline joint definitions
<b>JOINTPR_DEFINE</b>	Pipeline joint definitions
<b>MATERIAL</b> <sup>(2)</sup>	Material property data
<b>MOVE_GROUP</b>	Move group during autostart
<b>NOCOOR</b>	Nodal coordinates of the model
<b>NODPROP</b>	Node interpolated element properties
<b>NOORIENT</b>	Orientation of nodes
<b>PELOAD</b>	External pressure and gravity loading
<b>PILOAD</b>	Internal pressure load
<b>RAOPROP</b>	RAO properties
<b>READTRF</b>	Import RAO properties
<b>REEL</b>	Reeling and straightening simulation scenario definition
<b>SEALO</b>	Sea load specification
<b>SIMLA</b>	Lay simulation scenario definition (requires constant FE-grid)
<b>TABLE</b>	Table data
<b>THIST</b> <sup>(3)</sup>	Time history data
<b>TIMECO</b> <sup>(4)</sup>	Time control data
<b>TLOAD</b>	Temperature loading
<b>UNITS</b>	SIMVIS unit conversion factors
<b>VISRES</b>	Activate results for visual presentation by XPOST
<b>WAVELOAD</b>	Wave loading

- (1) The properties for **BEAM** damping model are set at analysis start-up and cannot be changed during analysis restarts. For the **RALEIGH** damping model, the properties can be changed if the optional parameter **UPDATE=1** is applied, see Section 3.17 for further details.
- (2) Material properties can be changed if the number of points in material curves is kept unchanged. This is convenient e.g. for seabed friction description in combination with autostart, see Section 3.6.
- (3) It is not allowed to define new **THIST** input cards during analysis restarts. However, existing time histories can be extended with additional data lines if required.
- (4) **TIMECO** input cards that relate to load steps prior to the restart load step are not allowed to change. This is because restart information and analysis results are stored based on the load step number calculated according to the presently included **TIMECO** input cards.
- (5) The **DROPS\_GRID**, **DROPS\_HCOEF** and **DROPS\_LOAD** input cards shall only be changed during analysis restarts that occur before the time **TSTART** in Section 3.15.

In general, most structural input data can be modified during a restart. Input data that affects the element and node definitions cannot be changed, such as the **CONTROL**, **NOCOOR**, **ELORIENT**, **ELCON** and **ELECC** input cards.

In the following, the input data related to each identifier will be explained. The units are specified by using F = Force, M = mass, L = length, T = time, K = Temperature, E = Energy, R=radians. Square brackets ([ ]) are used to identify optional parameters.

The coordinate system is shown in Fig. 3.1. A right handed Cartesian coordinate system is applied. Gravity is applied in the negative global z-direction. If a sea surface element is present in the model, this surface is assumed to be positioned at  $z = 0$ . See Sections 3.47, 3.5, 3.10, 3.42, 3.55 and 3.44. Further, if a seabed contact surface is specified, this may be arbitrarily oriented and located in the SIMLA model coordinate system by specifying the direction angle and  $(x_0, y_0)$  position vector relative to the SIMLA coordinate system, see Section 3.8.

Moreover, wave loading may be specified from an arbitrary position  $x_0, y_0$ , assuming a sinus wave elevation function with zero phase angle if not otherwise specified, see Section 3.55. In order for the RAO functions to be valid, it is most convenient to ensure that the  $x_0, y_0$  for the wave is positioned at the vessel COG  $x_0, y_0$ , see Section 3.44. By the **AUTOSTART**, **TIMEINIT**, **SDDSTART** and **TIMEINITSDD** options in the **CONTROL** card (Section 3.6) this is automatically taken care of.

The local element end node coordinate system is shown in Fig. 3.2 where the element end is placed eccentric to the node. Positive eccentricities means that the element end is translated along positive axis relative to the element node. In the initial configuration the node coordinates system is parallel to the global system.

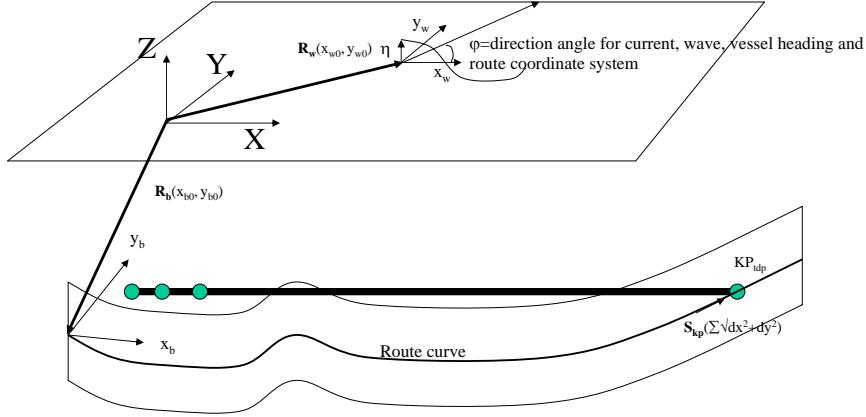


Figure 3.1: Coordinate system.

The model will always consist of nodes and elements. The orientation of each node is governed by the node coordinate system whereas the orientation of each element is governed by the element coordinate system. As default the node coordinate system is positioned parallel to the global coordinate system. However, the user may change that by the **NOORIENT** command, see Section 3.41.

The **REPEAT** command is used for several of the input cards. In all these cases the number of **REPEATs** are the total number of instances created. Thus, if a REPEAT of 1 is applied, the **REPEAT** has no effect.

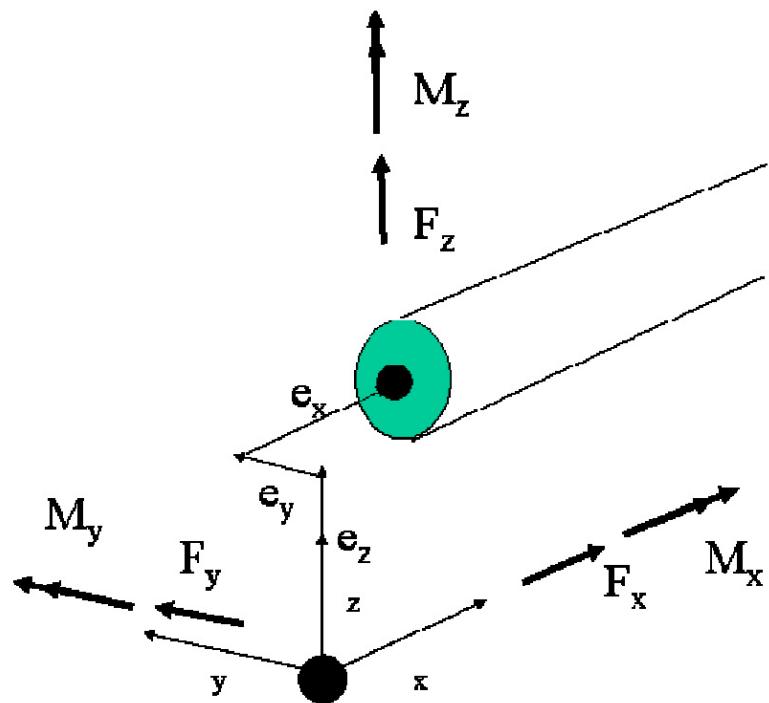


Figure 3.2: Node coordinate system.

### 3.2 BONCON - Boundary Conditions

The boundary conditions may be specified as global or local boundary conditions. The following format is applied:

**BONCON TYPE NODID DOF [MASTERNODE XANG YANG ZANG]  
[REPEAT N NODINC]**

where

**TYPE**: Type of boundary condition, which may have the values: **LOCAL**, **GLOBAL** or **SPECIAL**. **GLOBAL** and **LOCAL** means that the specified **DOF** is fixed in the global or local coordinates system, respectively. If the **SPECIAL** option is applied, the specified **DOF** is fixed in the local system of the **MASTERNODE**, or in a system with a specified rotation relative to the local **MASTERNODE** system.

**NODID**: Node ID number.

**DOF**: Nodal DOF number, where 1-3 is related to translation in xyz whereas 4-6 is related to rotation about xyz.

**MASTERNODE**: Master node number at which the applied boundary condition is referred. Only for **SPECIAL**.

**XANG**: Tait–Bryan angle around local x axis, applied to define the orientation of the applied boundary condition measured relative to the local system of the master node. Only for **SPECIAL** (unit: rad).

**YANG**: Tait–Bryan angle around local y axis, applied to define the orientation of the applied boundary condition measured relative to the local system of the master node. Only for **SPECIAL** (unit: rad).

**ZANG**: Tait–Bryan angle around local z axis, applied to define the orientation of the applied boundary condition measured relative to the local system of the master node. Only for **SPECIAL** (unit: rad).

The **REPEAT** command causes the previous sequence to be repeated N times:

**N**: Number of repeats.

**NODINC**: Node increment.

**EXAMPLE:**

```
#-----
# Boundary condition data
#-----
#
#      type    node    dof   repeat n nodinc
BONCON GLOBAL  1       1     REPEAT 3 2
BONCON GLOBAL  1       2
```

The example above will put a boundary condition in direction 1 on nodes 1 3 5. Boundary conditions in directions 2 and 3 are active only for the first node.

### 3.3 CLOAD - Concentrated loads

The concentrated nodal loads are defined by the following format:

```
CLOAD HIST DIR NODE LOAD [NODE2 LOAD2]
```

where

**HIST**: Load history number.

**DIR**: Load direction (**1-3** = load along x-z, **4-6** = moment about x-z).

**NODE**: First node ID.

**LOAD**: Load magnitude for the first node.

**NODE2**: Last node ID.

**LOAD2**: Load magnitude for the last node. Linear load interpolation is applied for intermediate nodes.

The load components refer by default to the global coordinate system. The only exception is when a special reference system is used for the motion of the node. A special reference system will be applied in the following cases:

- When the node is assigned the boundary condition types **LOCAL** or **SPECIAL**, see the **BONCON** card in Section 3.2. Then the load components will refer to the local or special coordinate system of the node, which rotates together with the node during its motion.
- When the node is a helix node. This is the case if the node also is defined as element node 3 or 4 for the HSHEAR342 element type, see **NOD3** and **NOD4** defined by the **ELCON** card in Section 3.16. Then the load components will refer to the local coordinate system of the helix node.

To summarise:

<b>DIR</b>	<b>Description</b>
<b>1</b>	force along x-axis (unit: F)
<b>2</b>	force along y-axis (unit: F)
<b>3</b>	force along z-axis (unit: F)
<b>4</b>	moment about x-axis (unit: FL)
<b>5</b>	moment about y-axis (unit: FL)
<b>6</b>	moment about z-axis (unit: FL)

EXAMPLE: \_\_\_\_\_

```
#  
# Concentrated nodal loads:  
#-----  
#      hist   dir   node   load  
CLOAD 50     1     3001   1.18  
#
```

---

### 3.4 CONSTR - Constraints

The **CONSTR** command allows the user to define constraints in the FE-model.

**CONSTR CTYPE ...**  
**[REPEAT ...]**

The repeat card can be applied.

**CTYPE:** Type of constraint.

The following types of constraints can be specified:

**PDISP :** Prescribed displacement.

**CONEQ :** Constraint equation.

**HISTCONEQ :** Time dependent constraint equation.

**FEEDCONEQ :** Constraint equations for feeding of elements.

For prescribed displacements the command has the following format:

#### 3.4.1 PDISP

**CONSTR PDISP PDTYPE NODID DOF ...**  
**[REPEAT N NODINC]**

**PDTYPE:** Type of prescribed displacement.

Allowed types are:

**LOCAL** : Apply prescribed displacement in local system of a node.

**GLOBAL** : Apply prescribed displacement in global system of a node.

**WAVE** : Create motion of the sea surface nodes linked to a certain wave load, see Section 3.55.

**RAO** : Apply motion in a node by using a RAO type response description which is linked to wave load, see Section 3.4.1.

**SPECIAL** : Create a rigid link between a slave node and a master node (e.g. vessel node).

Generally, the **REPEAT** command is used to repeat a sequence.

However, **REPEAT** is not allowed for the **RAO** and **SPECIAL** options. For all other options, the **REPEAT** card has the following format:

**N**: Total number of times the constraint is repeated.

**NODINC**: Nodal increment.

The other input data related to each option are defined below:

## LOCAL

For **CONSTR PDISP LOCAL** the format is as follows:

```
CONSTR PDISP LOCAL NODID DOF DISPVAL HISTNO
[REPEAT N NODINC]
```

**NODID**: Node ID.

**DOF**: DOF number (1-6).

**DISPVAL**: Prescribed displacement value (unit: L for DOF 1-3, R for DOF 4-6).

The displacement value is multiplied with the time dependent load factor given in time history specified by the **HISTNO** parameter.

**HISTNO**: History ID, specifying the time dependent load pattern. See Section 3.50.

## GLOBAL

For **CONSTR PDISP GLOBAL** the format is the same as for **LOCAL**:

```
CONSTR PDISP GLOBAL NODID DOF DISPVAL HISTNO
[REPEAT N NODINC]
```

**NODID**: Node ID.

**DOF:** DOF number (1-6).

**DISPVAL:** Prescribed displacement value (unit: L for DOF 1-3, R for DOF 4-6).

The displacement value is multiplied with the time dependent load factor given in time history specified by the **HISTNO** parameter.

**HISTNO:** History ID, specifying the time dependent load pattern. See Section 3.50.

#### EXAMPLE:

---

```
#      constrtype type    nodid  dof   value   histno
CONSTR PDISP      GLOBAL  1041   5    0.1745  200
```

---

## WAVE

For **CONSTR PDISP WAVE** the format is:

**CONSTR PDISP WAVE NODID DOF WAVNO**  
**[REPEAT N NODINC]**

where:

**NODID:** Node ID.

**DOF:** DOF number (1-3).

**WAVNO:** Wave ID number.

Note that **PDISP** is only applied for DOF 3. The other DOFs 1-2 of the sea surface nodes, need to be restrained by applying the **BONCON** command, see Section 3.2.

#### EXAMPLE:

---

```
#  

# Wave elevation:  

#-----  

#      Type  node   dof   waveno  

CONSTR PDISP WAVE  2101   3    100  

#          n   m  

REPEAT  121   1
```

---

## RAO

For **CONSTR PDISP RAO** the format is:

<b>CONSTR PDISP RAO NODID DOF VHEAD WAVENO RAONAME</b>
--

**NODID:** Node ID.

**DOF:** DOF number (1-6). *Note:* The RAO definition must be given for all dofs and in subsequent order 1-6, i.e. up to 6 command lines in the 3D case.

**VHEAD:** Vessel heading angle relative to global x-axis see Fig. 3.1 and Fig. 3.38, (unit: R).

**WAVENO:** Wave ID number, see Section 3.55.

**RAONAME:** Name of the RAO properties for the **DOF** defined by the **RAOPROP** and **READTRF** cards, see Section 3.44 and Section 3.45.

The **RAO** option introduces a vessel type coordinate transformation matrix for **NODID**. The heading angle **VHEAD** corresponds to the angle between the global x-axis and the local x-axis of **NODID**. It is possible to change the vessel static position and heading by **LOCAL** or **GLOBAL** prescribed displacements and then perform restart by the **RAO** option. Note that once the **RAO** option is introduced, the transformation matrix used to calculate prescribed displacements is not updated, in order for the RAO displacements to always be referred to the same system, see Section 3.44.

For the analysis types **AUTOSTART**, **TIMEINIT TIMEINITSDD**, **INIJLAY**, **IN-ISLAY** or **SDDSTART**, see Section 3.6, SIMLA will compute the heading angle and overrule the one defined by **VHEAD** above. The computed heading angle is printed to screen and the log file at start of the analysis run. However, if the analysis is continued by a **RESTART** analysis run, the user must set the heading angle **VHEAD** equal to the heading angle at the end of the previous analysis run.

For verification purposes, the following data will be printed to screen and the log file at the time step when the wave kinematics is generated:

- **NODID** of the RAO node
- Wave direction relative to global x-axis
- Angle between vessel node x-axis and global x-axis
- Vessel heading angle **VHEAD** relative to global x-axis
- RAO heading angle

**EXAMPLE:** \_\_\_\_\_

```
#          slavnod  dof heading wave no name of function
CONSTR PDISP RAO    3001     1      0           100 surge
CONSTR PDISP RAO    3001     2      0           100 sway
```

```

CONSTR PDISP RAO    3001    3    0          100  heave
CONSTR PDISP RAO    3001    4    0          100  roll
CONSTR PDISP RAO    3001    5    0          100  pitch
CONSTR PDISP RAO    3001    6    0          100  yaw
#

```

---

## SPECIAL

For **CONSTR PDISP SPECIAL** the format is:

**CONSTR PDISP SPECIAL NODID DOF MNOD XANG YANG ZANG XECC  
YECC ZECC**

where

**NODID**: Node ID.

**DOF**: DOF number (1-6).

**MNOD**: Master node ID number to which the prescribed displacement is referred.

**XANG**: Tait–Bryan angle for x-rotation (local system) applied to define the orientation of the applied prescribed displacement relative to the local system of the master node.

**YANG**: Tait–Bryan angle for y-rotation (local system) applied to define the orientation of the applied prescribed displacement relative to the local system of the master node.

**ZANG**: Tait–Bryan angle for z-rotation (local system) applied to define the orientation of the applied prescribed displacement relative to the local system of the master node.

**XECC**: Nodal x-eccentricity relative to the local master node system.

**YECC**: Nodal y-eccentricity relative to the local master node system.

**ZECC**: Nodal z-eccentricity relative to the local master node system.

## EXAMPLE:

---

```

#           sn   dof  mn   fi1 fi2 fi3 ex     ey  ez
CONSTR PDISP SPECIAL 441 1   3001 0   0.0 0   71.56 0  25.000
CONSTR PDISP SPECIAL 441 2   3001 0   0.0 0   71.56 0  25.000
CONSTR PDISP SPECIAL 441 3   3001 0   0.0 0   71.56 0  25.000

```

---

### 3.4.2 CONEQ

A linear constraint equation has the general form:  $r_{sl} = C_0 + C_1r_{m1} + C_2r_{m2} + C_3r_{m3}$ , where *sl* denotes slave and *mi* denotes the masters.

In the SIMLA-input a constant constraint equation with **CONEQ** option is specified by the following format:

```
CONSTR CONEQ PDTYPE ...
[REPEAT ....]
```

where:

**PDTYPE:** Type of constraint equation.

Allowed types are:

**LOCAL** : Apply constraints in local system of the slave and master nodes.

**LOCSL** : Apply constraints in local system of the slave node as reference.

**GLOBAL** : Apply constraints in global system of a node.

The other input data related to each option are defined below:

#### LOCAL

For **CONSTR CONEQ LOCAL** the format is as follows:

```
CONSTR CONEQ LOCAL SLNOD SLDOF C0 MNOD1 MDOF1 C1 ...
[REPEAT N SLAVEINC MASTINC]
```

**SLNOD:** Node ID of the prescribed (slave) node.

**SLDOF:** DOF number.

**C0:** Constant displacement ( $C_0$ ).

**MNOD1:** ID for the master node.

**MDOF1:** Master DOF number.

**C1:** Constraint coefficient for the master number 1 ( $C_1$ ).

The last parameters **MNOD1**, **MDOF1**, **C1** may be repeated k (number of masters) times.

The **REPEAT** card is described as follows:

**N:** Total number of times the constraint is repeated.

**SLAVEINC:** Nodal increment for slave.

**MASTINC:** Nodal increment for master(s).

## LOCSL

For **CONSTR CONEQ LOCSL** the format is as follows:

```
CONSTR CONEQ LOCSL SLNOD SLDOF C0 MNOD1 MDOF1 C1 ...
[REPEAT N SLAVEINC MASTINC]
```

**SLNOD**: Node ID of the prescribed (slave) node.

**SLDOF**: DOF number.

**C0**: Constant displacement ( $C_0$ ).

**MNOD1**: ID for the master node.

**MDOF1**: Master DOF number.

**C1**: Constraint coefficient for the master number 1 ( $C_1$ ).

The last parameters **MNOD1**, **MDOF1**, **C1** may be repeated k (number of masters) times.

The **REPEAT** card is described as follows:

**N**: Total number of times the constraint is repeated.

**SLAVEINC**: Nodal increment for slave.

**MASTINC**: Nodal increment for master(s).

## GLOBAL

For **CONSTR CONEQ GLOBAL** the format is as follows:

```
CONSTR CONEQ GLOBAL SLNOD SLDOF C0 MNOD1 MDOF1 C1 ...
[REPEAT N SLAVEINC MASTINC]
```

**SLNOD**: Node ID of the prescribed (slave) node.

**SLDOF**: DOF number.

**C0**: Constant displacement ( $C_0$ ).

**MNOD1**: ID for the master node.

**MDOF1**: Master DOF number.

**C1**: Constraint coefficient for the master number 1 ( $C_1$ ).

The last parameters **MNOD1**, **MDOF1**, **C1** may be repeated k (number of masters) times.

The **REPEAT** card is described as follows:

**N**: Total number of times the constraint is repeated.

**SLAVEINC**: Nodal increment for slave.

**MASTINC:** Nodal increment for master(s).

If all the constraint coefficients except for **C0** are zero, the **CONEQ** definition is equivalent to a **PDISP** definition. If all constraints coefficient except for **C1** are zero, the **MDOF1** of **MNOD1** completely defines the behaviour of the **SLDOF** of **SLNOD**.

EXAMPLE:

---

#	ctype	type	slavnod	slavdof	disp	masternod	masterdof	constrcoeff
CONSTR	CONEQ	GLOBAL	1	1	0	1001	1	1.0
CONSTR	CONEQ	GLOBAL	1	2	0	1001	2	1.0
CONSTR	CONEQ	GLOBAL	1	3	0	1001	3	1.0

---

### 3.4.3 HISTCONEQ

A constraint equation with time dependent coefficients is specified by the following format:

```
CONSTR HISTCONEQ PDTYP SLNOD C0 THIST0 MNOD1 MDOF1 C1 THIST1
...
[REPEAT N SLAVEINC MASTINC]
```

where:

**PDTYP:** Type of constraint, can have values **LOCAL** or **GLOBAL**.

**SLNOD:** Node ID of the prescribed (slave) node.

**SLDOF:** Slave DOF number.

**C0:** Constant displacement ( $C_0$ ) The dispacement value is multiplied with the time dependent load factor given in time history specified by the **THIST0** parameter.

**THIST0:** Time history number for  $C_0$ , refering to the definition of the time dependent load factor, see Section 3.50.

**MNOD:** ID for the master node.

**MDOF:** Master DOF number.

**C1:** Constraint coefficient for the master number 1 ( $C_1$ ). The constrain coefficient value is multiplied with the time dependent load factor given in time history specified by the **THIST1** parameter.

**THIST1:** Time history number for  $C_1$ , refering to the definition of the time dependent load factor, see Section 3.50.

The sequence **MNOD1 MDOF1 C1 THIST1** can be repeated k (the number of masters) times.

The **REPEAT** card is described as follows:

**N:** Total number of time the constraint is repeated.

**SLAVEINC:** Nodal increment for slave.

**MASTINC:** Nodal increment for master(s).

EXAMPLE: \_\_\_\_\_

```
#  
# Constraint equations between vessel and riser:  
#-----  
# sl   dof  C0    thist  ma   dof  C1    thist  
CONSTR HISTCONEQ GLOBAL  101   3    0.0   100    55   3    1.0   200
```

---

### 3.4.4 FEEDCONEQ

Constraint equations defining the constraint equations needed when a FEED-command is given has the following format:

```
CONSTR FEEDCONEQ SLNOD MNOD1 MNOD2 MNOD3  
[REPEAT N SLAVINC MASTINC]
```

where:

**SLNOD:** Slave node ID.

**MNOD1:** Master node ID under feeding (end 1 of the guide element, see **FEED**-command, Section 3.26). This must be the first node in a element which has the other end connected to the vessel by constraint equations.

**MNOD2:** Master node ID after feeding (end 2 of subsequent feed-element).

**MNOD3:** Master node ID for storage of elements before feeding. This should be oriented with the local x-axis of the node is pointing upwards. The stack of stored elements should be directed the same way.

The **REPEAT** card is described as follows:

**N:** Total number of time the constraint is repeated.

**SLAVEINC:** Nodal increment for slave.

**MASTINC:** Nodal increment for master, mnod2.

EXAMPLE: \_\_\_\_\_

```
#          slave  m1      m2      mf  
CONSTR FEEDCONEQ  5101   5012    5210   5003  
CONSTR FEEDCONEQ  5102   5012    5201   5003
```

#

### 3.5 CONTINT - contact interfaces

In order to optimize the contact search, the contact interfaces need to be defined for all CONTACT, and SEA type interfaces. The following format is applied.

```
CONTINT GRPNAME MASTERNAMESPACE SLAVENAMEI [IS1 ISN TX TY TZ MAXIT  
IGAP]
```

where

**GRPNAME**: Name of the contact element group.

**MASTERNAMESPACE**: Name of the master element group.

**SLAVENAMEI**: Name of the slave element group or the name of a contact surface, see the **COSURFPR** command. If the element type of the contact element group is SEA150, then the master group shall have the same name as the sea group, and the slave group shall have the name of the structural element group that may have sea contact. For the SEA150 element type, no more parameters are needed. For structural elements or contact surfaces, additional parameters need to be given, see below.

**IS1**: First slave element in contact range. Dummy for HCONT and CONT152 elements.

**ISN**: Last slave element in contact range. Dummy for HCONT and CONT152 elements.

Search for contact is carried out between **IS1** and **ISN**.

**TX**: Time at which the contact elements are to be activated in the local x direction.

**TY**: Time at which the contact elements are to be activated in the local y direction.

**TZ**: Time at which the contact elements are to be activated in the local z direction.

**MAXIT**: Gap iteration parameter which gives the maximum number of iterations to be performed in order to find whether there is a gap or not. If the number of iterations exceeds this number, gap is assumed for the remaining iterations. For CONT125 and CONT126 reasonable values are 4-6. For the other ones 20-60 are reasonable values. Dummy for CONT153.

**IGAP**: Control parameter used to control contact element features depending on contact element type. Dummy for elements CONT125, CONT128 and CONT153.

For CONT124 and CONT164 the appropriate values are **IGAP  $\geq 0$**  or **IGAP  $< 0$** .

If **IGAP  $\geq 0$**  then the direction of the normal vector is set once at the start of the analysis. If **IGAP  $< 0$**  then the direction of the normal vector will be updated during the simulation as long as the distance between the master roller and the slave pipe is greater than zero.

For CONT126, **IGAP=2** means that the  $x$ -moment is set to zero and that the second term on the right-hand side of Eq. (3.39) is set to zero.

For CONT130, CONT170 and CONT152 **IGAP = 0** means that the feature is turned off.

For CONT130 and CONT170, **IGAP > 10** means that normal vector spline interpolation is applied in the longitudinal direction to obtain  $C^1$ -continuity between straight geometry segments. Spline interpolation gives improved accuracy for the axial force distribution along the slave system if the master is conical, eg. bell-mouth (INSIDE = -1).

For CONT130, **IGAP = 2** or **12** means that the torsion moment due to friction in the circumferential direction is not included. For **IGAP = 12**, the torsion moment is not included while normal vector spline interpolation is applied in the longitudinal direction of the tube geometry.

For CONT152, **IGAP < 0** means that the meaning of igap transforms into time at which a new contact search is performed. For the CONT152 element the contact search is performed initially. If large motions occur this may lead to a large number of contact elements to ensure that contact is obtained at the right location. By this, the user can suppress a new contact search to update the candidate contact points.

#### EXAMPLE:

---

```
#  
# Contact interface data:  
#-----  
# grpname mastername slavename is1 isn tx ty tz maxit igap  
CONTINT seabed ormpipe1 cosurf1 1 201 1. 10000 1 6 1  
CONTINT ormcontact ormcontact ormpipe2 341 441 10000 10000 1 50 1  
CONTINT ormcontact1 ormcontact1 ormpipe2 341 441 10000 10000 1 50 1  
CONTINT ormcontact2 ormcontact2 ormpipe2 341 441 10000 10000 1 50 1  
CONTINT ormcontact3 ormcontact3 ormpipe2 341 441 10000 10000 1 50 1  
CONTINT ormcontact4 ormcontact4 ormpipe2 341 441 10000 10000 1 50 1  
CONTINT ormcontact5 ormcontact5 ormpipe2 341 441 10000 10000 1 50 1  
CONTINT ormcontact6 ormcontact6 ormpipe2 341 441 10000 10000 1 50 1  
CONTINT ormcontact7 ormcontact7 ormpipe2 341 441 10000 10000 1 50 1  
CONTINT ormcontact8 ormcontact8 ormpipe2 341 441 10000 10000 1 50 1  
CONTINT ormcontact9 ormcontact9 ormpipe2 341 441 10000 10000 1 50 1  
CONTINT ormcontact10 ormcontact10 ormpipe2 341 441 10000 10000 1 50 1  
CONTINT ormcontact11 ormcontact11 ormpipe2 341 441 10000 10000 1 50 1  
CONTINT ormcontact12 ormcontact12 ormpipe2 241 441 10000 10000 1 50 1  
CONTINT seal seal ormpipe1
```

---

### 3.6 CONTROL - CONTROL parameters

The control parameters are defined using one data card starting with **CONTROL** and in the following format:

**CONTROL MAXIT NDIM ISOLVR NPOINT IPRINT CONR GAC ISTRES ...**

where:

**MAXIT**: Maximum number of equilibrium iterations. This setting can be overwritten in the **TIMECO** card.

**NDIM**: Dimension of analysis:

**2** : 2-dimensional. This option was discontinued as of SIMLA version 3.17.

**3** : 3-dimensional.

**ISOLVR**: Equation solver parameter;

**1** : gives skyline solver of the equation system.

**2** : gives sparse solver, which is most efficient.

**NPOINT**: Number of integration points around the cross section. This is used both for the pipe non-linear material elements and for the visual model meshing.

**IPRINT**: Print parameter One or two digits can be applied to set print of model information to output file (.sof), and print of timestep and iterations to log file (.slf) and print to screen. Note that printing to the log file can be time-consuming.

**0** : turn off print to output file, log file and screen

**00** : turn off print to output file, log file and screen

**01** : turn off print to output file, turn on print to log file and screen. Using **1** will also toggle this behaviour

**02** : turn off print to output file, print only time steps and no information of convergence of iterations to log file and screen

**03** : turn off print to output file, print time steps and information of the converged iterations (last iteration) to log file and screen

**10** : turn on print to output file, turn off print to log file and screen

**11** : turn on print to output file, turn on print to log file and screen

**12** : turn on print to output file, print only time steps and no information of convergence of iterations to log file and screen

**13** : turn on print to output file, print time steps and information of the converged iterations (last iteration) to log file and screen

**CONR**: Convergence norm. The recommended value is  $10^{-5}$ - $10^{-7}$ .

**GAC**: Acceleration of gravity (unit:  $LT^{-2}$ )

**ISTRES**: Start procedure parameter which may have the following values: **STRESSFREE**, **RESTART**, **EIGEN**, **AUTOSTART**, **FILEINIT**, **TIMEINIT**, **INIJLAY**, **INISLAY**, **SDDSTART** or **TIMEINITSDD**. **STRESSFREE** means that the initial configuration is stressfree. **RESTART** means restart from a previous analysis. **EIGEN** means eigenvalue analysis for the free undamped oscillation problem at a load step from a previous analysis. **FILEINIT**, **TIMEINIT**, **INIJLAY**, **INISLAY** and **TIMEINITSDD** assume that the initial configuration in terms of nodal coordinates and nodal transformation matrix is stored on file and this info is used to initiate the stress in the elements at the first load step. The SIMPOST post processor can generate a file on the format required for **FILEINIT**, see Section 4.10. In static laying analysis (**STATIC-SIMLA** or **STATIC-SDD**), see Section 3.51 the pipe configuration is written to a binary file every time step during the analysis and can be retrieved for dynamic analysis at critical locations along the route by a start using **TIMEINIT**, **INIJLAY**, **INISLAY** or **TIMEINITSDD**.

The following parameters depend on the value given for **ISTRES**.

### 3.6.1 STRESSFREE

If **ISTRES** = **STRESSFREE**, no more parameters are required.

### 3.6.2 RESTART

If **ISTRES** = **RESTART** then one more parameter is needed:

```
CONTROL MAXIT NDIM ISOLVR NPOINT IPRINT CONR GAC RESTART
IRESTP
```

**IRESTP**: Restart load step.

### 3.6.3 EIGEN

If **ISTRES** = **EIGEN** then two more parameters are needed:

```
CONTROL MAXIT NDIM ISOLVR NPOINT IPRINT CONR GAC EIGEN IRESTP
NEVAL
```

**IRESTP**: Load step at which the eigenvalues are wanted.

**NEVAL**: Number of eigenvalues wanted. If a negative number is given, the eigenvalues are animated from irestp on the raf file

The computed eigenvalues and eigenmodes are based on the undamped free vibration problem including the mass and global stiffness matrices available at the load step requested.

The load step **IRESTP** must be a step available at an existing raf-file with the same prefix name as the input file. The eigenvalues and the eigenvectors are printed to the output file with suffix .sof , where each eigenvector is scaled to maximum one in translation. This also includes the the associated axial strain, curvature and torsion obtained for the same scaled to one eigenvector.

Animation of the eigenmodes starting with the lowest mode is also available on the raf-file after the analysis if a negative **NEVAL** is given. The animated eigenvectors can also be written out during postprocessing, see Section 4.6 applying the visual model TNDISP option.

For the time being, the eigenvalue analysis requires **ISOLVR=1**.

The number of accepted eigenvalues and the relative error for each eigenvalue are printed to the log file with suffix .slf . The maximum number of eigenvalue iterations is for the time being set equal to twice the value of **NEVAL**. Hence, if the number of accepted eigenvalues is less than the number of eigenvalues of interest, **NEVAL** must be increased such that the number of acceptable eigenvalues becomes large enough.

### 3.6.4 FILEINIT

For **ISTRES = FILEINIT**, the card takes the following form:

```
CONTROL MAXIT NDIM ISOLVR NPOINT IPRINT CONR GAC FILEINIT
COFILE
```

**COFILE**: Coordinate/transformation matrix file. The file is to contain a consecutive number of lines containing on each line: node ID number, 3 coordinates ( $x, y, z$ ) and 9 components of the node transformation matrix. At the start of the file one extra line is added containing first element ID number, element increment (positive or negative), first node number, node increment (positive or negative).

### 3.6.5 TIMEINIT

**ISTRES = TIMEINIT** gives initiation of pipe configuration from static laying analysis (STATIC-SIMLA), and needs the following parameters:

```
CONTROL MAXIT NDIM ISOLVR NPOINT IPRINT CONR GAC TIMEINIT IFILE
TSLINI PIPEGRP VESSELGRP SEAGRP TENSIONERGRP TCONGRP
```

**IFILE**: Name of input file for static laying analysis.

**TSLINI**: Time for relevant configuration.

**PIPEGRP**: Pipe group name.

**VESSELGRP:** Vessel group name.

**SEAGRP:** Sea group name.

**TENSIONERGRP:** Tensioner group name, can be **NONE**. This parameter can be used in some analysis of type **SIMLA-FEED** given in **TIMECO**, see Section 3.51.

**TCONGRP:** T-connection group name, can have value **NONE**. This parameter can be used in some analysis of type **SIMLA-FEED** given in **TIMECO**, see Section 3.51.

*Note:* This option requires that the pipe group in the model is identical to the one used in the static laying analysis. Other groups can be changed, added or removed.

EXAMPLE: \_\_\_\_\_

```
#      maxit    ndim   isolvr   npoint   ipri    conr   gacc   iproc
CONTROL 100      3       1        8       1     1e-4   9.81  timeinit
#
#      file          time   pipegrp   vesselgrp   seagrp
#           "megb-lay-test"  8.0    megpipe    vessel1     megsea
#
#      tensionergrp   tcongrp
#           none         none
```

---

### 3.6.6 INIJLAY

**ISTRES = INIJLAY** gives initiation of pipe configuration from static J-lay analysis (STATIC-SIMLA), and needs the following parameters:

<b>CONTROL MAXIT NDIM ISOLVR NPOINT IPRINT CONR GAC INIJLAY INITYP</b>
<b>IFILE TSLINI NPIPE VESSELGRP SEAGRP TENSIONERGRP PIPEGRP</b>

**INITYP:** Type of start, currently file is the only allowable value.

**IFILE:** Name of input file for static laying analysis.

**TSLINI:** Time for relevant configuration.

**NPIPE:** Number of pipe groups.

**VESSELGRP:** Vessel group name.

**SEAGRP:** Sea group name.

**TENSIONERGRP:** Tensioner group name.

**PIPEGRP:** Pipe group name.

*Note:* This option requires that the pipe group in the model is identical to the one used in the static laying analysis. Other groups can be changed, added or removed.

### 3.6.7 INISLAY

**ISTRES = INISLAY** gives initiation of pipe configuration from static S-lay analysis (STATIC-SIMLA), and needs the following parameters:

```
CONTROL MAXIT NDIM ISOLVR NPOINT IPRINT CONR GAC INISLAY INITYP
IFILE TSLINI NPIPE VESSELGRP SEAGRP TENSIONERGRP PIPEGRP
```

**INITYP**: Type of start, currently file is the only allowable value.

**IFILE**: Name of input file for static laying analysis.

**TSLINI**: Time for relevant configuration.

**NPIPE**: Number of pipe groups.

**VESSELGRP**: Vessel group name.

**SEAGRP**: Sea group name.

**TENSIONERGRP**: Tensioner group name.

**PIPEGRP**: Pipe group name.

*Note*: This option requires that the pipe group in the model is identical to the one used in the static laying analysis. Other groups can be changed, added or removed.

### 3.6.8 TIMEINITSDD

**ISTRES = TIMEINITSDD** gives initiation of pipe configuration from static laying analysis with SDD (STATIC-SDD), and needs the following parameters:

```
CONTROL MAXIT NDIM ISOLVR NPOINT IPRINT CONRGAC TIMEINITSDD
IFILE TSLINI LSEABED DLSDD DLTOP SDDRANGE ELM1 EML2 RATE
SDDNODE [SDDNODE2 SDDNODE3] SEAGRP VESSELGRP1 .. VESSELGRPN
UMBGRP1 .. UMBGRPN CABLEGRP1 [CABLEGRP2 CABLEGRP3] SDDGRP1 ..
SDDGRPN [ORHEAD]
```

**IFILE**: Name of input file for static laying analysis.

**TSLINI**: Time for relevant configuration.

**LSEABED**: Length of pipe resting on seabed (behind tdp).

**DLSDD**: Element lengths for umbilical around sdd.

**DLTOP**: Element length in top of umbilical.

**SDDRANGE**: Range around sdd for which **DLSDD** should be used.

**ELM1**: Max. element length towards umbilical tail.

**ELM2**: Max. element length in umbilical region between sdd and top.

**RATE**: Rate to change element lengths, recommended 0.1 ie. 10%.

**SDDNODE**: SDD center node.

**SDDNODE2:** Node of stub 1 (Optional).

**SDDNODE3:** Node of stub 2 (Optional).

**SEAGRP:** Sea group ID, only one allowed.

**VESSELGRP1:** First vessel group, starting with prefix VESSEL.

**VESSELGRPN:** Last vessel group, starting with prefix VESSEL.

**UMBGRP1:** Current model first umbilical group ID starting with the prefix UMB.

**UMBGRPN:** Current model last umbilical group ID starting with the prefix UMB.

**CABLEGRP1:** Current model cable group ID starting with the prefix CAB.

**CABLEGRP2:** Current model cable group ID, connects to stub 1, starting with the prefix CAB (Optional, must be given if **SDDNODE2** has been given).

**CABLEGRP3:** Current model cable group ID, connects to stub 2, starting with the prefix CAB (Optional, must be given if **SDDNODE3** has been given).

**SDDGRP1:** Current model first SDD group ID starting with the prefix SDD.

**SDDGRPN:** Current model last SDD group ID starting with the prefix SDD.

**ORHEAD:** Overrule heading, optional. If given, the vessel head angle will be overruled by this value.

#### EXAMPLE:

---

```
#      maxit ndim isolvrv npoint ipri conr gacc iproc
CONTROL 100    3     1      8      1   1e-4 9.81 timeinitsdd
#
#  file      time lseabed dlsdd dltop sddrange elmax1 elmax2 rate sddnodes
"test1"  8.0  70.0    0.35  0.4    15       0.7    5.0    0.1  13001 10001 10003
#
#  seagroup vesselgrp1 vesselgrp2 umbgrp1
  ormsea    vessel1    vessel2    umbilical
#
#  umbgrp2      cablegrp1    cablegrp2    cablegrp3    sddgrp1    sddgrp2
  umbilical2    cables1      cables2      cable        sddstub    sdd
```

---

### 3.6.9 AUTOSTART

If **ISTRES** = **AUTOSTART** then an initial configuration is determined by SIMLA automatically. It assumes that the pipeline is installed on a seabed defined by the **CO-SURFPR** card, see Section 3.8. It then finds the initial configuration by identifying candidate seabed contact points. A catenary section is included depending on lay vessel parameters. Both J-lay and S-lay is allowed. Note that the catenary section option **ICATEN** may be set to a non-positive integer which means that SIMLA assumes the

whole pipeline to be installed on seabed. The effective tension compatible with the initial configuration is computed by the following expression for submerged pipe elements:

$$T = \frac{w_s D}{\frac{1}{\cos \theta} - 1} \quad (3.1)$$

where  $D$  is the water column height above the pipe element,  $w_s$  is the pipe submerged weight and the departure angle is denoted  $\theta$ . The departure angle must therefore be defined also for the case when the whole pipeline is located on the seabed (**ICATEN** $\leq 0$ ). An external axial force with magnitude close to Eq. (3.1) should be applied at the end of the pipeline, otherwise the initial configuration may deviate too much from the static equilibrium configuration resulting in convergence difficulties at the first load step. Further, buoyancy and gravity loads are recommended to apply with full magnitude from time  $t = 0.0$ .

```
CONTROL MAXIT NDIM ISOLVR NPOINT IPRINT CONR GAC AUTOSTART
IN1PIP IN2PIP INCPIP NROLLS ICATEN IVSNOD TB DEPAN FREEB RAMPAN
RAMPLE STIRAD KPTDP0 SEABDGRP STINGERGRP VESSELGRP INOCOG
```

The associated parameters are defined as follows:

**IN1PIP**: The first pipe element ID number at the start of the catenary.

**IN2PIP**: The last pipe element ID number at the vessel end.

**INCPIP**: The node increment along the element segment (normally 1).

**NROLLS**: Number of roller stations along stinger.

**ICATEN**: Catenary parameter..

- 2** : then the catenary section is included, the seabed section is installed on seabed assuming small bending stiffness.
- 1** : then the catenary section is included, the seabed section is installed on seabed assuming steel pipe bending stiffness.
- 0** : no catenary section is included, the pipe is installed on seabed assuming steel pipe bending stiffness.
- 1** : no catenary section is included, the pipe section is installed on seabed assuming small bending stiffness.

Note that when setting **ICATEN=0**, or **ICATEN=-1** the pipe will be installed on seabed and SIMLA will carry out a free-span analysis positioning last end of pipe at **KPTDP0**, see Fig. 3.3.

**IVSNOD**: Vessel pipe node ID (if relevant), i.e. the pipe node where the pipe is terminated in the tensioner. In order to move associated vessel model nodes the same quantity as the pipe node in order to reach the initial configuration it is necessary to give the vessel pipe node ID as input, see Fig. 3.4.

**TB:** Bottom tension (dummy) (unit: F).

**DEPAN:** Vessel departure angle (unit: R).

**FREEB:** Vessel freeboard (unit L). See Fig. 3.4

**RAMPAN:** Vessel ramp angle (unit: L).

**RAMPLE:** Vessel ramp length (unit: L).

**STIRAD:** Vessel stinger radius R (unit: L).

**KPTDP0:** Required initial KP value for TDP. Must be larger than 0.0. If a negative value is supplied, the configuration is directed in oposite direction, such that laying is performed with decreasing KP value. This setting also takes effect for possible continuation fo the analysis with **ISTRESS** value **TIMEINIT**.

**SEABDGRP:** Name of seabed element group.

**STINGERGRP:** Name of stinger contact element group (may be **NONE** for e.g. J-lay).

**VESSELGRP:** Name of vessel group (may be **NONE**). This parameter is used to identify whether there is a vessel and which elements that are used to represent the vessel. If not **NONE**, all nodes that are connected to element groups named 'vessel' for the first six characters will be translated the same distance as the pipe vessel node **IVSNOD**. The reference coordinate system applied is the sea bed coordinates taken from the seabed coordinate text file, see Section 3.8. In many cases, teese coordinates are based on sea bed surveys and may have large numric values, which may cause numerical difficulties. This may be solved by moving the origo of the sea bed survey data, see Section 3.8.

**INOCOG:** Node ID number for vessel COG, see Fig. 3.4. This parameter may be ommitted, however to ensure that wave loads are generated from the vessel COG at time  $t = 0$ , the vessel COG node has to be given. It is therefore recommended to include this parameter for dynamic analyses.



Figure 3.3: Position of pipe in free span analysis (**ICATEN** = 0).

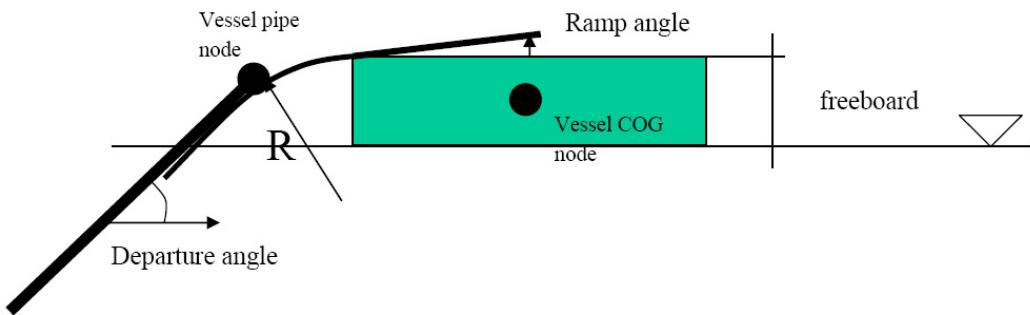


Figure 3.4: Vessel definition.

EXAMPLE:

---

```

CONTROL 120 3 1 16 1 1e-5 9.81 autostart
#
#      ie1pip ie2pip incpip nrolls icaten ivsnod
#          1       440     1      16      1       441
#
#      tens0 depang freeb rampan rample stirad kp
#          0      1.244   10.0   0.1745  0      120     1
#
#      seabedgrp stingergrp vesselgrp
#      seabed      ormcontact5 vessel1
#

```

---

### 3.6.10 SDDSTART

**ISTRES = SDDSTART** is the option for automatic start with a SDD configuration. This configuration puts restrictions on how the FEM model must be defined.

Two sections of pipe/umbilical must be defined. The first one will be the tail, reaching to the SDD location. This should be defined such that node number are increasing in direction of the vessel.

The second pipe/umbilical must have double nodes, and be connected by **FEEDCONEQ** on a **CONSTR** card. The first node must be at the same location as the last node in umbilical number one, and be connected to this by normal constraint equations **CONSTR**. Eventually, umbilical number two will be located from the SDD to the vessel. Leftover elements will be stored on the vessel, and must have a node on the vessel where they can be attached. Elements on the two umbilicals must be consecutively numbered, from tail to top.

A wire (must be pipe elements) must be defined, also with double nodes and by the use

of **FEEDCONEQ** constraints. The first node of the wire must be attached to the SDD by constraint equations. Elements in the wire must be consecutively numbered from the SDD and upwards. The SDD is defined as a pipe, bellmouth shape can be obtained by use of **NODPROP** card. Contact elements between SDD and both umbilicals must be defined. Wire will be positioned from the SDD to the vessel, with leftover elements attached to a storage node on the vessel, as for umbilical number two.

The vessel must have at least one node with global constraints for storage of elements. In addition, a feeding guide element must be defined for both umbilical and wire. They must be attached to two different nodes on the vessel by constraint equations.

The format of the **CONTROL** card is as follows:

```
CONTROL MAXIT NDIM ISOLVR NPOINT IPRINT CONR GAC SDDSTART
SEABEDGRP SDDGRP WIREGRP VESSELGRP IEL1 IEL2 SDDNODE PIPEVNO1
PIPEVNO2 WIREVNO1 WIREVNO2 HINI TDP0 FREEB TENS0 IVCOG [LAY-
BACK]
```

**SEABEDGRP**: Name of seabed group.

**SDDGRP**: Name of SDD group.

**WIREGRP**: Name of wire group.

**VESSELGRP**: Name of vessel group.

**IEL1**: First element in umbilical number one.

**IEL2**: Last element in umbilical number two.

**SDDNODE**: Node on pipe where SDD will be centered. (Must be the last node of umbilical number 1.)

**PIPEVNO1**: Node on vessel where pipe guide element is attached. Must be different from **WIREVNO1**.

**PIPEVNO2**: Node on vessel where pipe elements are to be stored. This must be consistent with the node given in the **FEEDCONEQ** constraints in the **CONSTR** card. Can be the same as **WIRENO2**.

**WIREVNO1**: Node on vessel where wire guide element is attached. Must be different from **PIPEVNO1**.

**WIREVNO2**: Node on vessel where wire elements are to be stored. This must be consistent with the node given in the **FEEDCONEQ** constraints in the **CONSTR** card. Can be the same as **PIPEVNO2**.

**HINI**: Initial height over seabed for the SDD.

**TDP0**: Required initial KP value for TDP.

**FREEB**: Free board of the umbilical.

**TENS0**: Required tension in umbilical at touch down point.

- IVCOG:** Vessel node, where a **CLOAD** with its own **THIST** shhould be connected if steering after tension. Note that the vessel should not have **ELECC** to this node.
- LAYBACK:** Optional parameter. If given, the specified **LAYBACK** is used for initialization instead of tension at touch down, **TENSO**.

### 3.7 COSUPR Contact surface material properties

The **COSUPR** command defines the material properties along a route/line on kilometer point (KP) basis. The command is required if a **COSURFPR** is defined. The following format is applied:

```
COSUPR MLINEID KP1 KP2 MNAME
```

```
.. .. ..
```

where:

- MLINEID:** Identity of the material line.
- KP1:** Starting point of material properties (KP value).
- KP2:** End point of material properties (KP value).
- MNAME:** Name of material for this KP range.

An arbitrary number of material sequences may be given.

The last parameters **KP1 KP2 MNAME** can be repeated as many times as needed to describe a varying material along the route. Note that **KP1** on a line must be the same as **KP2** of the preceeding line to ensure material to be defined at all locations.

The material line must be defined for the route section where contact between pipe and the contact surface may take place. The referenced material name must be defined with the **MATERIAL** command, see Section 3.37.

EXAMPLE:

---

#	route id	kp1	kp2	matname
COSUPR	100	-0.1	60000	soil1

---

### 3.8 COSURFPR - Contact surface properties

The **COSURFPR** command allows the user to define the contact surface properties, relative to the curvilinear position along the contact surface referred to as KP-points. The curvilinear length is measured on the route projected onto the xy-plane. The following format is applied:

**COSURFPR CONAME COFILE NLINES KP0 XSTART YSTART ANGSTART  
MLINEID [ .. MLINEIDN] [IDLINE]**

where:

**CONAME**: Name of contact surface.

**COFILE**: Name of the ASCII datafile containing the seabed geometry description.

**NLINES**: Number of seabed lines in contact surface file. Must be an odd number 1,3,5 etc. A positive number of lines means that the data is on the standard format in Section 3.8.1. A negative number of lines means that the data is on the survey file format in Section 3.8.2.

**KP0**: KP-value for 1. point in contact surface datafile, i.e. the KP-value for the first data line in **COFILE**.

**XSTART**: x-coordinate at start of contact surface relative to the global coordinate system (unit: L).

**YSTART**: y-coordinate at start of contact surface relative to the global coordinate system (unit: L).

**ANGSTART**: Angular orientation of contact surface relative to the global coordinate system (unit: R).

**MLINEID**: Material ID of the route line, i.e. the center line if several lines are given.

**MLINEIDN**: Material ID of additional seabed lines. To be able to apply several seabed lines, they must also be defined in the contact surface file.

**IDLINE**: Optional. A number between 1 and **NLINE**, specifying that this seabed line is the only one to be used.

The first seabed line is the center line coinciding with the route line, and any succeeding seabed lines must alternate between the left and the right side of the center line as shown in Fig. 3.6. The pipeline will be placed along the seabed center line when using the AUTOSTART option in Section 3.6.9.

The material ID must be given for each of the seabed lines, but the same ID may be repeated for several lines. The material ID describes the material properties of the seabed line on a KP basis, and is defined by the **COSUPR** command, see Section 3.7.

The seabed lines should preferably be close to parallel with the pipeline. This is because the KP-value applied for determining the seabed material properties is computed based on the minimum distance between the pipeline node and the seabed lines.

Large coordinate input values should be avoided as they may give round-off errors in arithmetic operations. In such cases, it is beneficial to move and/or rotate the seabed coordinate system to reduce the coordinate values. This may be done by applying the **XSTART**, **YSTART** and **ANGSTART** parameters.

EXAMPLE:

---

```
#      name   data file      nline kpstart    x0      y0      fi      route id
COSURFPR cosurf1 "seabed.txt"  1       0        0        0        0        100
```

---

### 3.8.1 Standard route file

For a route file described by only a single seabed line, the format is as follows for each data line in **COFILE**:

x y z nx ny nz

repeated for as many points there are in the route. x, y and z are the coordinates of the point and nx, ny, nz are the global components of the sea bottom normal vector at the point.

For a route file with several seabed lines, the x y z nx ny nz sequence is repeated for each seabed line in **COFILE** as shown in Fig. 3.5. The first sequence must be the center line, and should be followed by pairs of seabed lines located at each side of the center line, starting with the innermost pair, see Fig. 3.6. Note that SIMLA computes the KP-value based on the seabed center line, also for the additional seabed lines.

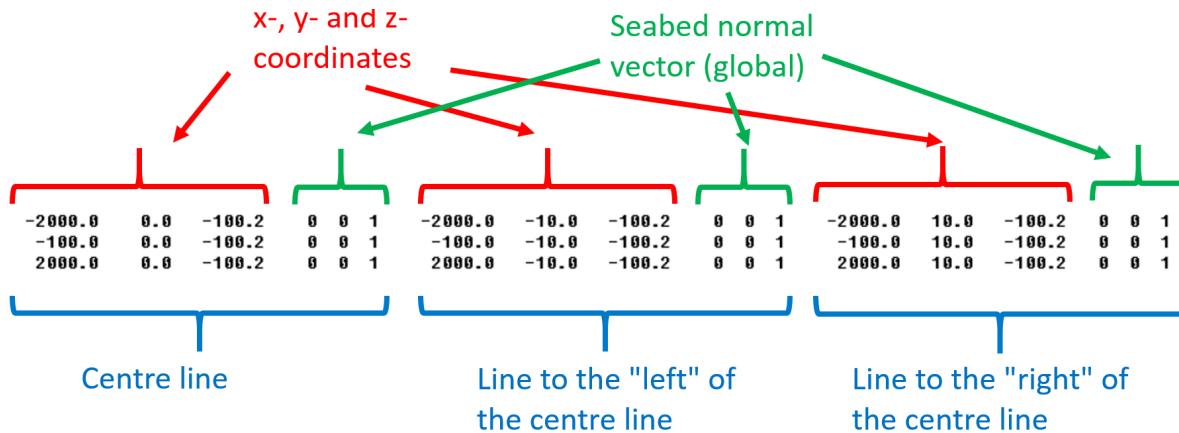


Figure 3.5: Input format for a route file with 3 lines.

### 3.8.2 Survey route file

If **COFILE** represents a survey file, then SIMLA will generate a seabed surface with the same performance as for the standard case in Section 3.8.1. The two first lines of the file consist of key data, and the succeeding lines of the file define the survey data as shown in Fig. 3.9.

The first line of **COFILE** must contain five values: NPOIN5, NINFO5, DIAM, TREF and RLEC. NPOIN5 refers to the number of survey data entries per line of the file, i.e.

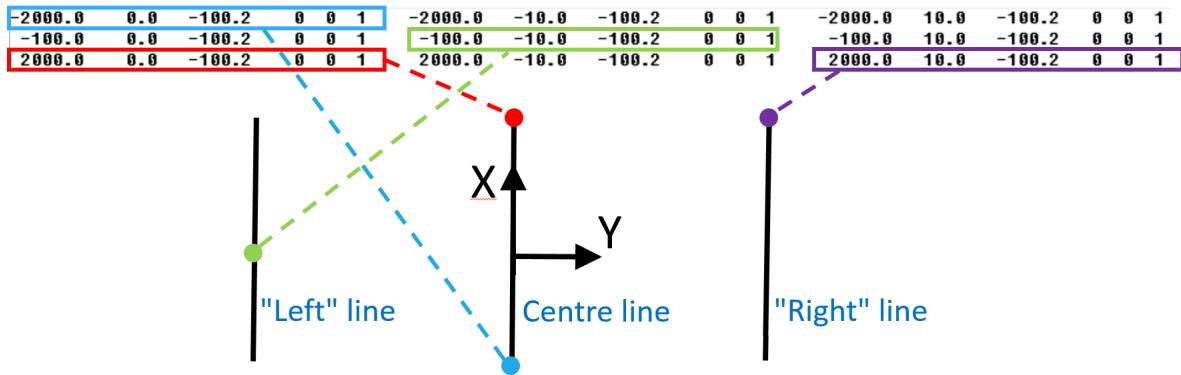


Figure 3.6: Illustration of line definition for a route file with 3 lines.

the number of columns describing the survey data. NINFO5 refers to the number of columns for the survey data that will be used for generating the seabed surface. The NINFO5 columns are selected through the pointers defined in second key data line. DIAM refers to the pipeline diameter, TREF is the reference time for the survey soil embedment elaborated in Section 3.8.3, and RLEC is the characteristic route length applied for smoothening of the user-defined x- and y-coordinates. A spline interpolation method based on a third order polynomial is applied for smoothening. Large values of RLEC increases the smoothening, while small values of RLEC gives less smoothening. The applied value of RLEC should reflect the bending stiffness of the pipeline, in which high stiffness typically requires that RLEC is large. Selecting a too low value of RLEC may introduce convergence issues if the AUTOSTART option in Section 3.6.9 is applied.

The second line of **COFILE** defines pointers for each of the NINFO5 columns where the relevant data are found. The sign of the pointers is applied to shift the direction of axes. If a negative pointer is defined, the input coordinate values will be reflected about the origin of the coordinate axis. For a 5 point file, the number of pointers is NINFO5=13 and must have the following representation for pointer 1 to pointer 13 as illustrated in Fig. 3.9:

1. The cover indicator. If non-zero, then  $z_1$  refers to top of soil.
2. The cover height. Only relevant for non-zero cover indicator.
3.  $x$ -coordinate of center line,  $x_1$ .
4.  $y$ -coordinate of center line,  $y_1$ .
5.  $z$ -coordinate of top pipe or soil,  $z_1$ .
6.  $s$ -coordinate for closest left survey point,  $s_2$ .
7.  $z$ -coordinate for closest left survey point,  $z_2$ .

8.  $s$ -coordinate for closest right survey point,  $s_3$ .
9.  $z$ -coordinate for closest right survey point,  $z_3$ .
10.  $s$ -coordinate for furthest left survey point,  $s_4$ .
11.  $z$ -coordinate for furthest left survey point,  $z_4$ .
12.  $s$ -coordinate for furthest right survey point,  $s_5$ .
13.  $z$ -coordinate for furthest right survey point,  $z_5$ .

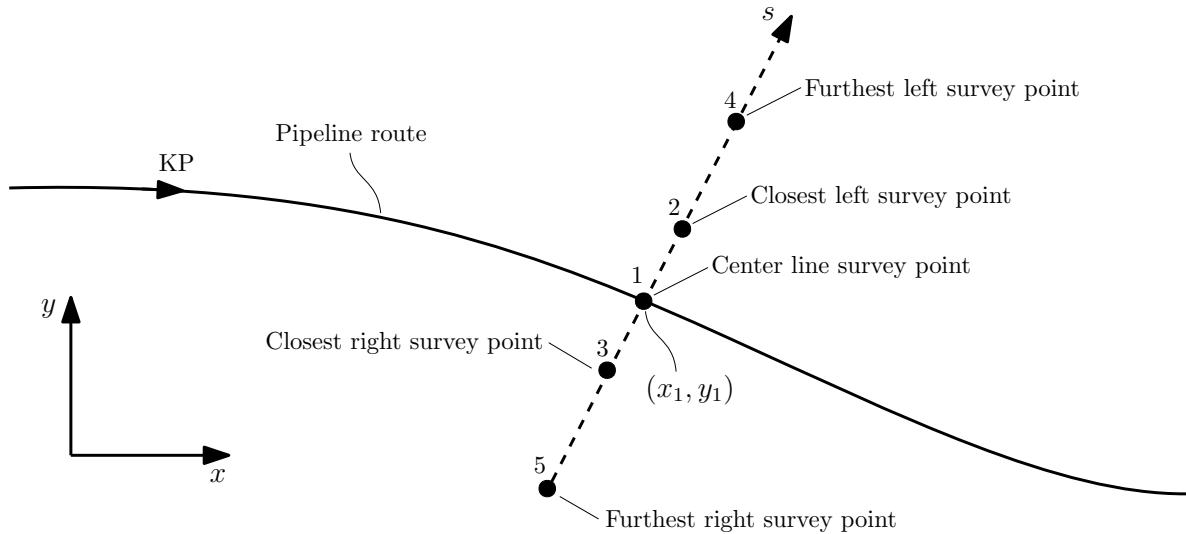


Figure 3.7: Definition of 5 survey points, the  $s$ -axis and the  $x_1$ - and  $y_1$ -coordinates.

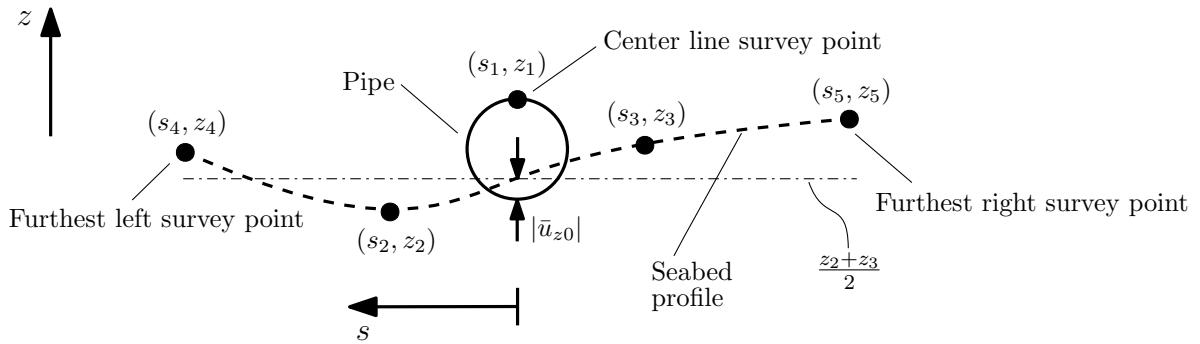


Figure 3.8: Definition of  $s$ - and  $z$ -coordinates for 5 survey points and penetration  $\bar{u}_{z0}$ .

After the two key data lines, an arbitrary number of datalines describing the survey data can be given as shown in Fig. 3.9. All the data must be integer or real numbers, where the indicators 'P' or 'C' used by survey contractors to indicate bare pipe or soil cover must be replaced by 0 and 1. The survey points are assumed to be located on a

Key data line 1													Key data line 2: Pointers to column data												
13	13	0.3	0.5	20.0	1	2	3	4	5	6	10	7	11	8	12	9	13								
0	0.0	-2.0	0.0	-99.4	0.5	-0.5	1.4	-1.4	-99.9	-99.9	-99.0	-99.0	-99.0	-99.0	-99.0	-99.0	-99.0	-99.0	-99.0	-99.0	-99.0	-99.0	-99.0		
0	0.0	0.0	0.0	-99.35	0.5	-0.5	1.4	-1.4	-99.7	-99.8	-99.0	-99.0	-99.0	-99.0	-99.0	-99.0	-99.0	-99.0	-99.0	-99.0	-99.0	-99.0	-99.0		
0	0.0	2.0	0.0	-99.4	0.5	-0.5	1.4	-1.4	-99.9	-99.9	-99.0	-99.0	-99.0	-99.0	-99.0	-99.0	-99.0	-99.0	-99.0	-99.0	-99.0	-99.0	-99.0		
0	0.0	4.0	0.0	-99.35	0.5	-0.5	1.4	-1.4	-99.8	-99.6	-99.0	-99.0	-99.0	-99.0	-99.0	-99.0	-99.0	-99.0	-99.0	-99.0	-99.0	-99.0	-99.0		
0	0.0	6.0	0.0	-99.5	0.5	-0.5	1.4	-1.4	-99.7	-99.6	-99.0	-99.0	-99.0	-99.0	-99.0	-99.0	-99.0	-99.0	-99.0	-99.0	-99.0	-99.0	-99.0		

Survey data

$x_1 \quad y_1 \quad z_1 \quad s_2 \quad s_3 \quad s_4 \quad s_5 \quad z_2 \quad z_3 \quad z_4 \quad z_5$

Figure 3.9: Input format of 5 point survey file.

straight line in the  $xy$ -plane that is perpendicular to the route as illustrated in Fig. 3.7. Further, the user-defined  $s$ - and  $z$ -coordinates of the survey points and the penetration  $u_{z0}$  computed by SIMLA are defined in Fig. 3.8.

### 3.8.3 Modelling of initial soil embedment

Modelling of initial soil embedment for the standard route file format in Section 3.8.1 is done by using the KP-dependent initial soil embedment feature available by the **EL-PROP** card in Section 3.23.11. This modelling option is available for both the CONT125 and the CONT126 element types.

To model initial soil embedment based on the survey route file in Section 3.8.2, the AUTOSTART option in Section 3.6.9 and the **INISTR** card in Section 3.34 must be used together. For all CONT126 seabed contact elements, the **INISTR** card must be applied for DOF 3 with load history factor equal to 1.0 and a dummy initial penetration value. The dummy initial penetration will then be over-ruled by the stressfree initial penetration  $u_{z0}$  computed by SIMLA according to,

$$u_{z0} = \begin{cases} f_1 \cdot \bar{u}_{z0} - f_2 \cdot u_z(t_0) & \text{if } u_z(t_0) \leq 0.0 , \quad t > t_0 \quad (\text{contact at } t=t_0) \\ f_1 \cdot \bar{u}_{z0} & \text{if } u_z(t_0) \geq 0.0 , \quad t > t_0 \quad (\text{no contact at } t=t_0) \\ f_1 \cdot \bar{u}_{z0} - f_2 \cdot u_z(t) & \text{if } u_z(t) \leq 0.0 , \quad t \leq t_0 \end{cases} \quad (3.2)$$

where  $f_1$  is the load factor from the **INISTR** card,  $f_2$  is set equal to  $f_1$  but is limited by  $f_2 \leq 1.0$ ,  $u_z$  is the contact element elastic penetration used for computing the normal contact force  $F_z$ . Here,  $u_z < 0.0$  means that the pipeline penetrates into the survey seabed, i.e. the virgin seabed modified by the stressfree initial penetration  $u_{z0}$ . The survey penetration reference time is denoted  $t_0=\text{TREF}$  and  $\bar{u}_{z0}$  is the survey penetration estimated as,

$$\bar{u}_{z0} = z_1 - \frac{z_2 + z_3}{2} - \text{DIAM} \leq 0.0 \quad (3.3)$$

where the survey point  $z$ -coordinates are defined in Fig. 3.8, and the sign convention  $\bar{u}_{z0} < 0.0$  means that the pipeline penetrates into the virgin seabed. With this, the pipeline will penetrate  $\bar{u}_{z0}$  into the seabed as given by Eq. (3.3) at the simulation time  $t_0 = \text{TREF}$ . The purpose of Eq. (3.2) is to adjust for the contact element elastic penetration  $u_z$  such that the same soil penetration  $\bar{u}_{z0}$  as in the survey is achieved at simulation time  $t_0 = \text{TREF}$ .

The survey soil penetration  $\bar{u}_{z0}$  will only be applied if the analysis is initiated by using the AUTOSTART option, the CONT126 element type and a dummy **INISTR** card for the CONT126 elements, otherwise  $u_{z0}$  in Eq. (3.2) will be set equal to zero. Note that Eq. (3.3) assumes that the three survey points are located sufficiently close to each other and that the horizontal distance between survey points 1 and 2 and between survey points 1 and 3 are equal. Further, the survey soil penetration  $\bar{u}_{z0}$  will be set to zero for non-zero cover indicators.

The survey soil penetration  $\bar{u}_{z0}$  is included in the total soil penetration used for calculating the transverse resistance for the **CLAY\_Y** and **SAND\_Y** models in Sections 3.37.12 and 3.37.13.

The user may manipulate the survey soil penetration computed by SIMLA. The computed values of the survey soil penetration  $\bar{u}_{z0}$  in Eq. (3.3) are accessible in the output file 'penetration.txt' as a function of KP. These values can be manipulated and re-applied in the following ways,

- Using the survey file format in Section 3.8.2, one can de-activate the survey soil penetration by removing the dummy **INISTR** input card, and then instead use the KP-based initial soil embedment feature in Section 3.23.11 for applying modified values based on the 'penetration.txt' file.
- Using the survey file format in Section 3.8.2, one can retain the survey soil penetration according to Eqs. (3.2) and (3.3), and then introduce non-zero corrections at certain KP-ranges by using the KP-based initial soil embedment feature in Section 3.23.11. The total soil penetration will then be a sum of the model in Eqs. (3.2) and (3.3) and the model in Eq. (3.19).

If the survey soil penetration model and the KP-based initial soil embedment model are applied simultaneously, the reference time  $t_0$  in Eq.(3.2) will be selected as the largest of TREF and **T0\_UZ** in Section 3.23.11. Further, the load factor  $f_2$  in Eq. (3.2) will be selected from the survey penetration model limited by  $f_2 \leq 1.0$ .

### 3.9 CROSSGEOM - Cross sectional geometry

The format of the card is as follows:

```
CROSSGEOM NAME CTYPE ....
```

where

**NAME:** name of cross sectional geometry. This will be used as reference in the **ELPROP** card, see Section 3.23.

**CTYPE:** Type of geometry for this section. The following types are implemented:

**TUBE** : Circular cross section with finite thickness.

**CYLR** : Cylinder cross section.

**CORR** : Corrugated cylinder.

**BOX** : Box shaped cross section. A thin walled rectangular profile.

**RECT** : A filled rectangular cross section.

**GENERAL** : General cross section.

### 3.9.1 TUBE

For **CTYPE = TUBE** the format is:

```
CROSSGEOM NAME TUBE RM TH
```

where with reference to Fig. 3.10:

**RM:** Mean radius. (unit: L)

**TH:** thickness. (unit: L)

### 3.9.2 CYLR

For **CTYPE = CYLR** the format is:

```
CROSSGEOM NAME CYLR RI RO
```

where:

**RI:** Inner radius (zero for e.g. round armours). (unit: L)

**RO:** Outer radius. (unit: L)

### 3.9.3 CORR

For **CTYPE = CORR** the format is:

```
CROSSGEOM NAME CORR VAR RI RO TH LP
```

where, with reference to Fig. 3.11:

**VAR:** Variation parameter, 1=Harmonic, 2=Circular If **VAR** = 1, the radial variation along the cylinder is taken to be  $r(x) = \frac{1}{2}(r_o - r_i) \sin \frac{2\pi x}{L_p}$ . Then if **VAR** = 2, the radial variation along one quarter circle, starting from one peak, follows the circular relation  $r(x) = \sqrt{\frac{1}{4}(r_o - r_i)^2 - x^2}$  with  $L_p = 2(r_o - r_i)$  and where  $L_p$  is the length between two consecutive peaks. For the time being, only harmonic geometry is allowed for.

**RI:** Inner radius of corrugated body. (unit: L)

**RO:** Outer radius of corrugated body (unit: L)

**TH:** Plate thickness of profile. (unit: L)

**LP:** Length between each top of corrugation. Dummy for **VAR** = 2 since in that case  $L_p = 2(r_o - r_i)$  (unit: L)

### 3.9.4 BOX

For **CTYPE** = **BOX** the format is:

**CROSSGEOM NAME BOX WI HI WO HO**

where, with reference to Fig. 3.12:

**WO:** Outside width of box. (unit: L)

**HO:** Outside height of box. (unit: L)

**WI:** Inner width of box. (unit: L)

**HI:** Height inside of box. (unit:L)

### 3.9.5 RECT

For **CTYPE** = **RECT** the format is:

**CROSSGEOM NAME RECT W T**

where:

**W:** Width of rectangle. (unit: L)

**T:** Thickness (height) of rectangle. (unit: L)

If **CTYPE** = **GENERAL**, the geometry is defined by a local right handed Cartesian xyz-coordinate system as shown in Fig. 3.13. The local x-axis points along the pipe.

The geometry is defined by boundary curves where the boundary is divided into a number of segments which again may be divided into an arbitrary number of intervals **NINTER**. The geometry of the boundary curves is always to be specified in counter-clockwise order about the local x-axis.

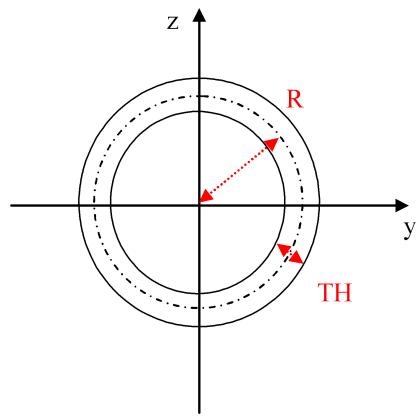


Figure 3.10: **CROSSGEOM** of type **TUBE**.

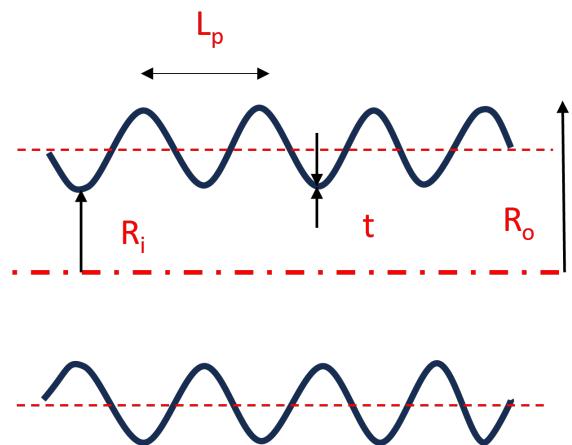


Figure 3.11: **CROSSGEOM** of type **CORR**.

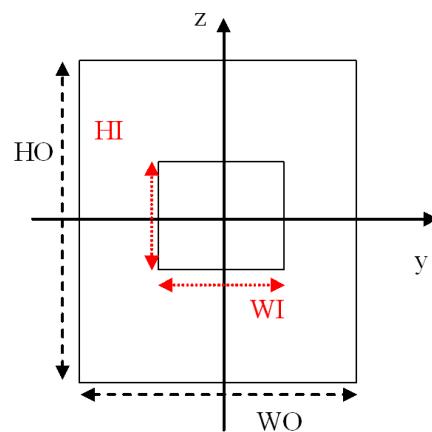


Figure 3.12: **CROSSGEOM** of type **BOX**.

### 3.9.6 GENERAL

For **CTYPE = GENERAL** the format is:

```
CROSSGEOM NAME  GENERAL Y0 Z0 CURVCODE P1 P2 P3 NINTER INTERFACE
```

```
...  ...  ...
```

- Y0:** In order to describe the geometry of hollow section BEAM elements it is necessary to use two boundary curves. **Y0** defines the y-start point of the inner boundary curves relative to where the outer boundary curve starts.
- Z0:** In order to describe the geometry of hollow section BEAM elements it is necessary to use two boundary curves. **Z0** defines the z-start point of the inner boundary curves relative to where the outer boundary curve starts.

**CURVCODE:** Curve code, given as a character string which may have the following values:

**S :** Segment is straight

**CI :** Segment is circular with the circle centre at the cross section inside.

**CO :** Segment is circular with the circle centre at the cross section outside.

**P1:** If **CURVCODE = S**: length of segment, unit: mm. If **CURVCODE = CI** or **CO**: angle at start of circle, unit: deg.

**P2:** If **CURVCODE = S**: angular orientation of segment local YZ system, where the Y-axis is directed along the element and the Z-axis is pointing outwards, unit: deg. If **CURVCODE = CI** or **CO**: angle at end of circle, unit: deg.

**P3:** Circle radius. Note that when specifying shells, the radius is always the outer radius, unit: mm. Dummy for **CURVCODE =S**.

**NINTER:** Number of intervals within the segment.

It is preferred that the boundary is defined with nodes exactly on the top, at the bottom and at the leftmost and rightmost location of the geometry. This will minimize unintended effects from constraints that are applied to obtain stability.

**INTERFACE:** The local interface number. Interface numbers higher than 1 are used to describe holes in a geometry.

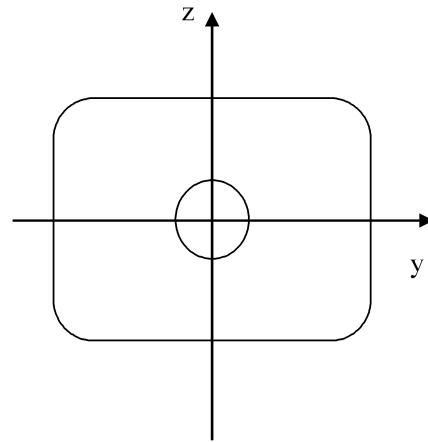


Figure 3.13: **CROSSGEOM** of type **GENERAL**.

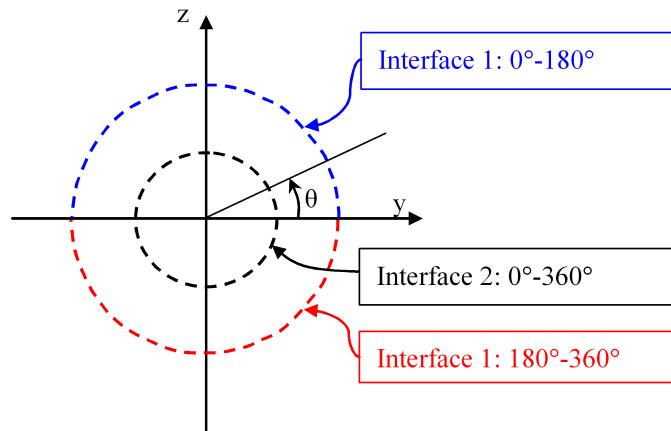


Figure 3.14: **CROSSGEOM pipegen** of type **GENERAL** from example.

EXAMPLE:

---

```

#          name      ctype    W0      H0      WI    HI
CROSSGEOM boxsec   BOX      0.5     0.5     0.4   0.4
#
#          name      ctype      R      TH
CROSSGEOM pipesec  PIPE     0.45    0.1
#
#          This will result in a pipe with OD 2 and ID = 1
#          name      ctype    Y0 Z0 CURVCODE P1  P2  P3  NINTER INTERFACE
CROSSGEOM pipegen  GENERAL  1   0    CI        0   360  1. 30       1
#                                         0.5 0    CI        0   360  .5 30       2
#

```

---

### 3.10 CURLOAD - current loading

The current loading is specified along the curvilinear path defined in the **SEALO** input card, allowing for interpolation of an arbitrary number of current profiles. The following format is applied:

```
CURLOAD NO TYPE DEPTH VELO ZROT
```

```
.. .. ..
```

where the following definitions and units apply:

**NO**: Current profile number.

**TYPE**: **GLOBAL** or **LOCAL**. The current direction vector is aligned with the global *x*-axis for the former and in the direction of the vessel heading for the latter.

**DEPTH**: Depth coordinate, at sea surface = **0.0**. (unit: L)

**VELO**: Velocity along the current direction vector. (unit: LT<sup>-1</sup>)

**ZROT**: Rotation of the current direction vector about the global *z*-axis. (unit: rad)

An arbitrary sequence of depths, velocities and rotations may be given, as long as the depth is constantly decreasing.

EXAMPLE: \_\_\_\_\_

```
#  
#      no   type   depth   velo   zang  
CURLOAD 100 LOCAL      0    1.0    0.0  
                  -100   1.5    0.0  
                  -500   0.1   1.57  
                 -5000  0.1   1.57
```

### 3.11 DYNCONT - Control parameters for dynamic analysis

The parameters controlling the dynamic analysis are defined by the following format:

```
DYNCONT MSTAT ALPHA1 ALPHA2 ALPHA [BETA GAMMA]
```

where:

**MSTAT**: Parameter for choosing mass matrix type:

**1** : Concentrated mass matrix.

**2** : Consistent mass matrix.

**ALPHA1**: Mass proportional damping factor,  $\alpha_1$ .

**ALPHA2**: Stiffness proportional damping factor,  $\alpha_2$ .

**ALPHA**:  $\alpha$  in the HHT- $\alpha$  time integration method.

**BETA**:  $\beta$  in the HHT- $\alpha$  time integration method.

**GAMMA**:  $\gamma$  in the HHT- $\alpha$  time integration method.

**BETA** and **GAMMA** are optional. If only **ALPHA** is specified, default values for **BETA** and **GAMMA** are used. The default values are  $\gamma = \frac{1}{2}(1 - 2\alpha)$  and  $\beta = \frac{1}{2}(1 - \alpha)^2$ . The recommended value for the  $\alpha$ -parameter is  $-1/3 \leq \alpha \leq 0$ . By setting  $\alpha = 0$ ,  $\gamma = 1/2$  and  $\beta = 1/4$ , the Newmark  $\beta$  method with constant average acceleration is obtained.

EXAMPLE:

---

```
#      imass  alfa1  alfa2  alfa
DYNCONT   1       0.0    0.095  -0.05
```

---

### 3.12 DYNRES\_ - dynamic results

By the **DYNRES\_** cards, user selected time histories will be stored on the .dyn file on a format that enables plotting in XPOST and post-processing by DYNPost. The results are available for all time steps. The following format is applied:

**DYNRES\_N, E OR I ...**

where **N** means nodal results, **E** means element results and **I** means integration station results. Note that there exists a memory limitation for **DYNRES\_** for memory demanding analyses using the restart option. The allowable options are defined below:

#### 3.12.1 DYNRES\_N

For **DYNRES\_** with the **N** (nodal results) option the format is as follows:

**DYNRES\_N TYPE NODEID DOF [NODEID2 DOF2]**

where

**TYPE**: Type of result, where allowable values are:

- 1** : nodal displacement
- 2** : nodal velocity
- 3** : nodal acceleration
- 4** : relative displacement

**NODEID**: Node ID.

**DOF**: Degree of freedom number:

- 1** : global x-displacement
- 2** : global y-displacement
- 3** : global z-displacement
- 4** : global x-rotation
- 5** : global y-rotation
- 6** : global z-rotation

**NODEID2**: ID of node 2 (optional, applies only for **TYPE=4**).

**DOF2**: DOF of node 2 (optional, applies only for **TYPE=4**).

If the optional numbers **NODEID2** and **DOF2** are specified the relative displacements between the two nodes will be presented.

### 3.12.2 DYNRES\_E

For elements of type PIPE, CONT, SPRING and BODY, **DYNRES\_** with the **E** option (element results) can be applied. The format is as follows:

**DYNRES\_E TYPE ELID ELNOD DOF [REFSYS]**

where:

**TYPE**: Type of result where allowable values are:

- 1** : Displacements.
- 2** : Internal forces and moments.
- 3** : Hydrodynamic loads. Only for DROPS loading and element types PIPE31, PIPE33 and COMPIPE42.
- 4** : HYDROPRO interpolation variables. Only for element type BODY502.
- 5** : HYDROPRO hydrodynamic mass at the HYDROPRO mass center. Only for element type BODY502.
- 6** : HYDROPRO drag and lift load at the HYDROPRO drag center. Only for element type BODY502.
- 7** : Wave kinematics. Only for DROPS loading and element types PIPE31, PIPE33 and COMPIPE42.

**ELID**: Element ID.

**ELNOD**: Element node (max. 2 for PIPE and SPRING, max. 1 for CONT and BODY502)

**DOF**: Degree of freedom number. The meaning is element type dependent:

**PIPE31-39, SPRING136-137, BODY502** : element types with **TYPE=1,2**

- 1** : x-displacement/x-force

- 2** : y-displacement/y-force
- 3** : z-displacement/z-force
- 4** : x-rotation/x-moment
- 5** : y-rotation/y-moment
- 6** : z-rotation/z-moment

**CONT** : element types with **TYPE=1,2**

- 1** : x-displacement/x-force
- 2** : y-displacement/y-force
- 3** : z-displacement/z-force
- 4** : x-rotation/x-moment (only for CONT126 and CONT128)

**PIPE31,33,42** : element types with **TYPE=3**

- 1** : x-force (unit: F)
- 2** : y-force (unit: F)
- 3** : z-force (unit: F)

**BODY502** : element type with **TYPE=4**

- 1** : Distance to seabed (unit: L)
- 2** : Euler-x angle for hydrodynamic mass (unit: deg)
- 3** : Euler-y angle for hydrodynamic mass (unit: deg)
- 4** : Hydrodynamic angle of attack about x-axis for drag and lift (unit: deg)
- 5** : Hydrodynamic angle of attack about y-axis for drag and lift (unit: deg)
- 6** : Hydrodynamic angle of attack about z-axis for drag and lift (unit: deg)
- 7** : Geometric angle between body element x-axis and seabed surface unit normal vector for drag and lift (unit: deg)
- 8** : Geometric angle between body element y-axis and seabed surface unit normal vector for drag and lift (unit: deg)
- 9** : Geometric angle between body element z-axis and seabed surface unit normal vector for drag and lift (unit: deg)

**BODY502** : element type with **TYPE=5**

- 1** : Hydrodynamic mass in x-dir (unit: M)
- 2** : Hydrodynamic mass in y-dir (unit: M)
- 3** : Hydrodynamic mass in z-dir (unit: M)
- 4** : Rotational hydrodynamic mass about x-axis (unit:  $ML^2$ )
- 5** : Rotational hydrodynamic mass about y-axis (unit:  $ML^2$ )
- 6** : Rotational hydrodynamic mass about z-axis (unit:  $ML^2$ )

**BODY502** : element type with **TYPE=6**

- 1** : Drag and lift x-force (unit: F)

- 2** : Drag and lift y-force (unit: F)
- 3** : Drag and lift z-force (unit: F)
- 4** : Drag and lift x-moment (unit: FL)
- 5** : Drag and lift y-moment (unit: FL)
- 6** : Drag and lift z-moment (unit: FL)

**PIPE31,33,42** : element types with **TYPE=7**

- 1** : wave-induced velocity x-direction (unit:  $LT^{-1}$ )
- 2** : wave-induced velocity y-direction (unit:  $LT^{-1}$ )
- 3** : wave-induced velocity z-direction (unit:  $LT^{-1}$ )
- 4** : wave-induced acceleration x-direction (unit:  $LT^{-2}$ )
- 5** : wave-induced acceleration y-direction (unit:  $LT^{-2}$ )
- 6** : wave-induced acceleration z-direction (unit:  $LT^{-2}$ )
- 7** : wave elevation at the surface (unit: L)

**REFSYS**: Reference coordinate system for the result, which can be assigned the values **LOCAL** or **GLOBAL** where the former is default. **REFSYS** is only implemented for **TYPE=2** and the CONT153 and CONT164 elements, i.e. **LOCAL** will be applied for all other element types. Optional parameter.

For the CONT element types, the  $z$ -axis of the local system is pointing outwards from the master geometry in the contact point normal direction and the tangential plane is spanned by the  $x$ - and  $y$ -axes. Further, the output contact force acts on the master geometry for **REFSYS=LOCAL**, while it acts on the slave geometry for **REFSYS=GLOBAL**. The local system for the SPRING and PIPE element types coincide with the element system as defined by the **ELORIENT** card in Section 3.22.

The hydrodynamic loads for **TYPE=3** are calculated based on the fluid relative motion, i.e. the structural motion is taken into account.

### 3.12.3 DYNRES\_I

For elements types PIPE33, PIPE39 and CONT126, **DYNRES\_** with the **I** option (integration station results) can be applied. The format is as follows:

**DYNRES\_I TYPE ELID IGAU IPOINT**

**TYPE**: Type of result, where allowable values are:

- 1** : sigma-xx (unit:  $FL^{-2}$ )
- 2** : strain-xx (unit: -)
- 3** : uz-soil, penetration of pipe into virgin soil (unit: L)
- 4** : uz-soil-max, maximum penetration of pipe into virgin soil (unit: L)

**5** : energy-soil, penetration-dependent energy  $E$  defined in the SIMLA Theory Manual, (unit:  $\text{FLL}^{-1}$ )

**6** : fy-friction, soil friction force (unit: F)

**7** : fy-passive, soil passive resistance force (unit: F)

**ELID**: Element ID.

**IGAU**: Element gauss station 1-3, 1 and 3 are at element ends, see Fig. 3.15 for illustration.

**IPOINT**: Integration point number, max. **NPOINT** see Section 3.6 and Fig. 3.15.

The result types **3-7** are only available for element type CONT126 with the **CLAY\_Y** and **SAND\_Y** material types.

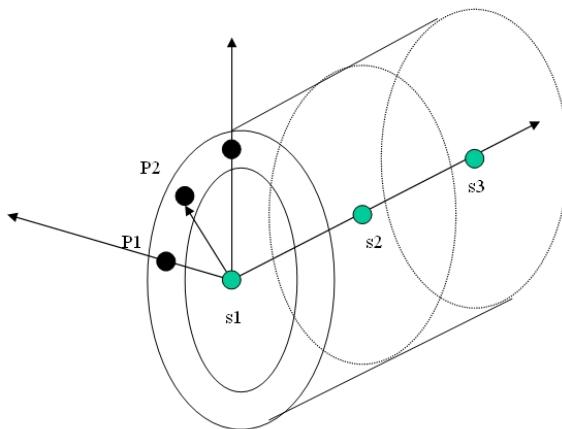


Figure 3.15: Element Gaussian station numbers  $s_1$  to  $s_3$  along the element axis. Integration points  $P_1$  and  $P_2$ .

EXAMPLE:

---

```
#          type  elno end  dof  refsys
DYNRES_E   2      2    1    1
DYNRES_E   2     440    1    1  global
DYNRES_E   2     439    1    1  global
#          type node  dof
DYNRES_N   1    3001    1
DYNRES_N   2    3001    1
DYNRES_N   1    3001    2
DYNRES_N   2    3001    2
```

---

### 3.13 DROPS\_GRID - Grid for hydrodynamic loading

The grid defines spatial points where time series of wave kinematics and fixed pipe hydrodynamic forces are pre-generated. The format is as follows:

```
DROPS_GRID CONAME LOADNO KPSTART KPEND DX [VAZ_STORE  
WAVE_STORE]
```

where

**CONAME**: Name of contact surface as defined by the **COSURFPR** card.

**LOADNO**: Load number. Must exist in a **DROPS\_LOAD** card.

**KPSTART**: Grid domain start point given in terms of curvilinear coordinate along route center line (unit: L).

**KPEND**: Grid domain end point given in terms of curvilinear coordinate along route center line (unit: L).

**DX**: Spacing of grid points along the route center line (unit: L)

**VAZ\_STORE**: Storage of kinematic velocity and acceleration in z-direction. Allowable values are **ON** or **OFF**.

Optional, default value **VAZ\_STORE=ON**

**WAVE\_STORE**: Storage of wave elevation. Allowable values are **ON** or **OFF**.

Optional, default value **WAVE\_STORE=ON**

Fig. 3.16 illustrates how a single hydrodynamic grid is defined. The grid domain is specified by **KPSTART** and **KPEND** which refer to the curvilinear coordinate system defined by the **COSURFPR** input card in Section 3.8. The *x*-, *y*- and *z*-coordinates of the grid points are calculated based on interpolation of the route center line data points specified in the file specified for **COFILE** in Section 3.8. The pipeline must be directed along the route center line. Otherwise, the fluid velocity normal direction for the pipeline will become incorrect.

The spacing of the grid points may be very coarse for long-crested waves perpendicular to the pipeline. For short-crested waves, the spacing **DX** should be able to capture at least 15 points of the shortest wave length that gives significant horizontal wave-induced velocity at the seafloor. A convergence study with respect to the grid resolution is recommended. Note that **DX** should not be set uncritically small for long models as the CPU time for pre-generation and memory consumption for storage of time series may become significant for long analysis durations.

Time series of the wave kinematics are pre-generated by inverse Fast Fourier transformation. Thereafter, time series of the fixed pipe hydrodynamic loads for **LOAD-TYP=WAKE** specified in the **DROPS\_HCOEF** input card are pre-generated. The

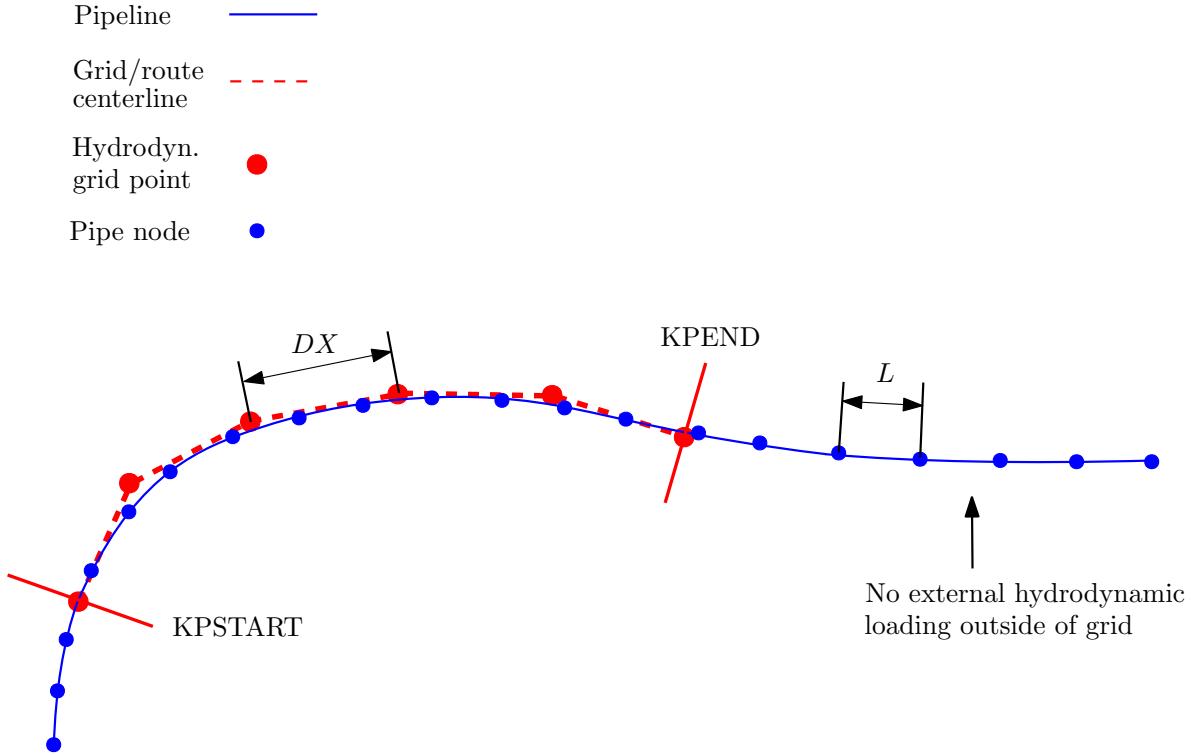


Figure 3.16: Definition of hydrodynamic grid

wave kinematics and the fixed pipe hydrodynamic wake model loads are then spatially interpolated from the grid points towards the pipe elements, where hydrodynamic loads for the moving pipeline are computed.

Hydrodynamic load time series must be pre-generated for all diameters and pipe roughnesses specified in the **DROPS\_HCOEF** input card(s) when **LOADTYP=WAKE** is selected, see Section 3.14. This could give many excessive time series for long models with varying diameter and roughness, in particular for grid points located far away from the diameter and roughness in question. Hence, in order to reduce memory consumption, the fixed pipe hydrodynamic wake model loads are set equal to zero for grid points that are located more than  $DS_{max} = 5.0 \cdot \max [L, DX]$  away from the closest pipe node with a given diameter and pipe roughness, at the time **TSTART** specified in the **DROPS\_LOAD** input card. Here,  $L$  is the pipe element length and  $DX$  is the mesh longitudinal spacing **DX**, see Fig. 3.16. For consistency purposes, the hydrodynamic loading for all **LOADTYP** options in Section 3.14 is set equal to zero for pipe nodes that are located farther away from the grid domain than  $DS_{max}$  at time **TSTART**.

The following items should be considered in relation to modelling and input data verification:

- A pipe node will only be subjected to hydrodynamic loading when located be-

tween grid points belonging to the same grid domain. No external hydrodynamic loading is applied if the pipe node is located outside the grid domain, see Fig. 3.16. Outside of the grid domain, the hydrodynamic loading consists only of drag damping and added mass loads due to the pipe motion using coefficient values valid at zero pipe-seabed gap.

- A pipe node shall not displace longitudinally more than  $DS_{max}$  away from the position occupied at the instant **TSTART** in the **DROPS\_LOAD** input card. Otherwise, the fixed pipe hydrodynamic wake model loads will be set equal to zero as explained in relation to  $DS_{max}$  above. A warning message will be printed to the SIMLA log file (prefix.slf) for pipe nodes that displace into grid regions where fixed pipe hydrodynamic wake model load time series do not exist.
- The hydrodynamic load status of all pipe element nodes is printed to the SIMLA output file (prefix.sof) at the time **TSTART** specified in the **DROPS\_LOAD** card in Section 3.15. Element nodes that are located farther away than  $DS_{max}$  from the grid domain at **TSTART** will be assigned zero hydrodynamic load status, and will consequently never be subjected to hydrodynamic loading, also if they move inside the grid domain. See definition of  $DS_{max}$  above.
- The grid coordinates and information about the pre-generated fixed pipe wake model load time series are printed to the SIMLA output file (prefix.sof), see **IPRINT** in Section 3.6. At each grid point, combinations of hydrodynamic diameters and pipe roughness parameters **IROUGH** from Section 3.14 are tabulated, as well as the existence of pre-generated fixed pipe hydrodynamic wake model loads, the water depth and the seabed z-coordinate. All of the data is printed at time **TSTART** specified in the **DROPS\_LOAD** input card.
- The hydrodynamic load time series for the moving pipe are available by means of the **DYNRES\_E** command in Section 3.12.2.
- Hydrodynamic loads are not applied for pipe nodes that are connected to the CONT128 element type. These nodes are assumed to be buried into the seabed and should consequently not be subjected to hydrodynamic loads.

*Note:* Wave kinematics in the  $z$ -direction is not important for cases where the pipe is resting on the sea and will often be negligible for free spans with low height. In such cases, the wave kinematics in the  $z$ -direction can be set to zero by specifying **WAZ\_STORE=OFF**. This will reduce the memory consumption.

*Note:* The memory consumption may be reduced by specifying **WAVE\_STORE=OFF**. This de-activates storage of the wave elevation time series and is the recommended option if wave elevation results are not requested in Section 3.12.2.

EXAMPLE: \_\_\_\_\_

```
#-----
#      routename   kpstart      kpend     dx
#-----
DROPS_GRID  cosurf1        100.    125.0   10.0
#-----
```

---

EXAMPLE: \_\_\_\_\_

```
#-----
#      routename   kpstart      kpend     dx   vaz-storage   wave-storage
#-----
DROPS_GRID  cosurf1        100.    125.0   10.0   OFF          ON
#-----
```

---

### 3.14 DROPS\_HCOEF - Hydrodynamic coefficients

This input card defines hydrodynamic coefficients and load scaling factors for computation of loads from the wave and current kinematics defined by the **DROPS\_GRID** and **DROPS\_LOAD** input cards. Only pipe element groups that are assigned properties from a **DROPS\_HCOEF** will be subjected to hydrodynamic loads. Further, the pipe elements must be located within the KP-interval specified for the grid and be linked by a **CONTINT** card towards the sea element group that is assigned the fluid kinematics.

**DROPS\_HCOEF IROUGH YFAC ZFAC NSTEP NELGRP T\_1 ... T\_N GAPMORI  
LOADTYP MASSNAME LIFTNAME DRAGNAME PIPE\_1 ... PIPE\_N | YPEN-  
FACNAME ZPENFACNAME RTRYFACNAME RTRZFACNAME]**

where

**IROUGH**: Integer defining pipe roughness normalized by diameter,  $R_k = k_s/D$ , where  $k_s$  is the mean roughness grain size. The following integer values are allowed:

- 1** :  $R_k = 1 \cdot 10^{-3}$  (unit: -)
- 2** :  $R_k = 1 \cdot 10^{-2}$  (unit: -)
- 3** :  $R_k = 5 \cdot 10^{-2}$  (unit: -) Recommended option

**YFAC**: Load scaling factor for the lateral hydrodynamic load (unit: -).

**ZFAC**: Load scaling factor for the hydrodynamic lift load (unit: -).

**NSTEP**: Number of time instants **T\_I** for updating the gap-dependent hydrodynamic coefficients and the load scale factors for soil penetration and trenching. Updating is performed for all load steps if **NSTEP=0**.

**NELGRP:** Number of pipe element groups referred to by **PIPE\_1–PIPE\_N**.

**T\_I:** Time instant(s) where the gap-dependent hydrodynamic coefficients are to be updated (unit: T).

**GAPMORI:** The seabed-pipe gap at which the loading is based solely on the Morison load model. See further explanation below. (unit: L).

**LOADTYP:** Load model type at zero gap:

**MORI** : The Morison load model is applied at zero pipe-seabed gap with  $C_m$ ,  $C_L$  and  $C_d$  from the gap-dependent tables.

**WAKE** : The wake load model is applied at zero pipe-seabed gap. Recommended option.

**MASSNAME:** Table name for gap-dependent mass coefficient  $C_m$  for the Morison load model.

**LIFTNAME:** Table name for gap-dependent lift coefficient  $C_L$  for the Morison load model.

**DRAGNAME:** Table name for gap-dependent drag coefficient  $C_d$  for the Morison load model.

**PIPE\_I:** Name of the pipe element group(s) subjected to hydrodynamic loading. Only element types PIPE31, PIPE33 or COMPIPE42 are allowed.

**YPENFACNAME:** Table name referring to the soil penetration dependent lateral hydrodynamic load scaling factor. If **YPENFACNAME=NONE** scaling is omitted. Optional, no scaling if omitted.

**ZPENFACNAME:** Table name referring to the soil penetration dependent hydrodynamic lift scaling factor. If **ZPENFACNAME=NONE** scaling is omitted. Optional, no scaling if omitted.

**RTRYFACNAME:** Table name referring to the trench KP-dependent lateral hydrodynamic load scaling factor. If **RTRYFACNAME=NONE** scaling is omitted. Optional, no scaling if omitted.

**RTRZFACNAME:** Table name referring to the trench KP-dependent lift hydrodynamic load scaling factor. If **RTRZFACNAME=NONE** scaling is omitted. Optional, no scaling if omitted.

The hydrodynamic loading is based solely on the **LOADTYP** model at zero gap. At gap values larger than **GAPMORI**, only the Morison load model with gap-dependent hydrodynamic coefficients contributes. The load factors for the **LOADTYP** model and the Morison model are calculated by linear interpolation between zero gap and **GAPMORI**.

The gap-dependent, soil penetration-dependent and trench KP-dependent quantities must be defined by the **TABLE** input card in Section 3.49. The table names must be identical to the names specified for the **MASSNAME**, **LIFTNAME**, **DRAGNAME**, **YPENFACNAME**, **ZPENFACNAME**, **RTRYFACNAME** and **RTRZFACNAME** charac-

ter strings. Further, the tables must consist of two columns, where the 1.column represents the argument values and the 2.column represents the function values.

The gap-dependent tables **MASSNAME**, **LIFTNAME**, **DRAGNAME** and the soil penetration-dependent tables **YPENFACNAME** and **ZPENFACNAME** apply dimensionless variables for the argument column. The gap-dependent table type uses the dimensionless gap-to-diameter ratio  $g/D$  as the variable. The penetration dependent tables employ the dimensionless penetration-to-diameter ratio  $z/D$  as the variable. The gap is positive when there is clearance between the pipe and the seabed, while the penetration is negative when the pipe penetrates into the soil. Hence, the tables must be defined in terms of positive  $g/D$ -values and negative  $z/D$ -values. Here,  $D$  is the pipe hydrodynamic diameter defined in Sections 3.23.1 and 3.23.2. Further, the contribution from soil penetration development according to the CLAY\_Y and SAND\_Y soil models, see Sections 3.37.12 and 3.37.13, is included for the penetration  $z$  applied for **YPENFACNAME** and **ZPENFACNAME**, while it is not included for the gap  $g$  applied for **MASSNAME**, **LIFTNAME** and **DRAGNAME**.

The trench KP-dependent tables **RTRYFACNAME** and **RTRZFACNAME** applies KP as the argument and the load reduction factor as the function value. The load reduction factor is dependent on the ratio between the trench depth and the pipe diameter,  $z_t/D$ . See formulae in (DNVGL, 2017) for calculation of the load reduction factor.

*Note:* The the mass coefficient  $C_m$  should be larger than 1.0, otherwise the added mass will be negative according to  $C_a = C_m - 1$ .

*Note:* For **NSTEP** ≠ 0, one of the time instants **T\_I** should be just after the time instant **TSTART** in Section 3.15. This is because zero gap will be assumed for the gap-dependent and penetration-dependent tables between **TSTART** and the next **T\_I** time instant. Likewise, the initial KP-value at analysis start will be applied for the KP-dependent table between **TSTART** and the next **T\_I** time instant.

*Note:* The load scaling factors **RTRYFACNAME** and **RTRZFACNAME** are overruled and set to 1.0 when the pipe is not in contact with the seabed.

*Note:* The load scaling factors **YFAC**, **ZFAC**, **YPENFACNAME** and **ZPENFACNAME** are applied regardless of whether the pipe is in contact with the seabed or not. This means that load scaling for gaps can be modelled by including positive  $\frac{z}{D}$ -values for **YPENFACNAME** and **ZPENFACNAME**.

*Note:* **ELPROP** card will be over-ruled by the **DROPS\_HCOEF** card.

EXAMPLE: \_\_\_\_\_

```
#  
#      irough    yfac      zfac      nstep nelgrp      t1      t2      gapmori  loadtyp
```

```
DROPS_HCOEF      3      0.85      0.85      2      2      3.0  50.0      0.3      WAKE
#
#      massname    liftname    dragname    pipe1    pipe2    yopenfacname zopenfacname
#              mass1       lift1       drag1     cable1     cable2      penfac1      penfac2
```

---

EXAMPLE:

---

```
#      irough   yfac      zfac      nstep nelgrp      t1      t2      gapmori loadtyp
DROPS_HCOEF      3      0.85      0.85      2      2      3.0  50.0      0.3      WAKE
#
#      massname    liftname    dragname    pipe1    pipe2    yopenfacname zopenfacname
#              mass1       lift1       drag1     cable1     cable2      penfac1      penfac2
#
# rtryfacname  rtrzfacname
loadrtry  loadrtrz
#
```

---

EXAMPLE:

---

```
#      irough   yfac      zfac      nstep nelgrp      t1      t2      gapmori loadtyp
DROPS_HCOEF      3      0.85      0.85      2      2      3.0  50.0      0.3      WAKE
#
#      massname    liftname    dragname    pipe1    pipe2    yopenfacname zopenfacname
#              mass1       lift1       drag1     cable1     cable2      None      None
#
# rtryfacname  rtrzfacname
loadrtry  loadrtrz
#
```

---

### 3.15 DROPS\_LOAD - Hydrodynamic loading

The fluid kinematics is based on constant uni-directional current and either (1) linear wave theory or (2) second order Stokes theory or (3) pre-generated wave kinematics read from file. Both long- and short-crested sea states are supported.

The hydrodynamic loading is computed at each grid point defined in Section 3.13 with hydrodynamic coefficients as specified in Section 3.14. No hydrodynamic loads will be computed if the input cards **DROPS\_GRID** and **DROPS\_HCOEF** are missing.

Only element types PIPE31, PIPE33 and COMPIPE42 will be subjected to DROPS hydrodynamic loading. If the element group is not assigned a **DROPS\_HCOEF** card, the

hydrodynamic loading will consist only of drag damping and added mass loads due to the pipe motion based on the hydrodynamic coefficients from the **ELPROP** card.

The input card format is as follows:

```
DROPS_LOAD ELGRP LOADNO HISTNO CURVEL CURANG CUHREF
CUROUGH WAVETYPE ...
```

where

**ELGRP**: Sea element group name.

**LOADNO**: Load number (unit: -).

**HISTNO**: Load time history number (unit: -).

**CURVEL**: Current velocity at **CUHREF** (unit: L/T).

**CURANG**: Current angle relative to global x-axis, see  $\varphi$  in Fig. 3.1 (unit: rad).

**CUHREF**: Current reference height above seabed for the logarithmic current profile (unit: L).

**CUROUGH**: Seabed roughness for the logarithmic current profile (unit: L).

**WAVETYPE**: Wave type. Allowable values are **IRREGULAR** for linear, irregular waves, **SECONDORDER** for wave kinematics based on second order Stokes theory and **FROMFILE** to read pre-generated time series of wave kinematics from file.

### 3.15.1 Irregular wave loading with current

The input card has the following format for **WAVETYPE=IRREGULAR**:

```
DROPS_LOAD ELGRP LOADNO HISTNO CURVEL CURANG CUHREF
CUROUGH IRREGULAR X0 Y0 WAVANG TP HS DT TDUR TSTART SEED SPEC
CURPROF ...
```

where

**X0**: global x-coordinate of the wave elevation origin (unit: L).

**Y0**: global y-coordinate of the wave elevation origin (unit: L).

**WAVANG**: Wave main direction measured by angle relative to the global x-axis, see  $\varphi_0$  in Eq. (3.8) and Fig. 3.1 (unit: rad).

**TP**: Peak period  $T_p$  (unit: T).

If **SPEC=10**, this is the regular wave period,  $T$ .

**HS**: Significant wave height  $H_s$  (unit: L).

If **SPEC=10**, this is the regular wave height,  $H$ .

**DT**: Time step  $\Delta t$  used for generation of wave kinematics and fixed pipe hydrodynamic loads (unit: T).

**TDUR:** Duration  $T_{dur}$  of wave and load time series measured from **TSTART** (unit: T).

**TSTART:** Start time  $T_{start}$  for generation of wave kinematics and fixed pipe hydrodynamic loads (unit: T).

**SEED:** Arbitrary integer between 0 and  $2^{31} - 1$  representing a set of random phase angles for the wave time series (unit: -).

**SPEC:** Wave spectrum type. Allowable values are:

**1** : Two parameter Pierson-Moskowitz spectrum.

**2** : Three parameter JONSWAP spectrum.

**10**: One component wave spectra. This will give a pure sine wave.

**CURPROF:** Current profile type. Allowable values are:

**STANDARD** : Standard logarithmic current profile.

**RPF109** : DNV-RP-F109 logarithmic current profile accounting for current-wave interaction.

### Irregular wave loading with standard current profile

A logarithmic current profile defined in terms of the **CUROUGH**-parameter is applied for **CURPROF=STANDARD**. The input card then has the following format:

```
DROPS_LOAD ELGRP LOADNO HISTNO CURVEL CURANG CUHREF
CUROUGH IRREGULAR X0 Y0 WAVANG TP HS DT TDUR TSTART SEED SPEC
STANDARD [PKDNESS SPREAD NDIR SPREADPAR]
```

where

**PKDNESS:** Peakedness parameter.

Dummy input if **SPEC≠2**.

Optional, default value: 3.3 (unit: -).

**SPREAD:** Wave spreading option.

Dummy input if **SPEC=10** (unit: -).

Allowable values are:

**SHORT** : Short-crested sea.

**LONG** : Long-crested sea, which is the default option.

**NDIR:** Number of directions for short-crested sea, see  $N_\varphi$  in Eq. (3.8).

Dummy for **SPREAD=LONG**.

Optional, default value: 1 (unit: -).

**SPREADPAR:** Spreading function exponent, see  $n$  in Eq. (3.10).

Dummy for **SPREAD=LONG**.

Optional, default value: 0 (unit: -).

The following example defines an irregular short-crested sea state with standard current

profile,

EXAMPLE:

---

```
#           elgrp  loadno  histno  curvel  curang  cuhref  curoough  type      x0
DROPS_LOAD  sea1     110     250      0.7     0.785    0.5     1e-4    IRREGULAR  10.0
#
#   y0    wavang  Tp   Hs   dt   tdur   tstart   seed   spec   curprof   pkdness   spread
#   0.0    0.785   9.0   7.0   0.5  10800    1.0     123     1    STANDARD   1.0      SHORT
#
#   ndir   spreadpar
#   8       2
```

---

### Irregular wave loading with DNV-RP-F109 current profile

A logarithmic current profile that takes into account current-wave interaction is applied for **CURPROF=RPF109**. This is done by replacing the user-defined parameter **CUROUGH** by an equivalent seabed roughness parameter as described in the SIMLA Theory Manual. The input card has the following format:

```
DROPS_LOAD ELGRP LOADNO HISTNO CURVEL CURANG CUHREF
CUROUGH IRREGULAR X0 Y0 WAVANG TP HS DT TDUR TSTART SEED SPEC
RPF109 D50 [PKDNESS SPREAD NDIR SPREADPAR]
```

where

**D50**: Grain size  $d_{50}$ , i.e. median diameter of soil particles. Note the warning about consistent units described below (unit: L).

**PKDNESS**: Peakedness parameter, dummy for **SPEC≠2**.

Optional, default value: 1.0 (unit: -).

**SPREAD**: Wave spreading option, dummy input if **SPEC=10**. Allowable values are:

**SHORT** : Short-crested sea.

**LONG** : Long-crested sea, which is the default option.

**NDIR**: Number of directions for short-crested sea, see  $N_\varphi$  in Eq. (3.8).

Dummy for **SPREAD=LONG**.

Optional, default value: 1 (unit: -).

**SPREADPAR**: Spreading function exponent, see  $n$  in Eq. (3.10).

Dummy for **SPREAD=LONG**.

Optional, default value: 0 (unit: -).

Be aware that the input numerical value of **D50** must be consistent with the unit applied as length dimension in the SIMLA input file. Typically, **D50**-values are tabulated in

the unit *millimetre*, while user-defined input data is commonly specified in the unit *metre*. In that case, the user must convert the numerical value of **D50** to be consistent with the length dimension unit applied in the SIMLA input file. Regarding input data verification, the equivalent seabed roughnesses computed by SIMLA will be printed to the SIMLA output file (prefix.sof) for all grid points. In the SIMLA output file, the quantity CUROUGH refers to the applied seabed roughness parameter, which is either the user-defined value or the equivalent seabed roughness computed by SIMLA.

Note that either none or all of the optional input parameters must be specified for **WAVETYPE=IRREGULAR**. It is not allowed to specify only some of the optional input parameters.

The following example defines an irregular short-crested sea state with DNV-RP-F109 current profile,

#### EXAMPLE:

---

```
#          elgrp  loadno  histno  curvel  curang  cuhref  curough  type      x0
DROPS_LOAD  sea1     110      250      0.7     0.785    0.5     1e-4     IRREGULAR  10.0
#
#  y0    wavang   Tp    Hs    dt    tdur    tstart   seed   spec   curprof   d50      pkdness
#  0.0   0.785   9.0   7.0   0.5   10800   1.0     123     1     RPF109   0.0625e-3  1.0
#
#  spread  ndir  spreadpar
#  SHORT    8      2
```

---

### 3.15.2 Second order wave loading with current

The input card has the following format for **WAVETYPE=SECONDORDER**:

<b>DROPS_LOAD ELGRP LOADNO HISTNO CURVEL CURANG CUHREF CUROUGH SECONDORDER X0 Y0 WAVANG TP HS DT TDUR TSTART SEED SPEC CURPROF [PKDNESS SPREAD NDIR SPREADPAR]</b>
--

where the inputs are identical to those described above for **WAVETYPE=IRREGULAR** and **CURPROF=STANDARD** is the only allowable option.

### 3.15.3 Wave kinematics read from file with current

The input card has the following format for **WAVETYPE=FROMFILE**:

<b>DROPS_LOAD ELGRP LOADNO HISTNO CURVEL CURANG CUHREF CUROUGH FROMFILE X0 Y0 WAVANG DT TDUR TSTART</b>
---

where

**X0**: global x-coordinate of the wave elevation origin (unit: L).

**Y0**: global y-coordinate of the wave elevation origin (unit: L).

**WAVANG**: Wave main direction measured by angle relative to the global x-axis, see  $\varphi_0$  in Eq. (3.8) and Fig. 3.1 (unit: rad).

**DT**: Time step  $\Delta t$  used for generation of wave kinematics and fixed pipe hydrodynamic loads (unit: T).

**TDUR**: Duration  $T_{dur}$  of wave and load time series measured from **TSTART** (unit: T).

**TSTART**: Start time  $T_{start}$  for generation of wave kinematics and fixed pipe hydrodynamic loads (unit: T).

The usage of the option **FROMFILE** allows analysis with kinematics obtained from a separate tool. The intended work-flow is as follows:

- First run the analysis with pre-generated time series of wave kinematics, using either the option **IRREGULAR** or **SECONDORDER** for **WAVETYPE**. This will result in wave kinematics stored on file, on the hdf5 binary format.
- Edit the wave kinematics stored on file, replacing time series as desired. This substitution of the file contents can be done in, e.g., Python or Matlab.
- Run the analysis again, replacing **WAVETYPE** with the option **FROMFILE**. Note that the values of **DT** and **TDUR** should be consistent with the new kinematics time series. Using the same values in both the original and replaced time series is recommended, as it will make the substitution of file contents easier.

### 3.15.4 General comments

The full wave kinematics times series are generated at time **TSTART**. Consequently, input data changes in restart analyses will not have any effect if the restart time is larger than **TSTART**.

The time increment **DT** used for generation of the time series should be able to capture the shortest wave period at the seafloor by e.g. 10 points per wave period. Using a too small time resolution **DT** is not recommended due to the significant CPU time and memory consumption.

The water depth is at each grid point calculated based on the vertical distance from the grid point to the first sea element in **ELGRP**. The vertical level of the sea element is computed by averaging over the 4 element nodes.

The load scale factor defined by **HISTNO** is applied on the hydrodynamic excitation load, not the wave kinematics. Also the drag and added mass loads due to pipe motion

will be equal to zero when the scale factor is zero. The radial added mass coefficient  $C_{A,n}$  is scaled as follows,

$$C_{A,n} = f(t) \cdot (C_{M,n} - 1.0) \quad (3.4)$$

where  $f(t)$  is the load scale factor and  $C_{M,n}$  is the radial mass coefficient. To obtain zero tangential added mass the user must define  $C_{M,t}=1.0$  in the **ELPROP** card.

The validity of Stokes wave theory (either linear or second order) is checked against the Ursell criterion,

$$\frac{k_z H_s}{2\sqrt{2} (k_z d)^3} \leq 0.2 \quad (3.5)$$

and the Miche criterion,

$$\frac{k_z H_s}{2\pi\sqrt{2} \tanh(0.875 k_z d)} \leq 0.142 \quad (3.6)$$

where  $H_s$  is the significant wave height and  $d$  is the water depth.  $k_z$  is the wave number corresponding to the zero-crossing frequency  $\omega_z$ , which are linked by the dispersion relation according to,

$$\omega_z^2 = g k_z \tanh(k_z d) \quad (3.7)$$

where  $g$  is the gravity acceleration and the zero-crossing frequency  $\omega_z$  is calculated based on the zero and second order moments of the spectrum for the wave elevation. A warning message will be printed to screen and the SIMLA log file (prefix.slf) if one or both of the criterion in Eqs. (3.5) and (3.6) are violated.

In contrast to Section 3.55, short-crested sea can be modelled in which the wave elevation is expressed as,

$$\eta(x, y, t) = \sum_{i=1}^{N_\omega} \sum_{j=1}^{N_\varphi} A_{ij} \sin [\omega_i t + k_i (x - x_0) \cos(\varphi_j - \varphi_0) + k_i (y - y_0) \sin(\varphi_j - \varphi_0) + \epsilon_{ij}] \quad (3.8)$$

where  $\epsilon_{ij}$  is the random phase angle drawn from a uniform distribution over  $[-\pi, \pi]$ .  $k_i$  is the wave number,  $\omega_i$  is the wave frequency and  $\varphi_i$  is the wave direction defined in Fig. 3.1. The wave main direction is denoted  $\varphi_0$ . The origin of the wave elevation is located at global coordinates  $(x_0, y_0, z_s)$  where  $z_s$  is the node-averaged vertical level of the first element in **ELGRP**.

The wave component amplitude in Eq. (3.8) is given by,

$$A_{ij} = \sqrt{2S_\eta(\omega_i)D(\varphi_j - \varphi_0)\Delta\omega\Delta\varphi} \quad (3.9)$$

in which  $S_\eta(\omega)$  is the wave spectrum, see Eqs. (3.58) and (3.59). The wave spreading function is expressed as,

$$D(\varphi) = K_n [\cos \varphi]^n \quad , \quad -\frac{\pi}{2} \leq \varphi \leq \frac{\pi}{2} \quad (3.10)$$

where  $n$  is coincident with **SPREADPAR** above.  $K_n$  is a scaling factor found from the condition,

$$\int_{-\pi/2}^{\pi/2} D(\varphi) d\varphi = 1.0 \quad (3.11)$$

which yields,

$$K_n = \frac{1}{\sqrt{\pi}} \frac{\Gamma(1 + \frac{n}{2})}{\Gamma(\frac{1}{2} + \frac{n}{2})} \quad (3.12)$$

The inverse Fast Fourier Transformation, FFT, method is applied for calculation of the time series. This method requires that the time step increment  $\Delta t$ , the frequency increment  $\Delta\omega$ , the number of time steps  $N_t$  and the number of frequency components  $N_\omega$  fulfill Eqs. (3.60)–(3.62).

Regular wave loading is generated using a one component wave spectrum, **SPEC**=10. The wave time series is pregenerated using FFT similar to irregular waves. The spectrum is defined using a set of four wave frequencies and corresponding wave spectrum values, see Fig. 3.17.

The center of the spectrum is defined by the regular wave period as  $\omega_{reg} = \frac{2\pi}{T}$ , where  $T = T_p$ . The wave energy,  $S(\omega)$ , is given by the regular wave height,  $H = H_s$ ,

$$S(\omega) = \left(\frac{H}{2}\right)^2 0.5 \frac{1}{\Delta\omega} \quad (3.13)$$

The width of the wave spectrum is defined by the simulation used in the inverse FFT,

$$\Delta\omega = \frac{2\pi}{T_{dur,FFT}} \quad (3.14)$$

where  $T_{dur,FFT}$  is the simulation time  $T_{dur}$  scaled to fulfill Eqs. (3.60)–(3.62).

The set of four wave frequencies below ensure that the wave energy is centered at  $\omega_{reg}$ .  $fac$  is a small value giving wave frequencies increasing in value,  $fac= 0.00005$ ,

$$\omega_1 = (\omega_{reg} - \frac{\Delta\omega}{2})(1 - fac) , \quad S(\omega_1) = 0.0 \quad (3.15)$$

$$\omega_2 = (\omega_{reg} - \frac{\Delta\omega}{2}) , \quad S(\omega_2) = \left(\frac{H}{2}\right)^2 0.5 \frac{1}{\Delta\omega} \quad (3.16)$$

$$\omega_3 = (\omega_{reg} + \frac{\Delta\omega}{2}) , \quad S(\omega_3) = \left(\frac{H}{2}\right)^2 0.5 \frac{1}{\Delta\omega} \quad (3.17)$$

$$\omega_4 = (\omega_{reg} + \frac{\Delta\omega}{2})(1 + fac) , \quad S(\omega_4) = 0.0 \quad (3.18)$$

*Note:* The frequency resolution of the wave spectrum is dependent on the length of the simulation,  $T_{dur}$ . If the frequency resolution is to coarse, this may influence the frequency of the single sine component, see Fig. 3.17.

*Note:* If a parameter study is performed, it is suggested to keep the wave kinematic input the same for all cases. If time duration **TDUR**,  $T_{dur}$ , and the time step **DT** is changed, this may affect the generated wave frequency components and thereby the wave kinematics time series, see Eqs. (3.60)–(3.62).

*Note:* For **CURPROF=RPF109**, the current and the wave directions must be identical, i.e. **CURANG=WAVANG**.

The following example defines an irregular short-crested sea state with current loading,

---

EXAMPLE:

---

```
#          elgrp  loadno  histno  curvel  curang  cuhref  curough  type      x0
DROPS_LOAD sea1     110      250      0.7    0.785     0.5   1e-4    IRREGULAR  10.0
#
#  y0    wavang  Tp    Hs    dt    tdur   tstart   seed   spec   spread  ndir  spreadpar
  0.0  0.785   9.0   7.0   0.5   10800   1.0     123     1    SHORT     8      2
```

---

the example below defines a regular wave combined with current,

---

EXAMPLE:

---

```
#          elgrp  loadno  histno  curvel  curang  href  crough  type      x0
DROPS_LOAD sea1     100      250      0.1   1.570796  1e-12  1e-3    IRREGULAR  0.0
#
#  y0    wavang  Tp    Hs    dt    tdur   tstart   seed   spec(regular)
  0.0  1.570796  10.   7.   0.1    512    0.0     123    10
#
```

---

and the example below defines a second order wave combined with current.

---

EXAMPLE:

---

```
#          elgrp  loadno  histno  curvel  curang  cuhref  curough  type      x0
DROPS_LOAD sea1     110      250      0.7    0.785     0.5   1e-4    SECONDORDER 10.0
#
#  y0    wavang  Tp    Hs    dt    tdur   tstart   seed   spec   spread  ndir  spreadpar
  0.0  0.785   9.0   7.0   0.5   10800   1.0     123     1    SHORT     8      2
```

---

### 3.16 ELCON - Element connectivity and properties

The elements in SIMLA are organised into element groups each having a specific name. Each group is further defined by a reference to element type and material type. By

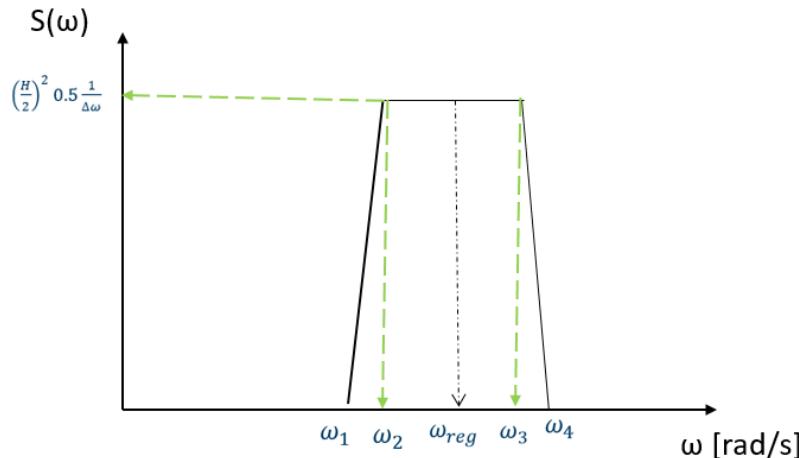


Figure 3.17: Definition of wave spectrum for regular wave

the **ELCON** command, the user defines the element group, references to element and material types as well as the element connectivity. An overview of element types, associated allowed material types and number of element nodes for each element type is found in Table 3.1. Note that for some contact elements, only the master node ID of the element connectivity matrix is given. The following format is applied:

```
ELCON ELGR ELTY MATNAME ELID NOD1 [NOD2 NOD3 ... NODN]
[REPEAT N NELINC NODINC]
```

where:

**ELGR**: Element group name.

**ELTY**: Element type. The complete list of available element types is given in Table 3.1 below.

**MATNAME**: For the contact elements CONT125, CONT126 and CONT128: Name of contact surface. For BODY502 the value must be **NONE**. For all other elements: material name. See the **COSURFPR** command in Section 3.8.

**ELID**: First element ID number

**NOD1**: Node 1 ID

**NOD2**: Node 2 ID, optional depending of element type

**NOD3**: Node 3 ID, optional depending of element type

**NODN**: Node n ID, optional depending of element type

If the **REPEAT** command is introduced, the previous sequence is repeated:

**N**: Number of repeats.

**NELINC**: Element increment.

**NODINC**: Node increment.

The element types that are allowed for in the version 3.24.0 of the program are shown in Table 3.1. The material is described in the **MATERIAL** card and the element type is the **ELPROP** card.

Table 3.1: Overview of SIMLA element library.

Type	Name	Description	Element prop- erty type	Material type	Nodes	Elforce system	Elforce - Confor/ Stress units
31	PIPE31	3D beam constant axial strain and torsion	PIPE	LINEAR	2	Local	F,FL,FL <sup>-2</sup>
33	PIPE33	3D beam constant axial strain and torsion	PIPE	ELASTOPLASTIC	2	Local	F,FL,FL <sup>-2</sup>
34	PIPE34	3D beam interpolation as for PIPE31 plus 6 ovalization dofs at midpoint	PIPE	LINEAR	2(3)	Local	F,FL,FL <sup>-2</sup>
36	PIPE36	3D beam cubic axial strain and constant torsion	PIPE	ELASTOPLASTIC	2(4)	Local	F,FL,FL <sup>-2</sup>
37	PIPE37	3D beam cubic axial strain and torsion	PIPE	LINEAR	2(4)	Local	F,FL,FL <sup>-2</sup>
39	PIPE39	3D beam cubic axial strain and torsion	PIPE	ELASTOPLASTIC	2(4)	Local	F,FL,FL <sup>-2</sup>
42	COMPIPE42	3D beam constant axial strain and torsion	COMPIPE	RESULTANT	2	Local	F,FL,FL <sup>-2</sup>
111	CABLE111	3D cable constant axial strain	CABLE	LINEAR, EPCURVE, HY-CURVE	2	Global	F,FL <sup>-2</sup>
124	CONT124	3D Stinger contact element	ROLLER	CONTACT	1(3)	Local	F,FL <sup>-1</sup>
125	CONT125	3D Seabed contact element (y force relative to route)	SOILCONTACT	CONTACT	1	Local	F,FL <sup>-1</sup>
126	CONT126	3D Seabed contact element (x and y force relative to last contact)	SOILCONTACT	CONTACT isotropic friction available	1	Local	F,FL <sup>-1</sup>
128	CONT128	3D User defined contact spring element	USERCONTACT	SPRING	1	Local	F,FL <sup>-1</sup>
130	CONT130	3D Bellmouth contact element	BELLMOUTH	CONTACT ISOCONTACT	1(3)	Local	F
136	SPRING136	3D linear spring element		LINSPRING	2	Local	F,FL
137	SPRING137	3D non-linear spring element	GENSESPRING	GENSESPRING	2	Local	F,FL
150	SEA150	Sea element		SEA	4		
152	CONT152	3D Pipe in body element	BELLMOUTH	CONTACT ISOCONTACT	1(2)	Local	
153	CONT153	3D body and pipe element	CONTBODY	ISOKXYCONTACT	3	Local	F

*Continued on next page*

Type	Name	Description	Element prop- erty type	Material type	Nodes	Elforce system	Elforce - Confor/ Stress units
164	CONT164	3D Stinger ca- ble/roller contact element	ROLLER	CONTACT ISOCONTACT ISOKXYCONTACT	1(3)	Local	F, F/L
170	CONT170	3D Bellmouth ca- ble/pipe contact el- ement	BELLMOUTH	CONTACT	1(3)	Local	F
342	HSHEAR342	3D Coating (bflex) element	COAT	CONCRETE & EPCURVE	4	Local	
502	BODY502	3D Body element	BODY	NONE	1	Local	

### 3.16.1 PIPE

PIPE31-33 are 2-noded beam elements, differing with respect to material model and applied displacement interpolation functions. From a user point of view they are all 2-noded elements even if the total number of element nodes shown in the parenthesis in the above table may differ, which means that the internal nodes are handled by the program. In order to consistently handle the pressure effect, it is necessary to work on element level. Hence the beam elements in SIMLA are basically PIPE elements assuming thin walled tubular cross-section with constant radius and thickness along each element. Accordingly, the concept 'PIPE' has been used rather than 'BEAM'. As a consequence of this, the pipe geometry properties are linked to the element concept, basically assuming that all PIPE elements in an element group have the same properties. The PIPE34 element type is a modified beam element with 6 internal DOFs that capture ovalization based on Von Karman theory. The element is made to allow for analysis of the reduced bending stiffness effect of bends. It requires curvature initialization by applying the **INISTR** command, see Section 3.34, then activating the von Karman terms when this procedure is completed via the **ELTIME** feature, see Section 3.23.1. However, overruling this concept is allowed for by using the **SIMLA** or **NODPROP** data cards, see Sections 3.48 and 3.40. The **NODPROP** data card is made to ease modelling of bellmouth/bend stiffener geometries.

### 3.16.2 COMPIPE

COMPIPE42 is a pipe element based on resultant material curves that may be either hyperelastic or elastoplastic. This is to allow modelling of e.g. bending restrictors or flexible pipes/umbilicals.

### 3.16.3 CABLE

CABLE111 is a 3D 6 DOF cable element allowing elastic and non-linear elastic stiffness to be described.

### 3.16.4 CONT

CONT164 is a 3-noded roller element which basically assume that contact is obtained between a user defined cylinder attached to node 1 and an arbitrary position between two pipe or cable nodes (node 2 and 3). Since the contact search may become time consuming, constraints may be introduced in the contact search by the **CONTINT** card, see Section 3.5. Note that only the master node is included in the connectivity input, as the other ones are set by the program.

The same governs for the pipe in pipe/bellmouth contact elements CONT130 and CONT170 which require that there are two node and element systems, one for the pipe node and one for the other bellmouth/pipe structure. The master node will be the pipe node of the master node system which may come in contact with two slave bellmouth structure/pipe/cable nodes of the second node system.

CONT125 and CONT126 are 1-noded contact elements which are linked to the contact surface definition, see Section 3.8. The difference between CONT125 and CONT126 is that the latter measures friction transverse displacement from the position where last contact was obtained whereas the first ones measure transverse displacement relative to the route described on the seabed profile file, or relative to the "as layed" route in case of a **STATIC-SIMLA** analysis. The CONT126 element includes torsion moment due to axial rotation and transverse displacement. For CONT125, the torsion moment is not included and these elements should only be used in connection with route lay simulation as defined by the **SIMLA** card, see Section 3.48. Since the contact search may become time consuming constraints may be introduced in the contact search by the **CONTINT** card, see Section 3.5. CONT128 is a 1 noded contact element which will work according to the spring characteristics given using the seabed geometry as a reference in the local  $y$ -direction only. In the local  $x$ - and  $z$ -direction the reaction forces are calculated according to the equilibrium state at which they were activated, see Section 3.5. This is to enable modelling of burial and rockdumping effects.

CONT152 is a two noded contact element to be applied for contact between pipe and body element. One node per element (the body node) is specified in the connectivity input. The second node is a node on the pipe. The pipe nodes being closest to the body object in the initial configuration is selected. Note that this element is linked to a geometrical object linked to the body element BODY502. If an eccentricity is given for the body element. The same eccentricity must be given for the contact element in order to capture the same eccentricity and associated bending moments.

CONT153 is a three noded contact element with 18 DOFs applied to describe contact between pipe elements and a three-dimensional body. Only the body node is specified in the **ELCON** card. The two other nodes belong to the pipe and are found through

the slave elements in the **CONTINT** card. The body geometry is linked to the contact element by means of the specified body node and the geometry name in the **ELPROP** card for the BODY502 element. The rigid triangle in Fig. 3.18 is the basic element applied for modelling of the three-dimensional body. Contact is assumed to occur at the green-colored edge and corner surfaces, while no contact can be detected at the top and bottom surfaces. A continuous description of the contact geometry is achieved because the triangles have equal edge and corner radius, denoted **RB** in Fig. 3.18. The coordinates, the connectivity and the radius **RB** are defined through the **GEOM** card in Section 3.28, where only the option **FROM \_ FILE** and **NENOD** = 3 are allowed. Due to topology reasons, one must define  $N$  identical contact elements to handle the presence of  $N$  simultaneous contact points between the body and the pipe. Further, due to the logic used for the friction algorithm, the contact points must always be separated by at least 2 edges located on distinct triangular geometry elements, see Fig. 3.19 and Fig. 3.20. Note that if an eccentricity is given for the body element, the same eccentricity must be given for the contact element in order to capture the same eccentricity and associated bending moments.

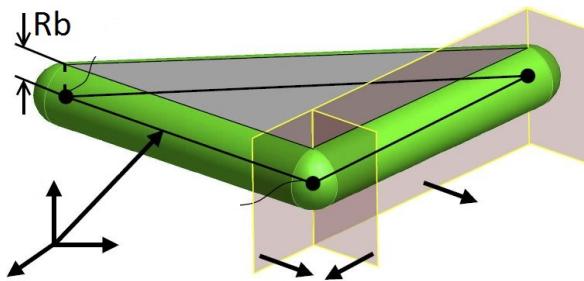


Figure 3.18: Rigid triangle with contact domain in green color.

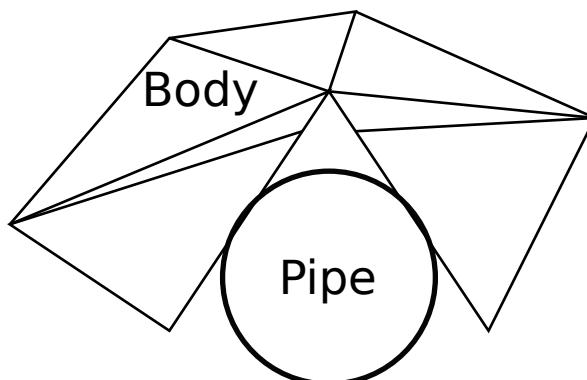


Figure 3.19: No failure because contact points are separated by at least 2 edges located on distinct triangular geometry elements.

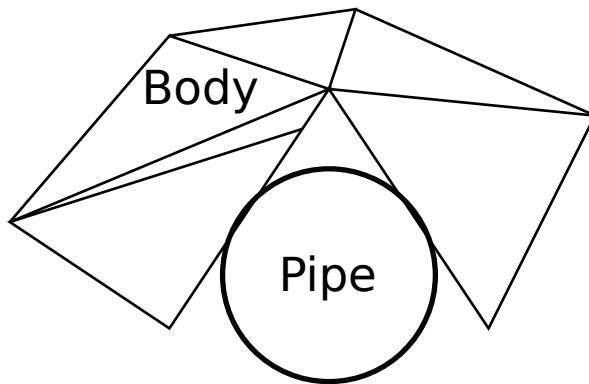


Figure 3.20: Failure in which contact is detected only at one contact point.

### 3.16.5 SPRING

SPRING137 is a 2-noded spring element where it is possible to specify mass, damping and arbitrary material curves for 6 nodal DOFs. Eccentricity is also allowed for. Hence, it can be used to model vessel properties including water plane stiffness, eccentricities to the vessel pipe node and tensioner effects. The element may be activated at a specified load step in the different DOFs, and allows for various coupling between the DOFs relevant for pipe-in-pipe and friction modelling, see Sections 3.23.10 and 3.37.20.

SPRING136 is a 2-noded spring element where it is possible to specify linear stiffness, mass and damping for 6 nodal DOFs.

### 3.16.6 SEA

SEA150 is an element that simulates the sea properties. In SIMLA, all physical effects is forced into the concept of elements and nodes, and so also the sea which is represented by an arbitrary set of 4-noded shell elements. The application of buoyancy, current and wave force effects requires that a contact interface is defined between the structural element and the sea element groups, see Section 3.5. Definition of current and wave forces are governed by the **SEALO**, **CURLOAD** and **WAVELOAD** cards, see Sections 3.47, 3.10 and 3.55. Surface elevation effects can be handled by prescribing prescribed displacement of the wave surface node which is assumed to have mean position at  $z = 0$ , see Section 3.4, i.e. it is checked whether the pipe node is submerged relative to the elevated surface.

### 3.16.7 HSHEAR

HSHEAR342 is a 4-noded 14 DOF coating element to model concrete coating, the 2 first nodes contain 6 DOFs each and is to be connected to the pipe centreline nodes whereas the two others are 1 DOF nodes to describe the relative displacement between concrete and pipe.

*Note:* The element has not been validated for cyclic load conditions.

The 1 DOF nodes must be defined by the **POLAR** option in Section 3.39. This will orient the nodal coordinates so that DOF number 1 is directed along the coating element. Further, it is not necessary to distinguish between the local and global options in the **BONCON** card, i.e. the global option can be applied for modelling fixed axial DOFs along the concrete element. The remaining DOFs 2-5 are excluded from the concrete element.

The material option **EPCURVE** in Section 3.37.3 must be applied for defining the shear stress versus relative displacement relation at the interface between the steel pipeline and the concrete coating. The material name is linked to HSHEAR342 through the element property type **COAT** in Section 3.23. The material curve units are defined in Table 3.2.

The material option **CONCRETE** in Section 3.37.21 must be applied for modelling of the coating material response. This material is linked to HSHEAR342 through the **ELCON** card in the standard way. Only the coating longitudinal stress is modelled.

### 3.16.8 BODY

BODY502 is a one node element, that can be connected to a geometry, see Section 3.28. Properties for mass, added mass, rotational mass may be given. Thus the element can be used to simulate hydrodynamic bodies. User defined coefficients for drag and added mass may be given as a function of orientation and height above seabed, see Section 3.33. Note the fairly complex definition of eccentricities that are allowed for this element, all referred to the local element coordinate system defined, see Section 3.22. First note that the element eccentricities given in Section 3.18 defines the eccentricity vector spanning between the coordinate node it is attached to and the element node from which the geometry is defined, e.g. by using Section 3.28. All physical quantities needs to be transformed to the COG so therefore the eccentricity vector from the element node where the geometry is defined to COG is given in Section 3.23. The total element eccentricity is therefore represented by the sum of these two eccentricities. With reference to Section 3.33 it is also possible to include the relative eccentricity (from COG) of the hydrodynamic drag and mass forces, respectively

EXAMPLE: \_\_\_\_\_

```
# Elcon input:
# Pipe section
#-----
#      group      elty      material   elID    n1    n2
ELCON  ormpipe1  pipe31      pipemat1     1      1    2
#          n j k
```

```

REPEAT 340 1 1
#
#      group      elty      material   elID    n1    n2
ELCON  ormpipe2  pipe31      pipemat1  341   341   342
#      n j k
REPEAT 100 1 1
#
#      group      elty      name of surface   elID    n1
ELCON  seabed    cont126     cosurf1        1001    1
#      n j k
REPEAT 200 1 1

```

---

### 3.17 ELDAMP - Element damping properties

Element damping can be specified for PIPE, SPRING and CONT elements. Damping constants  $c_i$  are given for each DOF  $i$ . The unit depends on element type, in which the damping constants for PIPE and soil CONT elements are per unit length (units:  $\text{ML}^{-1}\text{T}^{-1}$  for translation and  $\text{MLT}^{-1}$  for rotation), while lumped constants are applied for SPRING, CONT124, 164, 130 and 170 (units:  $\text{MT}^{-1}$  and  $\text{ML}^2\text{T}^{-1}$ ). For two-noded elements with physical length, the damping load relation for DOF  $i$  at element nodes A and B reads:

$$\begin{bmatrix} S_{iA} \\ S_{iB} \end{bmatrix} \begin{bmatrix} c_i \frac{1}{2} & -c_i \frac{1}{2} \\ -c_i \frac{1}{2} & c_i \frac{1}{2} \end{bmatrix} \begin{bmatrix} \dot{v}_{iA} \\ \dot{v}_{iB} \end{bmatrix}$$

In addition, a Rayleigh damping matrix which is proportional to the material stiffness matrix and the mass matrix can be given. The **ELDAMP** input data is only used for dynamic analyses.

The following format is applied:

**ELDAMP GRNAME TYPE ...**

where

**GRNAME**: Element group name.

**TYPE**: Element property type which may have three options:

**BEAM** : Covers PIPE, SPRING and BODY502 element types.

**CONTACT** : Covers the CONT element types and only translation DOFs are involved.

**RALEIGH** : Can be applied for structural elements only, i.e. CABLE or BEAM elements. Note that CABLE elements only allow for Rayleigh damping.

The damping constants for the BEAM and RALEIGH options refer to the element coordinates, which may be eccentric relative to the element nodes if the **ELECC** card in Section 3.18 is applied. For the CONTACT option, the damping force acts at the contact point and is present only when the surfaces are in contact. The damping constants refer to the contact point coordinate system where the z-direction always coincides with the normal direction of the contacting surfaces.

### 3.17.1 BEAM

For **TYPE = BEAM** the following format applies:

**ELDAMP GRNAME BEAM C1 C2 C3 C4 C5 C6**

where

- C1**: Damping constant in element x-direction.
- C2**: Damping constant in element y-direction.
- C3**: Damping constant in element z-direction.
- C4**: Rotational damping constant about element x-axis
- C5**: Rotational damping constant about element y-axis
- C6**: Rotational damping constant about element z-axis

The damping constants **C1-C6** are set at analysis start-up and cannot be changed during analysis restarts.

### 3.17.2 CONTACT

For **TYPE = CONTACT** the following format applies:

**ELDAMP GRNAME CONTACT C1 C2 C3**

where

- C1**: Contact point damping constant in x-direction tangential to contact interface.
- C2**: Contact point damping constant in y-direction tangential to contact interface.
- C3**: Contact point damping constant in z-direction normal to contact interface.

The damping constants **C1-C3** can be changed by the user during analysis restarts.

### 3.17.3 RALEIGH

The following format is applied for **TYPE = RALEIGH**:

**ELDAMP GRNAME RALEIGH ALFA1 ALFA2 [UPDATE=VALUE]**

where

**ALFA1**: Rayleigh damping factor for mass matrix contribution.

**ALFA2**: Rayleigh damping factor for material stiffness matrix contribution.

**UPDATE=VALUE**: **UPDATE=** is a character string and **VALUE** is an integer equal to either **0** or **1**. For **VALUE=0** which is default, the initial material stiffness matrix is applied, and for **VALUE=1** the updated stiffness matrix is applied. The option is only available for PIPE33 and COMPIPE42 elements.

The damping factors **ALFA1** and **ALFA2** are set at analysis start-up for the option **UPDATE=0** and can then not be changed during analysis restarts. **ALFA1** and **ALFA2** can however be changed if **UPDATE=1** is applied. For two-noded elements, the same damping factors are applied at both element ends. Note that **ALFA1** and **ALFA2** are not dimensionless.

EXAMPLE:

---

```
#      grname    type    c1      c2      c3      c4      c5      c6
ELDAMP ormcontact contact 0.000    0.000    0.1010
ELDAMP ormpipe2   beam    0.000e0 0.0000e0 0.0000e0 0.000e0 0.00e0 0.000e0
#      grname    type    alfa1_mass alfa2_stiff
ELDAMP ormpipe2   raleigh   0.000e0 0.01000e0
```

---

### 3.18 ELECC - Element eccentricity

The **ELECC** command allows the user to define element eccentricities relative to an arbitrary element node. The general format is as follows:

**ELECC ELTYP ELID ELEND ...**  
**[REPEAT ...]**

where

**ELTYP**: Type of element, which may have the following values:

**BEAM** : allowed element types PIPE31-39, COMPIPE42, BODY502, CONT152, CONT153 and SPRING137.

**RADIUS** : allowed element type CONT164.

**STINGER** : allowed element types CONT124 and CONT164.

### 3.18.1 BEAM

The **ELECC** card format has the following format for the **BEAM** option :

```
ELECC BEAM ELID ELEND XECC YECC ZECC
[REPEAT N ELINC]
```

where

**ELID**: Element ID number.

**ELEND**: Element end number, values **1** and **2** allowed.

**XECC**: Eccentricity in x-direction, see Fig. 3.2.

**YECC**: Eccentricity in y-direction.

**ZECC**: Eccentricity in z-direction.

The **REPEAT** card has the following format for the **BEAM** option:

**N**: Number of repeats.

**ELINC**: Element increment.

For PIPE 31-33 and PIPE34-39, the element origin eccentricity is defined in global coordinates. The eccentricity vector rotates together with the element system which follows the rotation of the nodes, including a possible initial rotation defined by the **NOORIENT** card. The nodal rotations contains both rigid body and deformational rotations, meaning that the element eccentricity refers to the deformed pipe configuration.

The eccentricity is defined in global coordinates for CONT152, CONT153, BODY502 and SPRING137. The eccentricity vector rotates together with the element system which follows the rotation of the element node specified as **NOD1** in the **ELCON** card, see Section 3.16, including a possible initial rotation defined by the **NOORIENT** card.

EXAMPLE:

---

#	type	elid	end	ex	ey	ez
ELECC	beam	3000	1	71.56	0	25.000

---

### 3.18.2 STINGER

The format of the **ELECC** card is as follows for the **STINGER** option:

**ELECC STINGER ELID ELEND XECC YECC ZECC YPHI DX1 DY1 DZ1 DX2 DY2  
DZ2 [REPEAT N ELINC DS DYPHI [RADN]]**

where

**ELID:** Element ID number.

**ELEND:** Element end number. Must apply **ELEND=1**.

**XECC:** x-component eccentricity in element coordinates relative to the master node, see  $e_x$  in Fig. 3.21 (unit: L).

**YECC:** y-component eccentricity in element coordinates relative to the master node (unit: L).

**ZECC:** z-component eccentricity in element coordinates relative to the master node, see  $e_z$  in Fig. 3.21 (unit: L).

**YPHI:** Orientation angle of the roller system relative to the element system, see  $\varphi$  in Fig. 3.21. Defined as positive for rotation about the negative y-axis of the element coordinate system (unit: rad).

**DX1:** x-component eccentricity in roller coordinates for the first roller end relative to the point defined by **XECC**, **YECC** and **ZECC** (unit: L).

**DY1:** y-component eccentricity in roller coordinates for the first roller end relative to the point defined by **XECC**, **YECC** and **ZECC**, see  $dy_1$  in Fig. 3.21 (unit: L).

**DZ1:** z-component eccentricity in roller coordinates for the first roller end relative to the point defined by **XECC**, **YECC** and **ZECC** (unit: L).

**DX2:** x-component eccentricity in roller coordinates for the second roller end relative to the point defined by **XECC**, **YECC** and **ZECC** (unit: L).

**DY2:** y-component eccentricity in roller coordinates for the second roller end relative to the point defined by **XECC**, **YECC** and **ZECC** (unit: L).

**DZ2:** z-component eccentricity in roller coordinates for the second roller end relative to the point defined by **XECC**, **YECC** and **ZECC** (unit: L).

The eccentricities **XECC**, **YECC** and **ZECC** refer to the element coordinate system, see x and z in Fig. 3.21, which follows the rotation of the element node specified as **NOD1** in the **ELCON** card, see Section 3.16, including a possible initial rotation defined by the **NOORIENT** card. The remaining six eccentricities refer to the roller coordinate system, see  $x_1$  and  $z_1$  in Fig. 3.21. The roller coordinate system is obtained by rotating the element coordinates system about its negative y-axis by the magnitude **YPHI**.

The **REPEAT** card allows for modelling of stingers with constant or linearly changing radius of curvature. In order to utilize the **REPEAT** card for this purpose, the roller axis must be directed along the  $y_1$ -axis, i.e. **DX1=DX2=DZ1=DZ2=0**, which may require a specific orientation of the element or the master node by means of the **ELORIENT** and

**NOORIENT** cards, respectively. The format of the **REPEAT** card is as follows for the **STINGER** option:

**N**: Number of repeats for defining rollers  $i = \{2, 3, \dots, N\}$ .

**ELINC**: Element ID increment.

**DS**: Constant arch length increment between corresponding points on  $x_1$ -axes of rollers  $i-1$  and  $i$ . **DS** is defined positive in direction of the negative  $x_1$ -coordinate axis of roller  $i-1$  for positive **DYPHI**, see **ds** in Fig. 3.21. For negative **DYPHI**, the positive direction of **DS** is in the direction of the positive  $x_1$ -coordinate axis of roller  $i-1$  (unit: L).

**DYPHI**: Constant angle increment of roller coordinate system orientation angle **YPHI**, see  $d\varphi$  in Fig. 3.21. Defined as positive for rotation about the negative element y-axis. If the parameter **RADN** is given, **DYPHI** will be applied only for roller  $i = 2$  while the increment of **YPHI** for the remaining rollers will be set equal to **DS** divided by the linearly interpolated curvature radius (unit: rad).

**RADN**: Stinger radius of curvature between the two last rollers,  $i = N-1$  and  $i = N$ . Linear interpolation of the curvature radius is then applied between rollers  $i = 2$  to  $i = N-1$ . Optional parameter. (unit: L).

If the optional parameter **RADN** is omitted, the  $y_1$ -axes of the rollers will be placed on a curve with constant radius of curvature in the  $xz$ -plane equal to **DS** divided by **DYPHI**. If the parameter **RADN** is given, the stinger radius of curvature will be equal to **DS** divided by **DYPHI** between rollers  $i = 1$  and  $i = 2$ , and thereafter change linearly towards **RADN** between rollers  $i = N-1$  and  $i = N$ .

The example below utilizes the **ELECC STINGER** option to create a spiral with radius of curvature increasing linearly from  $\frac{ds}{d\varphi} = \frac{1.0}{0.05} = 20.0$  to 2.0:

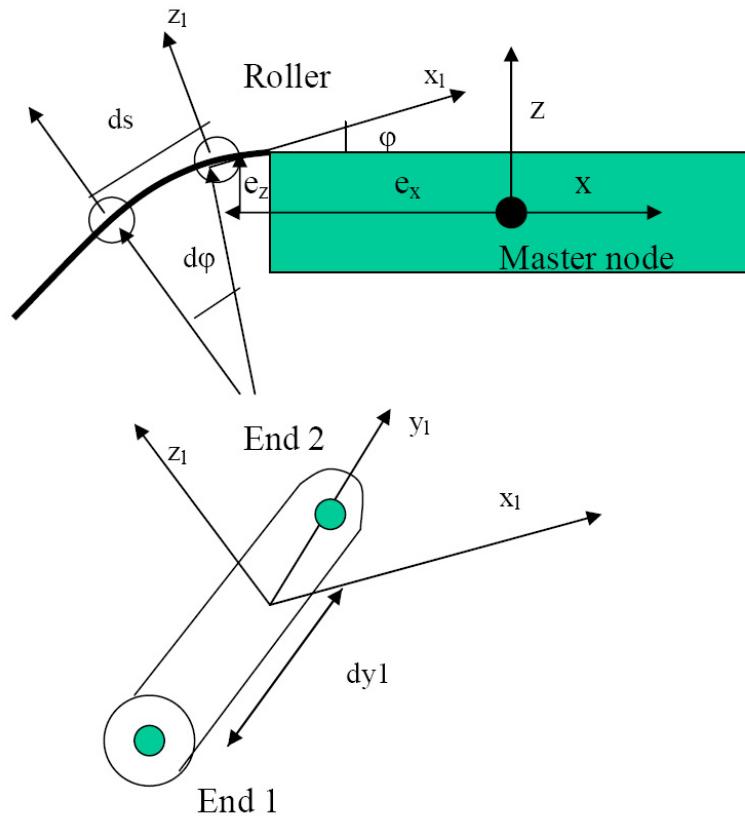


Figure 3.21: Element coordinates ( $x, y, z$ ), roller coordinates ( $x_1, y_1, z_1$ ), orientation angle  $\varphi$  of the roller coordinate system and incremental change of arch length  $ds$  and orientation angle  $d\varphi$  between rollers.

EXAMPLE:

---

```
# Elecc data:
#      type     elid    elend   xecc   yecc   zecc   yphi   dx1    dy1    dz1    dx2    dy2    dz2
ELECC stinger 1       1      0.0    0.0    0.0    0.0    0.0   -1.0    0.0    0.0    1.0    0.0
#
#          n     elinc   ds    dyphi   radn
repeat 100      1     1.0    0.05    2.0
```

---

where the resulting geometry is shown in Fig. 3.22.

### 3.18.3 RADIUS

The **ELECC RADIUS** option is used for modelling of rotation-symmetric geometries consisting of several rollers. The same functionality is available also through the **REPEAT** card for **ELECC STINGER**, however, the **RADIUS** option has increased flexibility as the roller can rotate-symmetric about each of the three coordinate axes. The **ELECC** card has the following format for the **RADIUS** option:

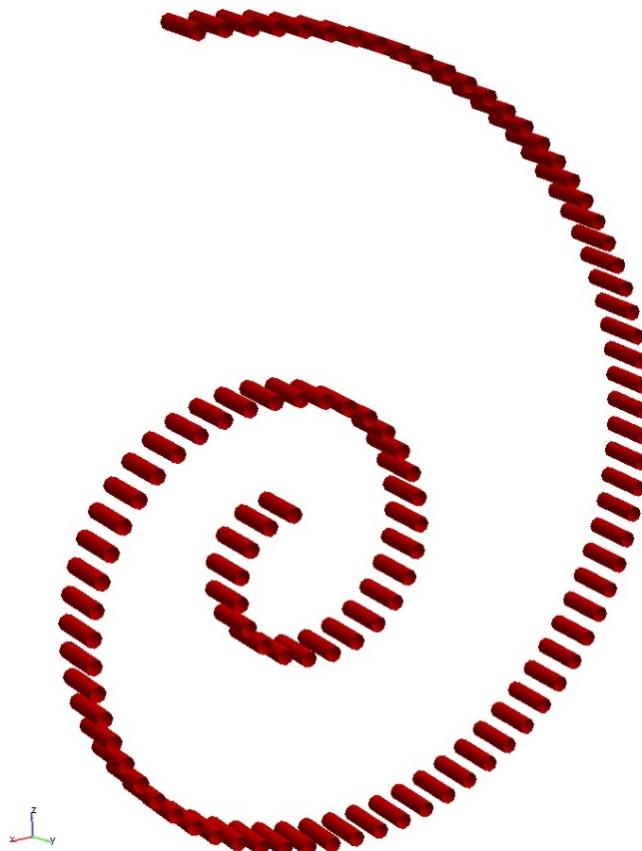


Figure 3.22: Modelling of spiral with linearly changing radius of curvature.

```
ELECC RADIUS ELID ELEND IAXIS DX1 DY1 DZ1 DX2 DY2 DZ2
[REPEAT N ELINC DPHI]
```

where

**ELID**: Element ID number.

**ELEND**: Element end number. Must apply **ELEND=1**.

**IAXIS**: The element axis which the roller end eccentricity vectors is rotated about.

The eccentricity vector for end 1 is defined by **DX1**, **DY1** and **DZ1** and by **DX2**, **DY2** and **DZ2** for end 2. The following values are allowed:

**1** : Rotation about element x-axis.

**2** : Rotation about element y-axis.

**3** : Rotation about element z-axis.

Dummy parameter if no **REPEAT** card is given or if only **N=1** repeat is applied.

**DX1**: x-component eccentricity in element coordinates for the first roller end relative to the master node (unit: L).

**DY1**: y-component eccentricity in element coordinates for the first roller end relative

to the master node (unit: L).

**DZ1:** z-component eccentricity in element coordinates for the first roller end relative to the master node (unit: L).

**DX2:** x-component eccentricity in element coordinates for the second roller end relative to the master node (unit: L).

**DY2:** y-component eccentricity in element coordinates for the second roller end relative to the master node (unit: L).

**DZ2:** z-component eccentricity in element coordinates for the second roller end relative to the master node (unit: L).

The eccentricities **DX1**, **DY1**, **DZ1**, **DX1**, **DY1** and **DZ1** refer to the element coordinate system, which follows the rotation of the element node specified as **NOD1** in the **ELCON** card, see Section 3.16, including a possible initial rotation defined by the **NOORIENT** card.

The **REPEAT** card has the following format for the **RADIUS** option:

**N:** Number of repeats.

**ELINC:** Element ID increment.

**DPHI:** Constant angle increment for rotation about **IAXIS=1,2 or 3**. Positive rotation directions are defined according to the right-hand rule (unit: rad).

Contact elements based on line-line search algorithms are known to fail for parallel lines. In contact modelling of two parallel pipelines, this issue can be avoided effectively by using the **RADIUS** option for modelling of the external pipe radius according to:

EXAMPLE: \_\_\_\_\_

```
# Elecc data
#      type    elid   elend   iaxis   dx1    dy1    dz1    dx2    dy2    dz2
ELECC radius  1        1        2     0.02   0.0   -0.3   -0.02   0.0   -0.3
#
#          n    elinc    dphi
repeat   49     1    -0.065
```

---

which defines a half circle consisting of 49 overlapping rollers as shown in Fig. 3.23. The roller with element ID 1 is located at  $z = -0.3$  and spans from  $x = 0.02$  to  $x = -0.02$ . The element IDs are increasing in the direction of negative rotation about the y-axis. All of the rollers are located approximately on the circle  $x^2 + z^2 = 0.3^2$ , and overlaps each other by half of the roller length to avoid deadzones in the contact search.

The example below defines 20 rollers as shown in Fig. 3.24 spanning from  $z = -4.0$  to  $z = 4.0$ :

EXAMPLE:

---

```
# Elecc data
#      type     elid    elend   iaxis   dx1   dy1   dz1   dx2   dy2   dz2
ELECC radius  1        1        3     3.0   0.0  -4.0   3.0   0.0   4.0
#
#          n   elinc   dphi
repeat   20   1       0.2
```

---

where element ID 1 is located at coordinate  $x = 3.0$  and  $y = 0.0$ . The other rollers are also located on the circle  $x^2 + y^2 = 3^2$  with the element IDs increasing in the direction of positive rotation about z-axis.

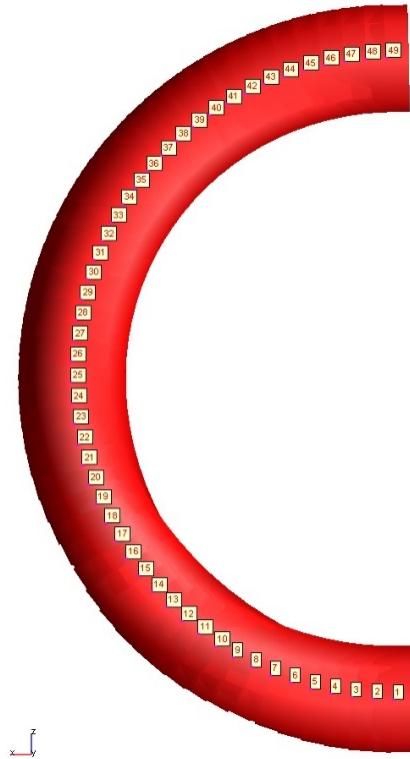


Figure 3.23: Rotation-symmetric geometry for pipe-pipe external contact interaction modelled by the **RADIUS** option.

### 3.19 ELHIST - element time histories

By the **ELHIST** command the user is allowed to scale the elastic properties of elements by associating different element ID numbers to a load history, see Section 3.50. If the value of the associated time history scaling factor is positive, the internal load vector and the stiffness matrix of the element will be scaled according to the factor given. If a negative scaling factor is specified, the element will be deactivated, i.e. no contributions from stiffness, damping and inertia will be included for the element. This

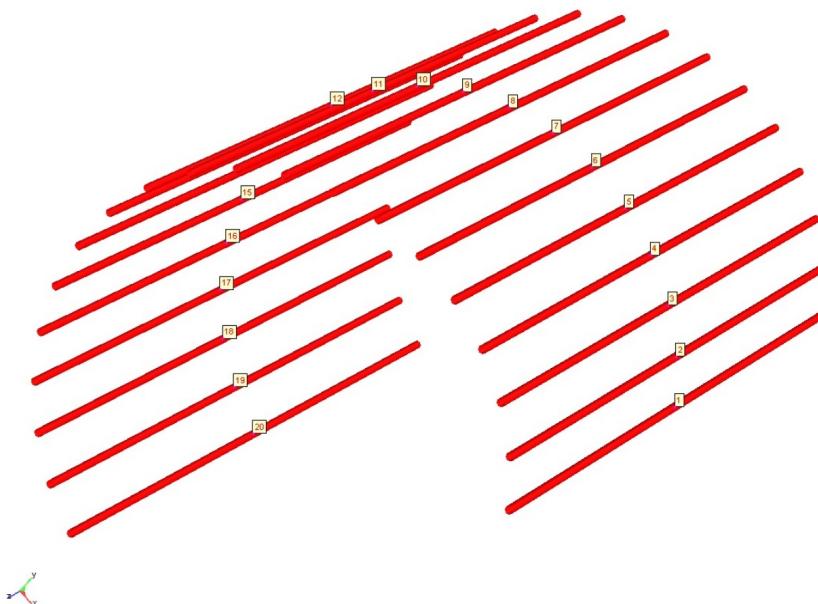


Figure 3.24: Rotation-symmetric geometry modelled by the **RADIUS** option.

will save computing time. However, it is not allowed to start with a negative factor at time zero.

The user is allowed to define an arbitrary sequence of element deactivation (death) and element activation (birth) by sequently associating negative or positive numbers in the time history. As for now this feature only exists for element type PIPE31. Please note that in the case of element death, the equation system will become singular, unless other active elements are connected to the nodes, loosing contributions from the dead elements. This feature must therefore be used with care.

Note that it is only the element internal load vector and the element stiffness matrix applied in the global equilibrium equations that will be scaled. The stresses are not scaled, and will still be reported based on the specified Young's modulus.

The following format is applied:

```
ELHIST ELID11 ELID12 HISTNO1
...
ELIDN1 ELIDN2 HISTNON
```

where:

**ELID11**: The first element ID of sequence 1.

**ELID12**: The last element ID of sequence 1.

**HISTNO1**: The time history ID number for sequence 1.

**ELIDN1**: The first element ID of sequence n.

**ELIDN2:** The last element ID of sequence n.

**HISTNON:** The time history ID number for sequence n.

EXAMPLE: \_\_\_\_\_

```
#           Element history feature
#
#           elid1 elid2  histno
ELHIST    1      100    200
            200    300    200
#
#           Element history by thist
#       no     time    fac
THIST    200     0     1.0
          5      1.0
          6     -1.0
          10    -1.0
          11     0.5
```

The above will cause the elements 1-100 and 200-300 to have an elastic stiffness according to the nominal input given until 6 s. Then from 6 s and up to 11 s they will be removed from the system, however, being reinstalled with half stiffness from 11 s.

### 3.20 ELLOAD - Element loads

The element loads are specified by the following format:

```
ELLOAD HIST DIR ELNR1 LOAD [ELNR2 LOAD2]
[REPEAT N ELINC]
```

where

**HIST:** Load history number.

**DIR:** Element load direction relative to the element axis **1-3** = load along x-z, **4-6** = moment about x-z.

**ELNR1:** Element ID of first element.

**LOAD:** Load for first element.

**ELNR2:** Element ID of last element.

**LOAD2:** Load for last element. Linear load interpolation is applied for the intermediate elements.

If the **REPEAT** command is introduced then the previous sequence of load generation is repeated:

**N:** Number of times to repeat.

**ELINC:** Element increment.

The **ELLOAD** command applies only for the PIPE, CABLE and SPRING element types, and will be a dummy command if applied for other element types.

The following distributed loads are applied for PIPE and CABLE elements:

Value of <b>DIR:</b>	<b>LOAD1/LOAD2</b> is:
<b>1</b>	force per length along local x axis (unit: $FL^{-1}$ )
<b>2</b>	force per length along local y axis (unit: $FL^{-1}$ )
<b>3</b>	force per length along local z axis (unit: $FL^{-1}$ )
<b>4</b>	moment per length about local x axis (unit: $FLL^{-1}$ )
<b>5</b>	moment per length about local y axis (unit: $FLL^{-1}$ )
<b>6</b>	moment per length about local z axis (unit: $FLL^{-1}$ )

The following concentrated loads are applied at the first element node for SPRING elements:

Value of <b>DIR:</b>	<b>LOAD1/LOAD2</b> is:
<b>1</b>	force in local x axis (unit: F)
<b>2</b>	force in local y axis (unit: F)
<b>3</b>	force in local z axis (unit: F)
<b>4</b>	moment about local x axis (unit: FL)
<b>5</b>	moment about local y axis (unit: FL)
<b>6</b>	moment about local z axis (unit: FL)

EXAMPLE:

```
#      hist   dof   elnr1   load1   elnr2   load2
ELLOAD  100    1     3       1.0     101     2.0
```

### 3.21 ELMASS - Element mass properties

Element mass matrices,  $m_0$ , can be specified. Element mass coefficients  $m_{ii}$  are given for each DOF in a node. The unit depends on element type, for PIPE and soil CONT elements the coefficients are per unit length (units:  $ML^{-1}$  for translation and  $ML$  for rotation), for SPRING, CONT124, 130, 164, 170 lumped coefficients are given (units: M and  $ML^2$ ).

The following format is applied.

<b>ELMASS GRNAME TYPE M1 M2 M3 M4 M5 M6</b>
---

where

**GRNAME**: Element group name.

**TYPE**: Element property type which may have only one option **BEAM** so far. The **BEAM** option covers the PIPE and SPRING element types.

**M1**: Mass coefficient for local x element direction.

**M2**: Mass coefficient for local y element direction.

**M3**: Mass coefficient for local z element direction.

**M4**: Mass coefficient for rotation associated to local x element direction.

**M5**: Mass coefficient for rotation associated to local y element direction.

**M6**: Mass coefficient for rotation associated to local z element direction.

For two-noded elements, the same mass coefficients are applied in both element ends. For the PIPE elements the resulting mass matrix is added to the mass matrix obtained from the **ELPROP** dry weight data. The **ELMASS** data are only applied in dynamic analyses.

EXAMPLE:

---

```
#      grname   type m1       m2       m3       m4       m5       m6
ELMASS vessel1 beam 0.0076e0 0.0076e0 0.0076e0 1.00e0 1.000e0 1.000e0
```

---

### 3.22 ELORIENT - Orientation of elements

The **ELORIENT** command is used to define the initial orientation of the element coordinate systems, and must be given. For the PIPE and HSHEAR elements, the orientations are defined by specifying the position of the *xy*-plane of the local element system relative to the global coordinate system. This is done by defining the position vector **R** in global coordinates of one point in the local *xy*-plane, see Fig. 3.25.

For the CONT, BODY and SPRING elements, the orientation is defined by specifying a set of three consecutive Tait–Bryan angles ( $\theta_x$ ,  $\theta_y$ ,  $\theta_z$ ) in global coordinates that rotates the element coordinate system relative to the coordinate system of the node specified as **NOD1** in the **ELCON** card, see Section 3.16. The relations between the global, nodal and element coordinates are shown in Fig. 3.26. As illustrated in Fig. 3.26, an initial nodal rotation specified by the **NOORIENT** card in Section 3.41 will also give a corresponding rotation of the element system relative to the global coordinate system.

Seabed contact elements shall be oriented with the initial *x*-axis coincident with the initial *x*-axis of the attached pipe element. Otherwise, the local *x*- and *y*-forces will not be directed along and transverse to the pipeline, respectively, and the local *x*-moment will not be acting about the pipeline *x*-axis, see Sections 3.37.8 and 3.37.11.

During the analysis, the element coordinate system for PIPE and HSHEAR elements follows the rotation of the end nodes in an averaged sense, while the element coordinate system for the CONT, BODY and SPRING elements follows the rotation of the node specified as **NOD1** in the **ELCON** card.

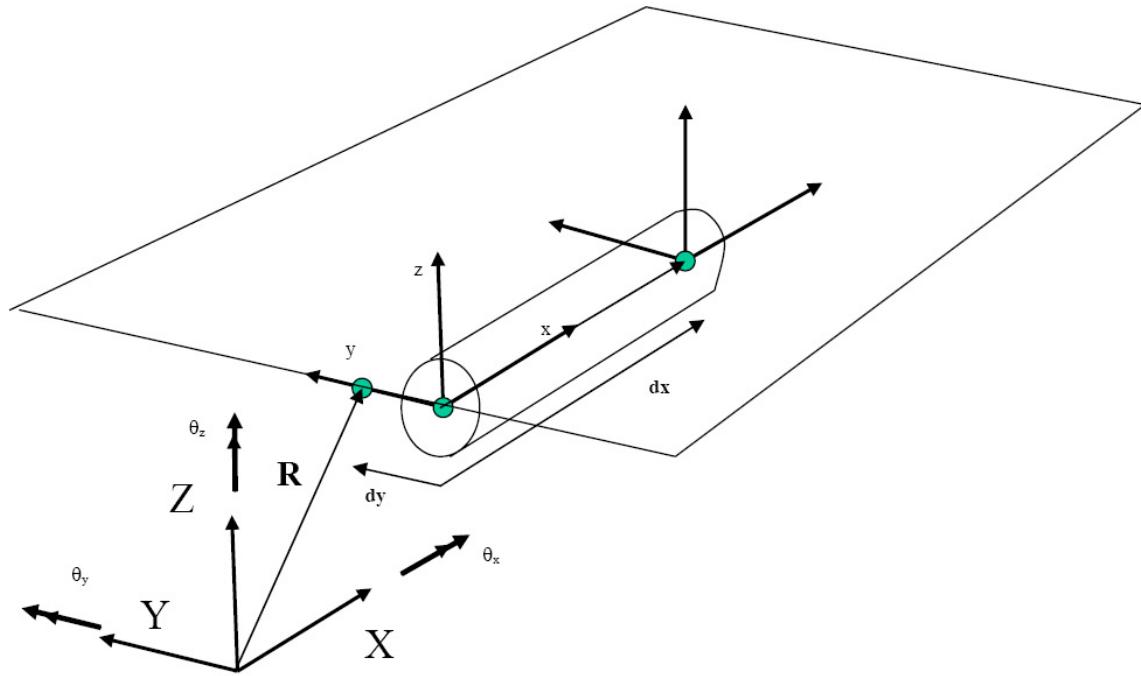


Figure 3.25: **ELORIENT** command for PIPE and HSHEAR elements.

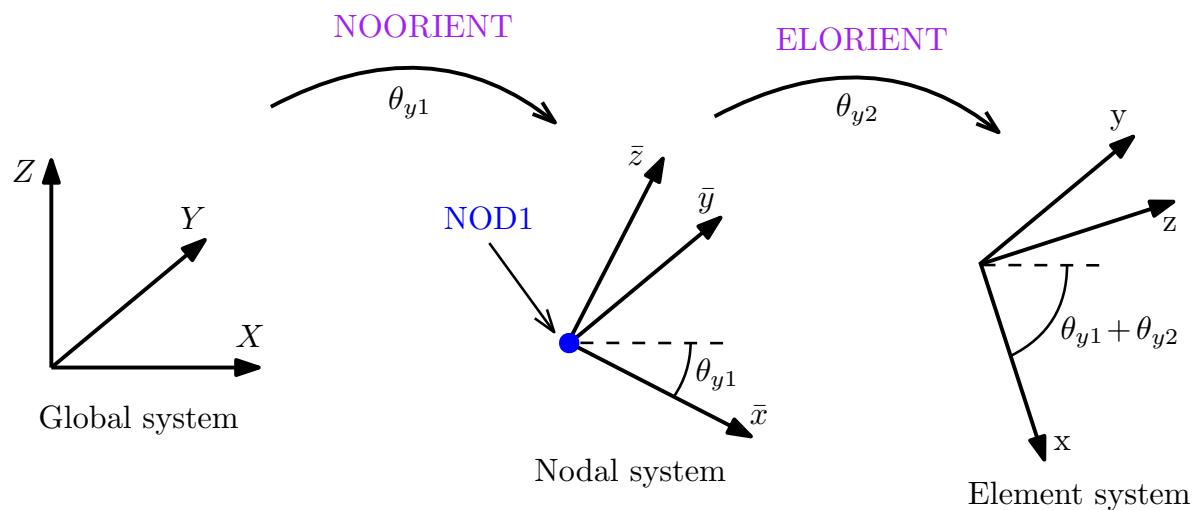


Figure 3.26: **ELORIENT** command for the CONT, BODY and SPRING elements.

The format is similar to the **NOCOOR** command, except that element ID numbers are used:

**ELORIENT TYPE ELID X Y Z**  
 [.. .. .. ..]  
**[REPEAT N INC XINC YINC ZINC]**

where

**TYPE:** Type of orientation. This parameter can have the values **COORDINATES** or **EULERANGLE**.

In case of **COORDINATES** the remaining parameters have the following meaning:

**ELID:** id of element.

**X:** Global x-coordinate of the position vector **R**.

**Y:** Global y-coordinate of the position vector **R**.

**Z:** Global z-coordinate of the position vector **R**.

By including the **REPEAT** command the **ELORIENT COORDINATES** sequence is repeated **N** times:

**N:** number of repetitions

**INC:** element increment

**XINC:** Global x-coordinate increment

**YINC:** Global y-coordinate increment

**ZINC:** Global z-coordinate increment

whereas if **TYPE = EULERANGLE**:

**ELID:** id of element.

**X:** Tait–Bryan angle around local x-axis.

**Y:** Tait–Bryan angle around local y-axis.

**Z:** Tait–Bryan angle around local z-axis.

By including the **REPEAT** command the **ELORIENT EULERANGLE** sequence is repeated **N** times:

**N:** number of repetitions

**INC:** element increment

**XINC:** Tait–Bryan angle increment around local x-axis

**YINC:** Tait–Bryan angle increment around local y-axis

**ZINC:** Tait–Bryan angle increment around local z-axis

In both cases an arbitrary sequence may be given, as long as **ELEMENT** is increasing. If the difference between two consecutive nodes is larger than 1, then linear interpolation is used to create intermediate element orientation.

EXAMPLE:

---

```

#
#          type           elemntnr      x      y      z
ELORIENT COORDINATES        1       0.0    1.0    0.0
                           11      0.0    1.0   -10.0
                           21      0.0    1.0   -15.0
                           40      0.0    1.0   -19.75
#
#          n   inc   dx   dy   dz
REPEAT     3     40   0    0   -20
#
#
#          type           elementnr      tx      ty      tz
ELORIENT EULERANGLE        3001      0       0       0

```

---

### 3.23 ELPROP - Element properties

The following element property types are available:

**PIPE** : Properties for the PIPE element types

**BELLMOUTH** : Bellmouth contact properties

**CONTBODY** : Body contact properties for element type CONT153

**COMPIPE** : Properties for COMPIPE42.

**CABLE** : Properties for the CABLE element types

**ROLLER** : Roller contact properties for element types CONT124 and CONT164

**USERCONTACT** : User defined contact properties for element type CONT128. This allows the user to specify stiffness properties others than those defined in the **COSURFPR** command. Typically used for modeling of rock dumping.

**GENSESPRING** : User defined non-linear spring properties for SPRING137

**COAT** : Coating properties

**BODY** : Properties for BODY502

**SOILCONTACT** : Properties for time-controlled hyper-elastic springs. Optional element property type only allowed for CONT126.

Each element property type is defined by the following format:

**ELPROP ELGRP EPTYP ...**

where

**ELGRP**: Element group name.

**EPTYP**: Element property type as defined above.

The number and definition of the following parameters depend on the choice of element property type.

### 3.23.1 PIPE

**ELPROP ELGRP PIPE RAD TH CDR CDT CMR CMT MD MS ODP ODW RKS**  
**[PHIST MHIST ELTIME=VALUE AUTOPLAST=VALUE SHEARA=VALUE SC-**  
**FAX=VALUE SCFMY=VALUE SCFMZ=VALUE]**

**RAD:** Structural radius i.e. the mean radius of the pipe wall (unit: L).

**TH:** Structural thickness, i.e. the wall thickness (unit: L).

**CDR:** Radial drag coefficient (unit: -).

**CDT:** Tangential drag coefficient (unit: -).

**CMR:** Radial mass coefficient. Must be  $\geq 1.0$ . Normally equal to 2.0 for circular pipes which results in an added mass of  $(\text{RMADD} - 1.0)\rho_w \frac{\pi}{4} D^2$  (unit: -).

**CMT:** Tangential mass coefficient. Must be  $\geq 1.0$ . A value of 1.0 results in zero tangential added mass according to  $(\text{RMADD} - 1.0)\rho_w \frac{\pi}{4} D^2$  (unit: -).

**MD:** Dry mass (unit:  $\text{ML}^{-1}$ ).

**MS:** Submerged mass = dry mass - buoyancy mass (unit:  $\text{ML}^{-1}$ )

**ODP:** Outer diameter  $D_{op}$  (unit: L)

**ODW:** External wrapping outer diameter  $D_{ow}$  (unit: L)

**RKS:** External wrapping fraction  $\eta$  (Range: 0-1). Hence the hydrodynamic diameter that will be applied to calculate drag and mass forces will be:  $D = (1 - \eta)D_{op} + \eta D_{ow}$ . Note that the torsion rotational inertia will be calculated as  $\rho \frac{\pi}{32} D_{op}^4$

**PHIST:** External pressure and buoyancy mass history ID number, optional.

**MHIST:** Dry mass history ID number, optional.

**ELTIME=VALUE:** This option is only relevant for PIPE33 and PIPE34. For PIPE33, the activation time for the elasto-plastic material model is specified by **VALUE**, i.e. the elements behave elastically until **VALUE** is exceeded. For PIPE34, the activation time for the initial bend curvature is specified by **VALUE**. The stress free bend radius need to be initiated by the **INISTR** card and completed at time=**VALUE**, see Section 3.34, optional (unit: T).

**AUTOPLAST=VALUE:** By specifying **AUTOPLAST=1**, the elements behave elastically until the proportionality limit is reached. The feature is not active if **VALUE** $\neq$ **1**. Only available for PIPE33, optional.

**SHEARA=VALUE:** Shear deformations with a shear rigidity (shear area times shear modulus) given by **VALUE** will be applied. An infinite shear stiffness results for **VALUE** $=$ **0**. Only available for PIPE31, optional (unit:  $\text{L}^2$ ).

**SCFAX=VALUE:** **VALUE** is the stress concentration factor used when calculating the axial stress. Only available for PIPE31, optional (unit: -).

**SCFMY=VALUE:** **VALUE** is the stress concentration factor used when calculating the bending stress from moment about local y-axis. Only available for PIPE31, optional (unit: -).

**SCFMZ=VALUE:** **VALUE** is the stress concentration factor used when calculating the bending stress from moment about local z-axis. Only available for PIPE31, optional (unit: -).

The **ELTIME=VALUE** and **AUTOPLAST=VALUE** options work such that no plasticity is considered until the time specified by **ELTIME=VALUE** is exceeded. If both quantities are nonzero this is still the case, however, only elements that actually are plastified after the given time will behave plastic. This option may speed up pipeline buckling analysis with a factor 5-10.

For the **ELASTOPLASTIC** material type, **RAD** and **TH** are employed for calculation of the cross-section stiffness, whereas for other material types the stiffness is taken directly from the **MATERIAL** input card in Section 3.37.

Regarding contact search, the inner diameter of the pipe is set equal to  $2 * \text{RAD} - \text{TH}$ . The outer contact diameter  $OD_c$  is taken as the maximum of the outer pipe diameter and the wrap diameter according to  $OD_c = \max(\text{ODP}, \text{ODW})$ .

#### EXAMPLE:

---

```
#      name      eptyp   rad      th      CDr   Cdt   CMr   CMt   wd   ws   ODp   ODw   rks
ELPROP  ormpipe1  pipe    0.3221  0.0346  0.8   0.1   2.0   1.1   5.6   1.7   0.67  0.67  0.5
#
# phist  mhists eltime=value autoplast=value sheara=value scfax=value scfmy=value
100      200     eltime=10.0    autoplast=1       sheara=0.0    scfax=1.1    scfmy=1.5
#
# scfmz=value
scfmz=1.5
```

---

### 3.23.2 COMPIPE

**ELPROP ELGRP COMPIPE RADE RADI CDR CDT CMR CMT MD MS ODP ODW RKS [PHIST MHIST ELTIME=VALUE AUTOPLAST=VALUE TCURV=VALUE]**

**RADE:** External radius where external pressure acts (unit: L).

**RADI:** Internal radius where internal pressure acts (unit: L).

**CDR:** Radial drag coefficient (unit: -).

**CDT:** Tangential drag coefficient (unit: -).

**CMR:** Radial mass coefficient. Must be  $\geq 1.0$ . Normally equal to 2.0 for circular pipes which results in an added mass of  $(\text{RMADD} - 1.0)\rho_w \frac{\pi}{4} D^2$  (unit: -).

**CMT:** Tangential mass coefficient. Must be  $\geq 1.0$ . A value of 1.0 results in zero tangential added mass according to  $(\text{RMADD} - 1.0)\rho_w \frac{\pi}{4} D^2$  (unit: -).

**MD:** Dry mass (unit:  $\text{ML}^{-1}$ ).

**MS:** submerged mass = dry mass - buoyancy mass (unit:  $\text{ML}^{-1}$ )

**ODP:** Outer diameter  $D_{op}$  (unit: L)

**ODW:** External wrapping outer diameter  $D_{ow}$  (unit: L)

**RKS:** External wrapping fraction  $\eta$  (Range: 0-1). Hence the hydrodynamic diameter that will be applied to calculate drag and mass forces will be:  $D = (1 - \eta)D_{op} + \eta D_{ow}$

**PHIST:** External pressure and buoyancy mass history ID number, optional.

**MHIST:** Dry mass history ID number, optional.

**ELTIME=VALUE:** The activation time for the elasto-plastic material model is specified by **VALUE**, i.e. the elements behave elastically until **VALUE** is exceeded. Only available for COMPIPE42, optional (unit: T).

**AUTOPLAST=VALUE:** By specifying **AUTOPLAST=1**, the elements behave elastically until the proportionality limit is reached. The feature is not active if **VALUE**  $\neq 1$ . Only available for COMPIPE42, optional.

**TCURV=VALUE:** The activation time for the elasto-plastic bending model is specified by **VALUE**. Gives zero bending moments and zero bending stiffness until **VALUE** is exceeded. Only available for COMPIPE42, optional (unit: T).

The **ELTIME=VALUE** and **AUTOPLAST=VALUE** options work such that the elements behaves elastic until the time specified by **ELTIME=VALUE** is exceeded. If both quantities are nonzero this is still the case, however, only elements that actually are plastified after the given time will behave plastic.

When the **TCURV=VALUE** option is applied, the two other activation parameters for the elasto-plastic bending model shall be set to: **ELTIME=0.0** and **AUTOPLAST=0**.

Regarding contact search, the inner diameter of the pipe is set equal to  $2 * \text{RAD} - \text{TH}$ . The outer contact diameter  $OD_c$  is taken as the maximum of the outer pipe diameter and the wrap diameter according to  $OD_c = \max(\text{ODP}, \text{ODW})$ .

**EXAMPLE:**

---

```
#      elgrp      eptyp      rade   radi   Cdr   Cdt   CMr   CMt   md    ms    ODP   OD    rks
ELPROP restrictor compipe    0.11   0.10   0.8   0.0   2.0   1.0   0.4   0.1   0.1   0.1   0.1   1.0
#
# phist  mhist  eltime=value  autoplast=value  tcurv=value
100      200     eltime=2.0      autoplast=1       tcurv=0.0
```

---

### 3.23.3 BELLMOUTH

#### ELPROP ELGRP BELLMOUTH DP [INSIDE GEOFAC]

**DP:** Pipe diameter, inside or outside depending on the meaning of the inside parameter defined below (unit: L).

**INSIDE:** Parameter where 1 means that the master is inside the slave group of larger diameter, and any other value means that the master is outside the slave group. Note that for CONT170 **INSIDE** cannot be 1, since the master need to be a pipe outside the cable. Optional, default value: 1

**GEOFAC:** Scaling factor for the geometric stiffness. A value greater than 1.0 can be applied to stabilize the pipe-bellmouth interaction behaviour. Optional, default value: 1.0

If the master is inside the slave pipe, **DP** is the outside pipe diameter of the master pipe. Otherwise **DP** is the inside diameter of the master pipe.

The **GEOFAC** scaling factor is only applied for the geometric stiffness associated with direction change of the contact point normal vector. The contact force at equilibrium will not be affected by **GEOFAC**.

EXAMPLE: \_\_\_\_\_

```
#      elgrp name   type      dp   inside   geofac
ELPROP bellcontact bellmouth 0.4  1        1.0
```

---

### 3.23.4 CONTBODY

#### ELPROP ELGRP CONTBODY DP [ANGPARALLEL IFULLEDGE]

**DP:** Pipe diameter (unit: L).

**ANGPARALLEL:** Control parameter for the CONT153 contact search algorithm that refers to the angle between a body edge and the pipe longitudinal axis. Optional, default value: 0.05 (unit: rad).

**IFULLEDGE:** Integer-valued control parameter for the CONT153 body contact geometry. See explanation below. Optional, default value: 0

The contact search for CONT153 is executed from the midpoint of the body edge for edge-pipe angles less than **ANGPARALLEL**, while the full body edge length is included in the contact search for edge-pipe angles larger than **ANGPARALLEL**. This parameter allows the user to avoid the well-known singularity failure for line-line based contact search algorithms that occurs in case of parallel lines. The minimum allowed value for

**ANGPARALLEL** is 0.00872 rad (0.5 degree), and will be overruled by 0.00872 rad if set below the minimum value. Note that unrealistic incremental tangential displacements may occur if **ANGPARALLEL** is set too large, which in turn affects the friction force.

The body edge contact domain for **IFULLEDGE** equal to 0 is illustrated by the  $\partial\bar{C}$  surface region in Fig. 3.27. The resulting contact geometry is  $C^0$ -continuous as shown in Fig. 3.28, however, contact cannot be described on concave parts of the body surface such as inside the tetrahedron in Fig. 3.28. If contact at concave surface regions is relevant, the **IFULLEDGE** parameter must be set equal to 1. This will allow for contact on the whole body edge circumferential range from 0 degree to 360 degree as illustrated by the surface  $\partial\bar{S}$  in Fig. 3.27, i.e. the body edges changes into circular cylinders. The  $C^0$ -continuity is lost for **IFULLEDGE** set to 1 because then the body edges overlap at the corner nodes. The loss of  $C^0$ -continuity occurs only at the concave surface part at the corners, and is of no importance for most problems as only pipes with very small diameters can achieve contact at these areas.

#### EXAMPLE:

---

#	elgrp	name	type	dp	angparallel	ifulledge
ELPROP	body-pipe		contbody	0.3	0.035	0

---

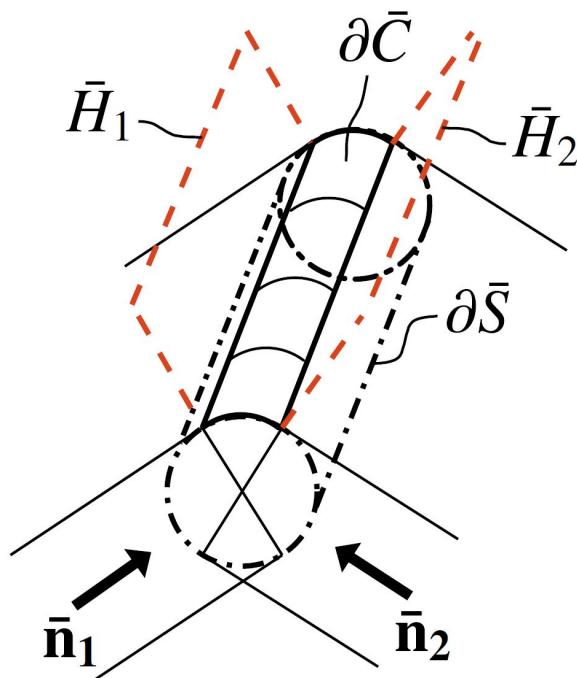


Figure 3.27:  $\partial\bar{C}$  and  $\partial\bar{S}$  illustrates **IFULLEDGE** set to 0 and 1, respectively.

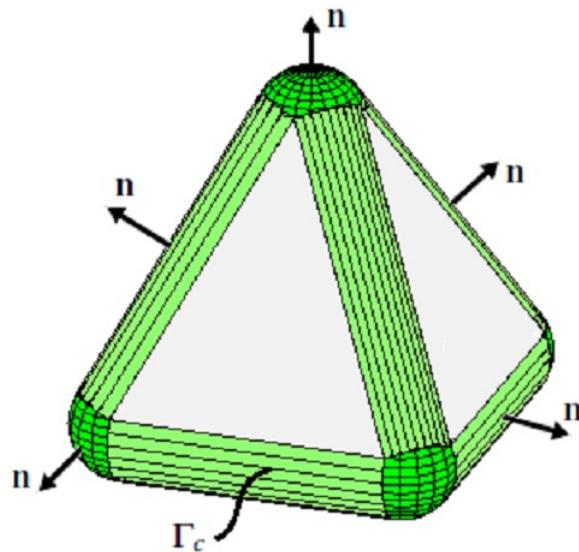


Figure 3.28:  $C^0$ -continuous contact geometry for **IFULLEDGE** set to 0.

### 3.23.5 BODY

```
ELPROP ELGRP BODY GEOM WS WD WDHX WDHY WDHZ CDX CDY
CDZ CDTHX CDTHY CDTHZ CMX CMY CMZ CMTHX CMTHY CMTHZ COGX
COGY COGZ [PHIST MHIST ICCHYD HCX HCY HCZ]
```

**GEOM:** Body geometry name, defined by the card **GEOM**.

**WS:** Submerged mass = dry mass - buoyancy mass (unit: M).

**WD:** Dry mass (unit: M).

**WDXH:** X-rotation structural mass at COG (unit: ML<sup>2</sup>).

**WDHY:** Y-rotation structural mass at COG (unit: ML<sup>2</sup>).

**WDHZ:** Z-rotation structural mass at COG (unit: ML<sup>2</sup>).

**CDX:** Drag coefficient for x-direction (unit: L<sup>2</sup>).

**CDY:** Drag coefficient for y-direction (unit: L<sup>2</sup>).

**CDZ:** Drag coefficient for z-direction (unit: L<sup>2</sup>).

**CDTHX:** Drag coefficient for x-rotation at hydrodynamic center (unit: L<sup>5</sup>).

**CDTHY:** Drag coefficient for y-rotation at hydrodynamic center (unit: L<sup>5</sup>).

**CDTHZ:** Drag coefficient for z-rotation at hydrodynamic center (unit: L<sup>5</sup>).

**CMX:** Added mass coefficient for x-direction (unit: L<sup>3</sup>).

**CMY:** Added mass coefficient for y-direction (unit: L<sup>3</sup>).

**CMZ:** Added mass coefficient for z-direction (unit: L<sup>3</sup>).

**CMTHX:** Added mass coefficient for x-rotation at hydrodynamic center (unit: L<sup>5</sup>).

**CMTHY:** Added mass coefficient for y-rotation at hydrodynamic center (unit: L<sup>5</sup>).

**CMTHZ:** Added mass coefficient for z-rotation at hydrodynamic center (unit: L<sup>5</sup>).

- COGX**: Local x-eccentricity from element origin to COG (unit: L).
- COGY**: Local x-eccentricity from element origin to COG (unit: L).
- COGZ**: Local x-eccentricity from element origin to COG (unit: L).
- PHIST**: External pressure and buoyancy mass history ID number, optional.
- MHIST**: Dry mass history ID number, optional.
- ICCHYD**: Hydrodynamic Coriolis-centripetal load switch, optional, default value: 0.
- 0 : No hydrodynamic Coriolis-centripetal loads are included.  
 1 : Hydrodynamic Coriolis-centripetal loads are included.
- HGX**: Local x-eccentricity from element origin to hydrodynamic center, optional, default value: **COGX**(unit: L).
- HGY**: Local y-eccentricity from element origin to hydrodynamic center, optional, default value: **COGY**(unit: L).
- HGZ**: Local z-eccentricity from element origin to hydrodynamic center, optional, default value: **COGZ**(unit: L).

The center of buoyancy coincides with the COG location.

With reference to **ICCHYD=1**, the hydrodynamic Coriolis-centripetal loads consist of added mass forces and moments in DOFs 1-6 due to body angular velocity and Munk moment loads in DOFs 4-6 due to linear velocity. The former load type may be important for large angular velocities combined with large hydrodynamic mass variation between the DOFs, such as for a trawl board colliding with a pipeline. The Munk moments are set to zero for DOFs that also are loaded by drag and lift moments from the **HYDROPRO** trawl board model in Section 3.33.1. This is because the drag and lift moment coefficients are assumed to be based on laboratory tests that include the Munk moments.

#### EXAMPLE:

---

```
#      name type  geom   ws    wd    wdthx  wdthy  wdthz  cdx   cdy   cdz   cdthx  cdthy  cdthz
ELPROP sdd  body  geo1  2.0  2.1  0.0   0.0    0.0    1.5   1.5   1.5   0.0   0.0   0.0
# cmx  cmy  cmz  cmthx  cmthy  cmthz  cogx  cogy  cogz  phist  mhist  icchyd  hcx  hcy  hcw
  0.15  0.15  0.15  0.0   0.0   0.0   0.0   0.0   0.0   100   110    1   0.1  0.0  1.0
```

---

### 3.23.6 CABLE

**ELPROP ELGRP CABLE CDR CDT CMR CMT MD MS ODP ODW RKS [PHIST  
MHIST]**

**CDR**: Radial drag coefficient (unit: -).

**CDT**: Tangential drag coefficient (unit: -).

**CMR:** Radial mass coefficient (Normally 2.0 for circular pipes. This will result in an added mass of  $(RMADD - 1.0)\rho_w \frac{\pi}{4} D^2$ ) (unit: -).

**CMT:** Tangential mass coefficient (Must be  $\geq 1.0$ . A value of 1.0 will result in no tangential added mass according to the above formula.) (unit: -).

**MD:** Dry mass (unit:  $ML^{-1}$ ).

**MS:** Dry mass - buoyancy mass (unit:  $ML^{-1}$ ).

**ODP:** Outer diameter  $D_{op}$  (unit: L)

**ODW:** External wrapping outer diameter  $D_{ow}$  (unit: L)

**RKS:** External wrapping fraction  $\eta$  (Range: 0-1). Hence the hydrodynamic diameter that will be applied to calculate drag and mass forces will be:  $D = (1 - \eta)D_{op} + \eta D_{ow}$

**PHIST:** External pressure and buoyancy mass history ID number, optional.

**MHIST:** Dry mass history ID number.

The outer contact diameter  $OD_c$  is taken as the maximum of the outer diameter and the wrap diameter according to  $OD_c = \max(\text{ODP}, \text{ODW})$ .

EXAMPLE: \_\_\_\_\_

```
#  
#      elgrpname elproptype Cdr Cdt CMr CMt wdry   wsub   ODpipe ODwrap extwrapfrac  
ELPROP wire      cable       0.8 0   2.0   1.0   0.0395 0.01   0.1    0.1    0
```

---

### 3.23.7 USERCONTACT

**ELPROP ELGRP USERCONTACT LOCGLOB MATERIAL**

**LOCGLOB:** Local-global parameter (dummy);

**GLOBAL :** Global system

**LOCAL :** Element system

**MLINEID:** ID of material line

The material line **MLINEID** describes the material properties on KP basis as defined by the **COSUPR** command in Section 3.7. The whole KP-range of the material line must be assigned the material type **SPRING** in Section 3.37.19. The material line ID does not need to be included in the **COSURFPR** command, however, the KP-coordinate will be calculated based on the centre line and **KP0** defined by the **COSURFPR** command in Section 3.8.

The parameter **LOCGLOB** is currently a dummy, meaning that the input spring characteristics, force and displacement components are referred to the element system.

EXAMPLE: \_\_\_\_\_

```
#      name    type        system IDline
ELPROP burial usercontact LOCAL   100
```

---

### 3.23.8 COAT

**ELPROP ELGRP COAT SHEARMAT WIDTH TH WD WS**

**SHEARMAT**: Coating shear interaction material name.

**WIDTH**: Coating segment width (unit: L).

**TH**: Coating thickness (unit: L).

**WD**: Dry mass (unit:  $ML^{-1}$ ).

**WS**: Submerged mass = dry mass - buoyancy mass (unit:  $ML^{-1}$ ).

EXAMPLE: \_\_\_\_\_

```
# Element property input:
#-----
#      name    type shearmat width th      wd      ws
ELPROP concrete coat intermat 0.10  0.035  0.5e-3  0.2e-3
#
```

---

### 3.23.9 ROLLER

**ELPROP ELGRP ROLLER RD [CONTPAR1=VALUE CONTPAR2=VALUE]**

**RD**: Roller diameter (unit: L).

**CONTPAR1=VALUE**: **CONTPAR1=** is a character string and **VALUE** is an integer equal to either **0** or **1**. For **VALUE=0** which is default, an arbitrary number of contact elements may obtain contact with the same pipe element. If **VALUE=1**, only one contact element can be active for each pipe element.

**CONTPAR2=VALUE**: **CONTPAR2=** is a character string and **VALUE** is an integer that defines the number of pipe elements a contact point is allowed to move along during one increment or iteration step before the contact is deactivated. For the example in Fig. 3.29, **VALUE=2** means that contact search will be performed for pipe element  $i$  which had active contact at the previous load step and for its 4 neighbouring elements  $i-2$ ,  $i-1$ ,  $i+1$  and  $i+2$ . If contact is not present at any of the 5 pipe elements at the converged equilibrium state, e.g. if the contact point has moved to elements  $i-3$  or  $i+3$ , the contact status will be set to inactive and

the friction variables will be reset. By default **VALUE=1** is applied. The **CONTPAR2=VALUE** option is currently only available for CONT164 in combination with the **ISOKXYCONTACT** material type in Section 3.37.10.

The **CONTPAR1=VALUE** option is applied in situations where two or more contact elements may detect the exact same pipe contact point. A typical example is shown in Fig. 3.30, where nine CONT164 elements with identical rollers are applied to handle the pipe contact. If **CONTPAR1=0** is applied, all of the contact elements will detect the same contact point, i.e. the one with largest interpenetration, resulting in too concentrated contact forces and too large penetrations at the locations where contact is falsely not detected. By instead using **CONTPAR1=1**, the contact element that first detects a contact point, will tag the pipe element with a contact indicator so that the remaining contact elements cannot be activated for the given pipe element.

Note that when **CONTPAR1=1**, the contact element group still detects contact with a pipe element that interacts simultaneously with a contact element group which has been assigned **CONTPAR1=0**. Likewise, a contact element group with **CONTPAR1=0** will detect contact at a pipe element that simultaneously interacts with a contact element group which has been assigned **CONTPAR1=1**. Hence, if a pipe element shall be fully restricted to only have one roller contact point, **CONTPAR1=1** must be assigned to all contact element groups that possibly can come into contact with the pipe element. Note also that the **CONTPAR1=VALUE** functionality only affects roller-based contact elements. Hence, a CONT164 element with **CONTPAR1=1** will detect contact at a pipe element that simultaneously interacts with a CONT152 element.

The **CONTPAR2=VALUE** option is used for avoiding that unrealistic tangential pipe contact point displacements occur during a load step. For the analysis in Fig. 3.30, a single contact element may detect contact points that differs with an angular coordinate of up to 90 degrees between the load steps, resulting in completely wrong tangential displacement increments for updating the friction force. By selecting e.g. **CONTPAR2=1**, the issue is avoided as the contact will then be deactivated if the pipe contact point detected by the contact element moves more than 1 pipe element during the load step. However, in that case it must be assured that there are enough contact elements available to detect the occurrence of new contact points, otherwise contact will falsely not be detected.

The input data for the example in Fig. 3.30 is as follows:

EXAMPLE:

---

```
#      name    type     rd
ELPROP  roll1   roller  0.5   contpar1=1   contpar2=1
#
```

```

#
#      name    eptyp   matname   elid    nod1           n  nelinc nodinc
ELCON  roll1  cont164 rollmat  40001   5301    repeat  9    1    0
#      type    elid  elend  xecc  yecc  zecc  yphi  dx1  dy1  dz1  dx2  dy2  dz2
ELECC stinger 40001   1    0.0   0.0   0.0   0.0   0.0  1.0  0.0   0.0  2.0  0.0
#      n  elinc  ds  dyphi
repeat 9    1    0    0.0

```

---

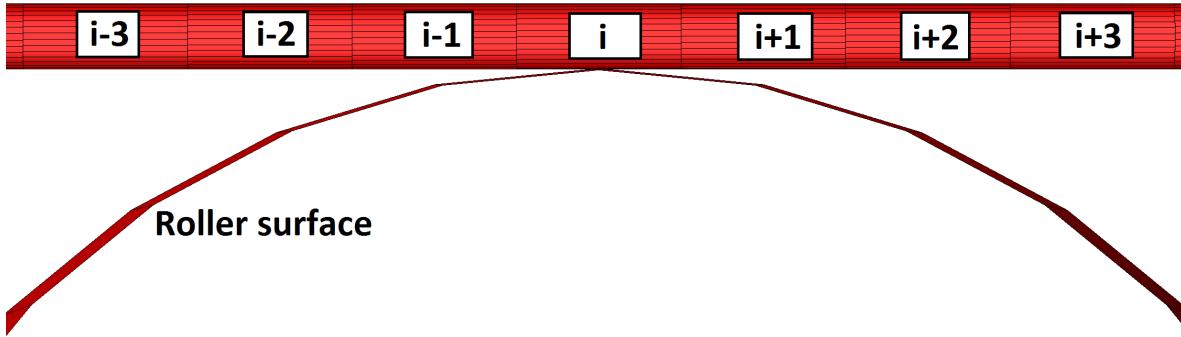


Figure 3.29: Defining maximum allowable incremental pipe contact point displacement by **CONTPAR2=VALUE**

### 3.23.10 GENSPRING

**ELPROP ELGRP GENSPRING TX TY TZ RX RY RZ [IREF ICOULCNTR]**

**TX**: Time step at which the x-spring is activated.

**TY**: Time step at which the y-spring is activated.

**TZ**: Time step at which the z-spring is activated.

**RX**: Time step at which the x-rotation spring is activated.

**RY**: Time step at which the y-rotation spring is activated.

**RZ**: Time step at which the z-rotation spring is activated.

**IREF**: Integer parameter for controlling the reference coordinate system applied for the material curves, the element displacements and the element forces. Further description of allowable values are given below.

**ICOULCNTR**: Integer parameter for controlling the **COULOMB** force scaling option specified by the **GENSPRING** material type in Section 3.37.20. Further description of allowable values are given below.

For **IREF ≠ 1**, the material curves, the element displacements and the element forces refer to the element coordinate system. This is the default behavior when the optional parameter **IREF** is omitted. During the analysis, the element coordinate system will

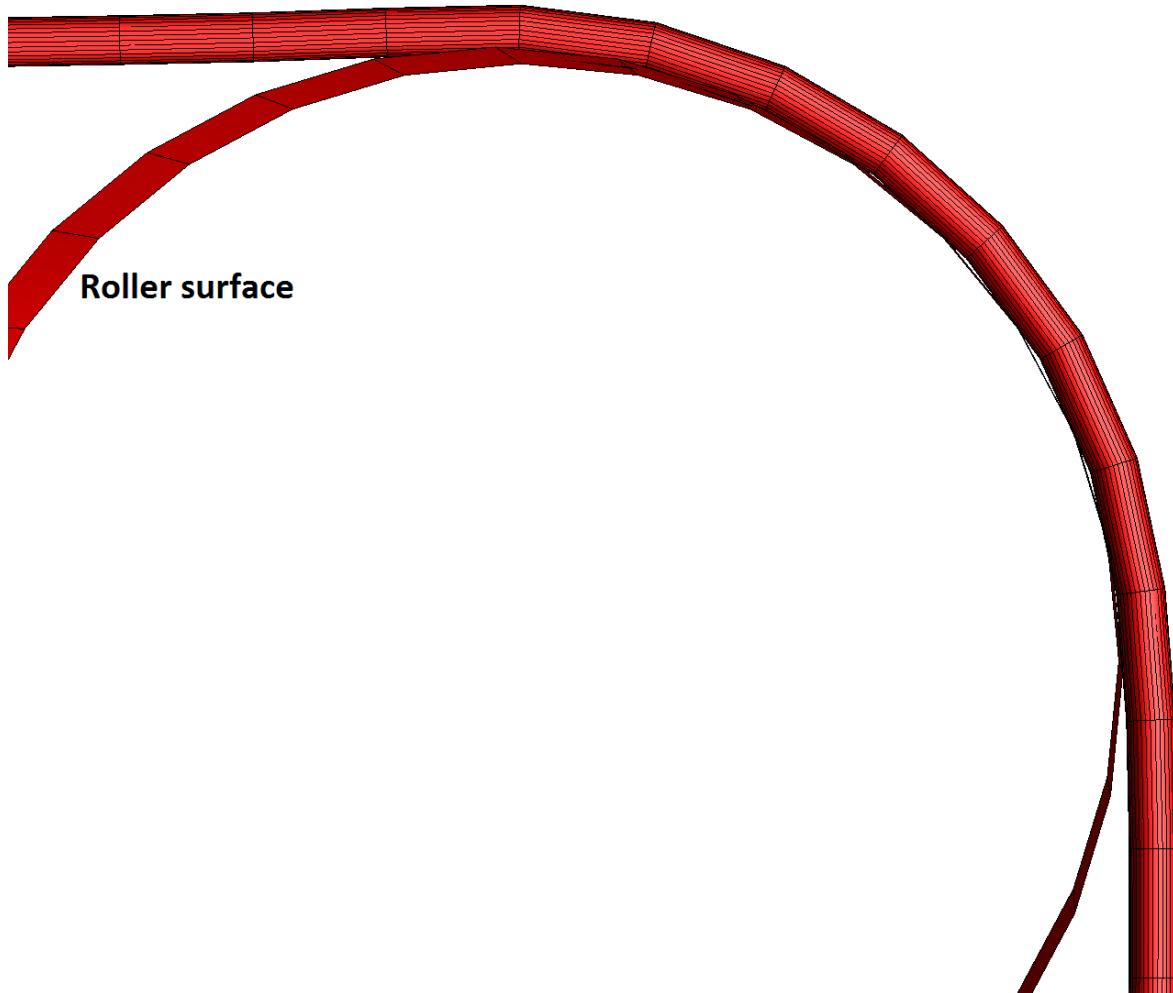


Figure 3.30: Analysis where **CONTPAR1=1** and **CONTPAR2=0** or **1** is recommended

follow the rotations of element node number 1, see **NOD1** for the SPRING element defined by the **ELCON** card in Section 3.16.

For **IREF=1**, the material curves, the element displacements and the element forces refer to the global coordinate system. The only exception from this is when a special reference system is used for the nodal motion of element node 1, see **NOD1** for the SPRING element defined by the **ELCON** card in Section 3.16. A special nodal motion reference system will be applied for **NOD1** in the following cases:

- When the node is assigned the boundary condition types **LOCAL** or **SPECIAL**, see the **BONCON** card in Section 3.2. In that case, the material curves, the element displacements and the element forces refer to the local or special coordinate system of the node, which rotates together with the node during its motion.
- When the node is a helix node. This is the case if the node also is defined as

element node 3 or 4 for the HSHEAR342 element type, see **NOD3** and **NOD4** defined by the **ELCON** card in Section 3.16. Then the material curves, the element displacements and the element forces refer to the local coordinate system of the helix node.

For **ICOULCNTR** $\neq$ **1**, the  $x$ -direction force  $F_x$  is calculated by performing **COULOMB** scaling of the force  $f_x$  from the  $x$ -direction material curve according to  $F_x = f_x \sqrt{F_y^2 + F_z^2}$ , where  $F_y$  and  $F_z$  are the spring forces in the  $y$ - and  $z$ -directions, respectively. The moment about the  $x$ -axis is calculated as  $M_x = m_x \sqrt{F_y^2 + F_z^2}$ , where  $m_x$  is taken from the  $x$ -rotation material curve. Note that these scaling conventions means that the friction coefficient  $\mu$  must be built into the dimensionless force  $f_x$  specified for the  $x$ -direction material curve, and for the  $x$ -rotation material curve both the friction coefficient  $\mu$  and the distance to the modelled contact point  $r$  must be incorporated into the moment quantity  $m_x$  which has unit  $L$ . The **COULOMB** force scaling option is not applied in the  $y$ -direction for **ICOULCNTR** $\neq$ **1**, i.e. the  $y$ -direction force  $F_y$  is set equal to  $f_y$  specified for the  $y$ -direction material curve. By default the option **ICOULCNTR** $\neq$ **1** is applied when the parameter **ICOULCNTR** is omitted.

By selecting **ICOULCNTR** $=$ **1**, a 2D isotropic friction model is applied by performing **COULOMB** scaling of  $f_x$  from the  $x$ -direction material curve. Here, the force  $f_y$  specified for the  $y$ -direction material curve is dummy, and the 2D friction force in the  $xy$ -plane is given by  $\mathbf{F}_t = f_x |F_z| \mathbf{t}$  where  $\mathbf{t}$  is a 2D unit tangent vector in the sliding direction. Note that this scaling convention means that the friction coefficient  $\mu$  must be built into the dimensionless force  $f_x$  specified for the  $x$ -direction material curve. The **COULOMB** scaling option is not applied for rotation about the  $x$ -axis for **ICOULCNTR** $=$ **1**, i.e. the moment  $M_x$  about the  $x$ -axis is set equal to the moment  $m_x$  specified for the  $x$ -rotation material curve.

The parameter **ICOULCNTR** is dummy when the default **USERDEFINED** force scaling option in Section 3.37.20 is applied.

Note that the **COULOMB** scaled material curves should be defined by the **EPCURVE** material curve option in Section 3.37.3, otherwise reversal of the friction force will not be correctly modelled. A typical application for **ICOULCNTR** $\neq$ **1** is modelling of axial friction for pipe-in-pipe problems where the force  $\sqrt{F_y^2 + F_z^2}$  represents the radial force.

#### EXAMPLE:

---

```
#      name      type      tstpx  tstpy  tstpz  tstprx  tstpry  tstprz
ELPROP vessel1    genspring     1.0     1.0     1.0     1.0     1.0     1.0
#
# iref  icoulcntr
0      1
```

### 3.23.11 SOILCONTACT

```
ELPROP ELGRP SOILCONTACT TX TY TZ TTX THISTX THISTY THISTZ  
THISTTX XNAME YNAME ZNAME TXNAME [T0_UZ THISTUZ UZNAME]
```

- TX:** Activation time for spring in pipe axial direction.
- TY:** Activation time for spring in pipe transverse direction.
- TZ:** Activation time for spring in pipe-seabed normal direction.
- TTX:** Activation time for spring about pipe axial rotation DOF.
- THISTX:** Time history for scaling of spring force and stiffness in pipe axial direction.
- THISTY:** Time history for scaling of spring force and stiffness in pipe transverse direction.
- THISTZ:** Time history for scaling of spring force and stiffness in pipe-seabed normal direction.
- THISTTX:** Time history for scaling of spring moment and stiffness about pipe axial rotation DOF.
- XNAME:** Material name for spring in pipe axial direction. Force unit: F/L. Displacement unit: L.
- YNAME:** Material name for spring in pipe transverse direction. Force unit: F/L. Displacement unit: L.
- ZNAME:** Material name for spring in pipe-seabed normal direction. Force unit: F/L. Displacement unit: L.
- TXNAME:** Material name for spring about pipe axial rotation DOF. Moment unit: F. Displacement unit: rad.
- T0\_UZ:** Reference time for the KP-based initial soil embedment. Optional.
- THISTUZ:** Time history for scaling of KP-based initial soil embedment. Optional.
- UZNAME:** Table name for KP-based initial soil embedment. Optional.

The main purpose of the time-controlled soil springs is to enhance the modelling capabilities of the analysis start-up phase. A typical application is to avoid singular stiffness matrix in static analysis due to e.g. lack of seabed contact or lack of axial and transverse contact element stiffness. All material curves must be defined by the **HYCURVE** material type in Section 3.37.4. The springs are active regardless of whether contact is present or not. The springs are deactivated by using zero load factor in the time history definitions. Spring forces and moments are added to the ordinary soil contact forces and moments, and will be included in results reported by SIMPOST and for results defined by the **DYNRES\_E** card. The element property type is optional. No time-controlled springs will be present if the **ELPROP** input card for **SOILCONTACT** is omitted.

KP-based initial soil embedment is modeled by the 2-dimensional table **UZNAME**. The first table column contains the KP-values with coordinate reference as defined by the **COSURFPR** card in Section 3.8. The second column of the table contains the initial soil embedment  $\bar{u}_{z0}$ . See Section 3.49 for definition of the table input format.

The reference time **T0\_UZ** refers to the simulation time where the pipeline shall have soil penetration equal to the specified KP-based initial soil embedment  $\bar{u}_{z0}$ . To achieve this, SIMLA adjusts for the contact element elastic penetration  $u_z$  as follows,

$$u_{z0} = \begin{cases} f_1 \cdot \bar{u}_{z0} - f_2 \cdot u_z(t_0) & \text{if } u_z(t_0) \leq 0.0 , \quad t > t_0 \quad (\text{contact at } t=t_0) \\ f_1 \cdot \bar{u}_{z0} & \text{if } u_z(t_0) \geq 0.0 , \quad t > t_0 \quad (\text{no contact at } t=t_0) \\ f_1 \cdot \bar{u}_{z0} - f_2 \cdot u_z(t) & \text{if } u_z(t) \leq 0.0 , \quad t \leq t_0 \end{cases} \quad (3.19)$$

where  $t_0=\text{T0\_UZ}$ , the load factor  $f_1$  is taken from the **THISTUZ** load history,  $f_2$  is set equal to  $f_1$  but limited by  $f_2 \leq 1.0$ ,  $u_{z0}$  is the force-free initial displacement that will be applied for the contact element,  $\bar{u}_{z0}$  is the specified KP-based initial soil embedment at simulation time  $t_0=\text{T0\_UZ}$ , and  $u_z$  is the contact element elastic penetration used for computing the normal contact force  $F_z$ .

The sign convention  $u_z < 0.0$  means that the pipeline penetrates a force-producing distance into the seabed, and  $u_{z0} < 0.0$  means that the pipeline sinks a force-free distance into the seabed. The quantity  $u_{z0}$  represents the irreversible modification of the virgin seabed, which when added together with the contact element elastic penetration  $u_z$  gives the total soil embedment  $\bar{u}_{z0}$  at simulation time  $t_0=\text{T0\_UZ}$ .

The KP-based initial soil embedment is included in the soil penetration used for calculating the transverse resistance for the **CLAY\_Y** and **SAND\_Y** models in Sections 3.37.12 and 3.37.13.

---

**EXAMPLE:**


---

```
#      elgrp     type          tx    ty    tz    trx   thistx  thisty  thistz  thistrx
ELPROP  seabed  soilcontact  0.0  0.0  0.0  0.0   400     400     400     400
#
# xname     yname     zname     txname     t0_uz   thistuz  uzname
soilxh    soilyh    soilzh    soiltxh   1.0      450      uzini
#
#       name     ncol
TABLE  uzini  2
-100.0  -0.2
 100.0   -0.1
 400.0   -0.2
```

---

### 3.24 ENVRES\_ - Envelope results

By the **ENVRES** cards, user select results for a given range of element or nodal nodes will be stored on the .raf file on a format that enable envelope results to be presented on ASCII matrix plot format by the SIMPOST postprocessor. The following format is applied:

```
ENVRES_N, E OR I ...
```

where **E** means element results, **N** means nodal results and **I** means Gaussian integration station results. The allowable options are defined below:

#### 3.24.1 ENVRES\_N

For **ENVRES** with option **\_N** (nodal results) the following format applies:

```
ENVRES_N TYPE NODE1 NODE2 DOF STAT [START STOP]
```

**TYPE**: Type or result, where allowable values are:

- 1** : nodal displacement
- 2** : nodal velocity
- 3** : nodal acceleration
- 4** : relative displacement

**NODE1**: First node ID

**NODE2**: Last node ID

**DOF**: Degree of freedom number;

- 1** : x-direction
- 2** : y-direction
- 3** : z-direction
- 4** : rotation x-direction
- 5** : rotation y-direction
- 6** : rotation z-direction

**TIME0**: Time to store static information.

**START**: Start time for storing results, optional.

**STOP**: End time for storing results, optional.

#### 3.24.2 ENVRES\_E

For elements of type PIPE, CONT and SPRING, the **ENVRES** card can be used with the **\_E** option (element results). The following format applies:

**ENVRES\_E TYPE EL1 EL2 ELNOD DOF TIME0 [START STOP]**

**TYPE:** Type of result, where allowable values are:

- 1 :** = displacement
- 2 :** = forces and moments
- 3 :** = torsion and curvatures
- 4 :** = absolute curvature

**EL1:** Element 1 ID.

**EL2:** Element 2 ID. Results are stored for element **EL1** to **EL2**.

**ELNODE:** Element node, i.e. element end. (1 or 2 for PIPE and SPRING, only 1 for CONT).

**DOF:** Degree of freedom number. The meaning is element type dependent:

**PIPE31-39, SPRING136-137 :** :

- 1 :** = x-disp/x-force /x-torsion
- 2 :** = y-disp/y-force/y-curvature
- 3 :** = z-disp/z-force/z-curvature
- 4 :** = x-rotation/x-moment
- 5 :** = y-rotation/y-moment
- 6 :** = z-rotation/z-moment

**CONT :** contact element types:

- 1 :** = x-disp, x-force
- 2 :** = y-disp, y-force
- 3 :** = z-disp, z-force

**TIME0:** Time to store static information.

**START:** Start time for storing results, optional.

**STOP:** End time for storing results, optional.

### 3.24.3 ENVRES\_I

For elements of type PIPE33 and PIPE39, the **ENVRES** card can be used with the **\_I** option (Gaussian interpolation station results). The following format applies:

**ENVRES\_I TYPE EL1 EL2 IGAU IPOINT TIME0 [START STOP]**

**TYPE:** Type of result, where allowable values are:

- 1 :** = sigma-xx
- 2 :** = strain-xx

**EL1**: Element 1 ID.

**EL2**: Element 2 ID. Results are stored for element **EL1** to **EL2**.

**IGAU**: Element integration station 1-3, 1 and 3 at element ends.

**IPOINT**: Integration point number, max NPOINT see Section 3.6.

**TIME0**: Time to store static information.

**START**: Start time for storing results, optional.

**STOP**: End time for storing results, optional.

EXAMPLE:

---

```
#  
#          type  nod1  nod2  dof  statstep  
ENVRES_N   1      1      441    1    1  
ENVRES_N   1      1      441    3    1  
#  
#          type  el1    el2    end   dof  statstep  
ENVRES_E   2      2301  2316   1    3    1  
ENVRES_E   2      1001  1200   1    3    1  
ENVRES_E   2          1  440    2    1    1  
ENVRES_E   2          1  440    2    5    1  
ENVRES_E   3          1  440    1    2    1  
#  
#          strain  el1  el2  gauss  point  statstep  
ENVRES_I   2          341  440   1      4      1
```

---

### 3.25 FATPROP - FATigue properties

The **FATPROP** card enables the user to associate fatigue properties stored on specific files names to the different material names

The format is as follows:

<b>FATPROP NAME FILE</b>
--------------------------

where

**NAME**: Material name.

**FILE**: File name at which the fatigue data are stored.

#### 3.25.1 Fatigue file format

One line on the format:

<b>NFDPO R1 IGERB INTCO SCF SIGUTS</b>
--

**NFDPO** : The number of points in the fatigue S-N diagram

**R** : The R-ratio defined as  $\sigma_{min}/\sigma_{max}$  for the S-N-diagram

**IGERB** : Method for taking the mean stress into account

**0** : No mean stress is taken into account

**1** : Goodman interpolation mean stress calculated as

$$\sigma_{xx} + \sigma_{yy} + \sigma_{zz}$$

**2** : Gerber interpolation mean stress calculated as

$$\sigma_{xx} + \sigma_{yy} + \sigma_{zz}$$

**3** : Goodman interpolation mean stress calculated as

$$\bar{\sigma} = \sqrt{\sigma_{xx}^2 + \sigma_{yy}^2 + \sigma_{zz}^2 - \sigma_{xx}\sigma_{yy} - \sigma_{zz}\sigma_{yy} - \sigma_{xx}\sigma_{zz} + 3\sigma_{xy}^2 + 3\sigma_{zy}^2 + 3\sigma_{xz}^2}$$

**4** : Gerber interpolation mean stress calculated as

$$\bar{\sigma} = \sqrt{\sigma_{xx}^2 + \sigma_{yy}^2 + \sigma_{zz}^2 - \sigma_{xx}\sigma_{yy} - \sigma_{zz}\sigma_{yy} - \sigma_{xx}\sigma_{zz} + 3\sigma_{xy}^2 + 3\sigma_{zy}^2 + 3\sigma_{xz}^2}$$

**INTCO** : Method for interpolation in S-N diagram:

**1** : both stress and N in log scale

**2** : stress in linear scale, N in log scale

**SCF** : Stress concentration factor (unit: )

**SIGUTS** : Ultimate stress for mean stress correction (unit: F/L<sup>2</sup>)

Then a number of NFDPO lines on the format:

**POINT SRANGE NCYFAL**

**POINT** : The point number =1- .... Number of points.

**SRANGE** : Stress range (in increasing order), starting with the minimum threshold value (unit: F/L<sup>2</sup>)

**N** : Corresponding number of cycles to failure.

*Note:* Only numeric values in the fatigue data file. No lines starting with # is possible in the fatigue data file, i.e. just numbers

### 3.26 FEED - lay simulation scenario definition (feed elements from lay vessel)

When the **FEED**-command is used, a group of uncoupled elements, all of them with end 1 on the exact same location, must be defined with the **ELCON** and **REPEAT** commands. The stack must be located at a storage node on the vessel.

In addition a guide element must be defined, and be coupled to the vessel in the upper end by **CONSTR** cards. The coupling must be global. If constraints are defined for all the dofs, the local x-axis of this element defines the departure angle relative to the lay

vessel during feed. If only the three dofs of translation is constrained, the top of the feed guide will act as a hinge, and direction of the guide element is continuously changed.

The feeding and connectivity of the pipe is defined by constraint coefficients. A **CON-STR**-command (with **FEEDCONEQ**-option) must be given for end 1 of all the elements in the group. This will cause all the elements in the group to be coupled to the storage node before feeding, to end 2 of the guide-element during feeding and to the subsequent element in the pipe group after the element is fed.

For a standard lay operation or a laydown/retrieval case the **TIMEINIT** option in the **CONTROL** card should be used. In this case the storage node and the feed node should be the same. The guide element will represent the last tensioner on the vessel and should be fully constrained to the connecting node on the vessel. The element model is set up in the same way as for the **AUTOSTART** option, but the pipe model should consist of guide element, feed group and pipe group from static screening. The static configuration of the pipe group will be read from file. The elements in feed group and the guide element will be moved to the top of the static pipe configuration. The connecting node on the vessel will be moved to the same x and y-coordinates as the upper end of the guide element, see Fig. 3.31.

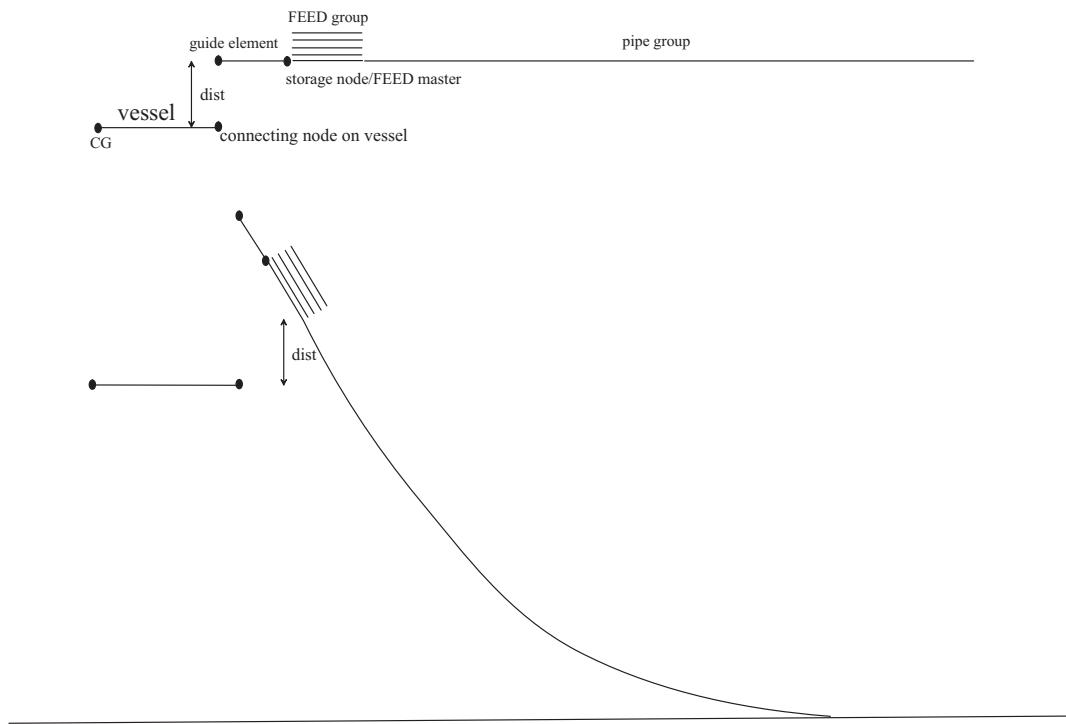


Figure 3.31: Standard FEED analysis.

A special start routine is available to configure the system for feeding during sdd-lay, see the **CONTROL** card, Section 3.6.

During feed two options for control of the feeding is available:

- Feed rate given in **SIMLA** card
- Specified distance from lowest node of pipe to seabed

The **FEED** command has the following format:

**FEED GRPNAME TIME IOP ...**

where:

**GRPNAME**: Name of the element group to be used in **FEED**.

**TIME**: Start time for feeding.

**IOP**: Type of feed approach. **IOP** can take the following values:

**1** : feed of pipe at a specified rate

**3** : distance from sea floor is specified

For option **1** feed rate must be given in the **SIMLA** card, see Section 3.48. Two optional parameters can be given to specify tensioner rotation. If not given no rotation in tensioner is assumed. If given, the rotation in tensioner [rad/m] is given by

$$r_x x = \text{sign}(T) c0 + c1 T$$

, where  $T$  is the torque moment in the tensioner.

**FEED GRPNAME TIME IOP [C0 C1]**

**C0**: Constant rotation rate (optional) .

**C1**: Torque dependent rate (optional).

For option **3** the format is as follows:

**FEED GRPNAME TIME IOP DIST THIST**

**DIST**: Distance from sea floor.

**HIST**: Corresponding time history (dummy).

EXAMPLE: \_\_\_\_\_

```
#      Feed control
#      grpname      tstart iop   c0      c1
FEED vesselwire  100.0  1    1.0e.3 1.0e-8
```

### 3.27 FLOWLOAD - Slug flow load model

The load model is only available for the PIPE31, PIPE33 and COMPIPE42 element types.

The slug flow load model consists of a consecutive number of fluid segments each described by its length  $L_i$  and fluid density  $\rho_i$ , which moves with velocity  $U$  along the pipe with internal cross-sectional area  $A_i$ . It includes the following four pipeline load contributions:

- The fluid gravity giving a vertical load per length of flowing mass:  $\rho_i g A_i$
- The static internal pressure  $p_i$  giving an effective compressive axial force:  $-p_i A_i$
- The centripetal effect providing an effective compressive axial force:  $-\rho_i A_i U^2$
- The Coriolis effect giving an external load per length:  $-2\rho_i U \frac{\partial \dot{x}}{\partial s}$

where  $s$  is the pipe longitudinal curvilinear coordinate and  $\frac{\partial \dot{x}}{\partial s}$  is the pipe velocity gradient. The fluid weight  $\rho_i g A_i$  comes in addition to the submerged mass and the associated dry mass specified by the **ELPROP** input card for the pipe in Section 3.23.1. Hence, the fluid must not be included in the pipe properties. Note that inertia loads due to change of flow velocity  $U$  is not accounted for.

The slug flow load model is specified by the following format:

```
FLOWLOAD UHIST IELREF UFLOW PREF PSWITCH CPSWITCH CORSWITCH
FLOWFILE ELGR1 [ ELGRP2 ... ELGRN ]
```

where

**UHIST**: Load history number for the flow velocity.

**IELREF**: The origin of the curvilinear coordinate  $s$  is placed at end 2 of element **IELREF**, and is also the location where the reference static internal pressure **PREF** is prescribed.

**UFLOW**: Flow velocity which is scaled by **UHIST** [ $LT^{-1}$ ].

**PREF**: Reference static internal pressure at end 2 of **IELREF**, which is scaled by the load history specified in the **PILOAD** input card [ $FL^{-2}$ ].

**PSWITCH**: On-off switch for fluid gravity and static internal pressure (1=on 0=off). The static internal pressure from the **PILOAD** input card will be applied for **PSWITCH=0**, while it will be overruled by the static internal pressure from the **FLOWLOAD** input card for **PSWITCH=1**.

**CPSWITCH**: On-off switch for the centripetal effect (1=on 0=off).

**CORSWITCH**: On-off switch for the Coriolis effect (1=on 0=off).

**FLOWFILE:** The text file name where the fluid segment data is defined. The file consists of consecutive lines specifying density and length of the fluid segments.

**ELGRI:** The pipe element group name(s) where the slug flow is acting. If the slug flow goes through several pipe element groups, their **ELCON** input cards must be given in consecutive order as defined by the positive flow direction.

The **FLOWFILE** contains consecutive lengths of fluid segments where the downstream end is located at curvilinear coordinate  $s = 0$  at zero time. The origin  $s = 0$  is located at end 2 of , which means that the upstream end of the fluid segments has a negative  $s$ -value at zero time equal to the total length of the slug flow specified in the **FLOWFILE**.

The slug flow moves with velocity  $U$  where the incremental longitudinal motion  $\Delta s$  of all fluid segments is computed by  $\Delta s = U\Delta t$  at every load step.

The static internal pressure  $p_i$  is calculated based on the reference pressure **PREF** at  $s = 0$  and by considering the fluid segment densities  $\rho_i$  and the vertical height difference as illustrated in Fig. 3.32. The fluid weight inside a pipe element is computed based on the densities  $\rho_i$  of the fluid segments that instantaneously occupies the element.

*Note:* **PREF** is prescribed at , and consequently negative static internal pressures may appear elsewhere in the pipeline if **FLOWFILE**,  and **PREF** are not defined with caution. The fluid segment densities  $\rho_i$  are averaged between adjacent elements, and therefore the simulated static internal pressure may deviate slightly from the true static internal pressure.

*Note:* The **PILOAD** input card in Section 3.43 must be given in order to activate the static internal pressure  $p_i$  since it is scaled by the load history specified in the **PILOAD** input card.

*Note:*  $\rho_i = 0$  is applied outside of the fluid segment range defined in the **FLOWFILE**.

EXAMPLE: \_\_\_\_\_

```
#          uhist    ielref   uref    pref     pswitch   cpswitch   corswitch   flowfile   elgr1
FLOWLOAD  200      1001     1.0    10.0      1           1           1           flow.txt   pipe1
```

---

The file 'flow.txt' contains the fluid densities  $\rho_i [ML^{-3}]$  in the second column and the slug flow segment lengths  $L_i [L]$  in the first column:

EXAMPLE: \_\_\_\_\_

```
2.0  40
3.0  60
4.0  80
```

---

The above example of the file 'flow.txt' is illustrated in Fig. 3.32 for a model consisting of 6 pipe elements. End 2 of element number **IELREF**=2 is the location where the static reference pressure **PREF** is prescribed. The origin of the curvilinear coordinate  $s$ , which is fixed throughout the analysis, is located at end 2 of **IELREF**. At beginning of the analysis, the downstream end of the third fluid segment with length  $L_3=3.0$  is located at end 2 of element **IELREF** where  $s = 0$ . The total length of the slug flow is 9.0. The upstream end of the slug flow is therefore located at coordinate  $s=-9.0$  at start of the analysis. The slug flow moves in the positive  $s$ -direction during the analysis with the reference velocity **UREF** multiplied by its load history **UHIST**.

### 3.28 GEOM - geometry of body

The **GEOM** card can be applied to specify a geometry for an element of type BODY. The geometry will define the contact surface between a pipe and the body if contact elements are defined between them. Note that only CONT152 and CONT153 can be applied for this sort of contact.

A geometry is defined by a sequence of segments, where each segment has its own geometric shape. There are two different concepts of segments:

- Line segments so far including **DTRUMPET**, **BELLMOUTH** and **CYLINDER**. These can be combined arbitrarily along the same line. The body element node will be in the centre of the first segment.
- **GENERAL** segments are 2D plates that are arbitrarily positioned and oriented relative to the body element system. The shape is defined by the **CROSSGEOM**, Section 3.9. The geometry is meshed by a tringularization procedure. The position relative to the node is user defined. Note that line segments and general segments cannot be combined.
- **FROM\_FILE** 3D geometry description that are stored on a file arbitrarily positioned and oriented relative to the body element system.

The orientation of the geometry is the same as the element it is connected to. The element node will be in center of the first segment, unless eccentricities are given by the **ELECC** card, see Section 3.18. Note that the same eccentricities must be defined for the contact elements as well.

The format of the card is as follows:

```
GEOM NAME NSEG TYPE THETA NVIS L ....
```

.... ....

where

**NAME**: Name of geometry. This will be used as reference in the **ELPROP** card, see Section 3.23.

**NSEG**: Number of segments in the geometry. A geometry can be built from several segment types aligned after each other in the sequence they are given in the geometry card.

**TYPE**: Type of geometry for the segment. The following types are implemented:

**BELLMOUTH** : Bellmouth.

**CYLINDER** : Cylinder.

**DTRUMPET** : Double trumpet.

**GENERAL** : General.

**FROM \_ FILE** : From file.

For **TYPE = DTRUMPET** the format is:

```
GEOM NAME NSEG DTRUMPET THETA NVIS LEN DIAM RCURV
```

.... ....

where:

**THETA**: Rotation about local x-axis. This rotation is relative to the previous segment. For the first segment, the rotation is relative to the orientation of the body element.

**NVIS**: Number of visual sections.

**LEN**: Length of the segment.

**DIAM**: Smallest diameter of segment.

**RCURV**: Radius of curvature for the double trumpet.

TIME=0.0

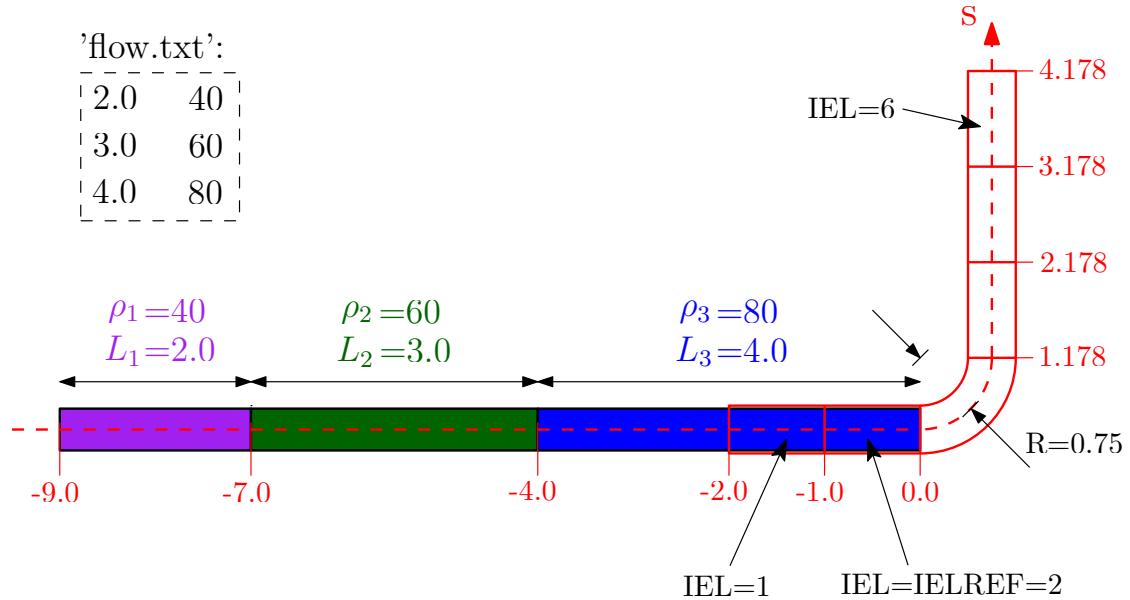
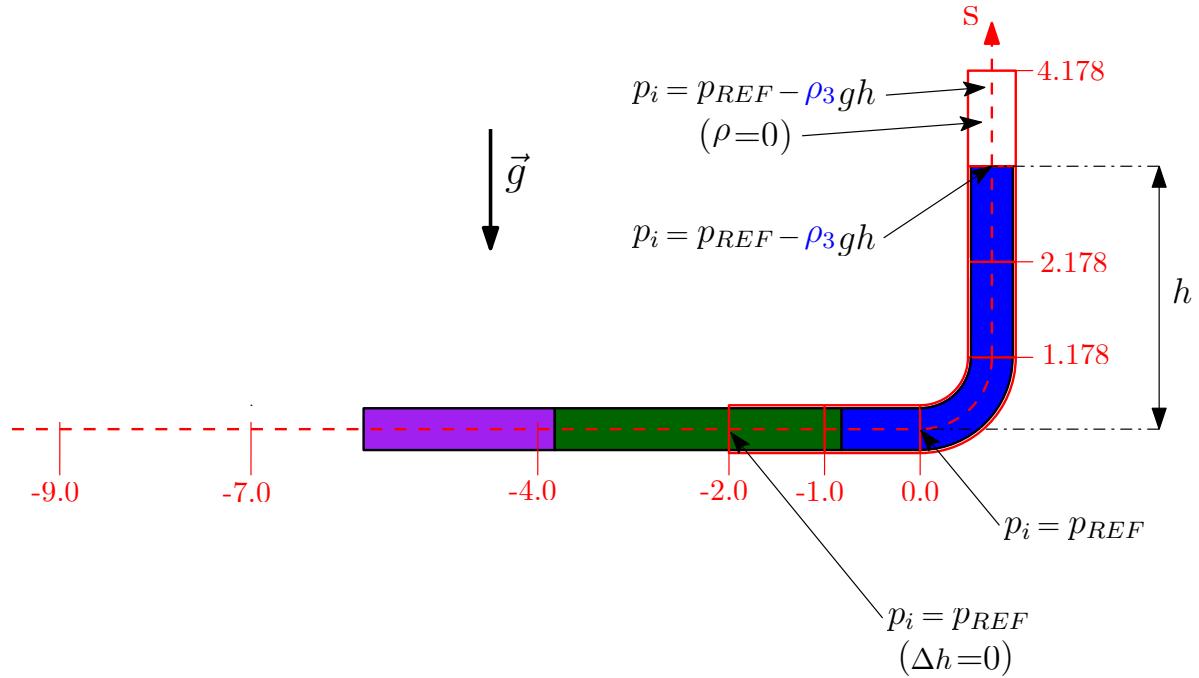
TIME=3.178,  $U=1.0$ 

Figure 3.32: The file 'flow.txt' defines the three fluid segments with lengths  $L_i$  in the first column and densities  $\rho_i$  in the second column. End 2 of element **IELREF** defines the origin of the curvilinear coordinate  $s$  where the reference static internal pressure **PREF** is prescribed. The static pressure elsewhere is calculated based on the vertical height difference  $h$  and by considering the fluid densities  $\rho_i$  along the slug flow.

## EXAMPLE:

---

```
#      name    nseg   type       theta   nvis   len    diam   rcurv
GEOM  geo1     1    DTRUMPET  0.0     4     3.5   0.17   4.2
```

---

For **TYPE = BELLMOUTH** the format is:

**GEOM NAME NSEG BELLMOUTH THETA NVIS LEN DIAM RADSTART RADEND PHI0**

.... ....

where:

**THETA**: Rotation about local x-axis. This rotation is relative to the previous segment. For the first segment, the rotation is relative to the orientation of the body element.

**NVIS**: Number of visual sections.

**LEN**: Length of the segment. (unit: L)

**DIAM**: Smallest diameter of segment. (unit: L)

**RADSTART**: Curvature radius at start of segment. (unit: L)

**RADEND**: Rate of curvature radius change along segment so that

$1/RAD=1/RADSTART+1/RADEND*s$  where s is curvilinear length along segment  
(unit:  $L^2$ )

**PHI0**: Slope at start of segment. The slope is measured along the longitudinal direction, and relative to the center line of the bellmouth.

The curvature of the bellmouth is taken to be zero if the end curvature and the start curvature is estimated to differ by less than a hundredth of the start curvature.

## EXAMPLE:

---

```
# Geometry data
#
#      name    nseg       type   theta   nvis    len    diam  radstart radend   phi0
GEOM  geo1     2    bellmouth  0.0     10    2.0   0.1650  150.0     0.0   0.0000
                                bellmouth  0.0     10    2.0   0.1917   20.0     0.0   0.0133
```

---

For **TYPE = CYLINDER** the format is:

**GEOM NAME NSEG CYLINDER THETA NVIS LEN DIAM1 DIAM2**

.... ....

where:

**THETA**: Rotation about local x-axis. This rotation is relative to the previous segment. For the first segment, the rotation is relative to the orientation of the body element.

**NVIS**: Number of visual sections.

**LEN**: Length of the segment.

**DIAM1**: Diameter of first segment.

**DIAM2**: Diameter of last segment.

EXAMPLE:

---

```
#      name    nseg     type      theta    nvvis   len    diam1    diam2
GEOM  cyl12     1    CYLINDER   0.0      4       3.5    0.17    0.17
```

---

For **TYPE = GENERAL** the format is:

**GEOM NAME NSEG GENERAL CROSSGEOMNAME X0 Y0 Z0 TX TY TZ**

..... .....

where:

**CROSSGEOMNAME**: Name of the cross sectional geometry

**X0**: *x*-coordinate for position vector relative to body element system.

**Y0**: *y*-coordinate for position vector relative to body element system.

**Z0**: *z*-coordinate for position vector relative to body element system.

**TX**: Tait–Bryan angle around *x*-axis. Tait–Bryan angle for *x*-axis relative to body element system.

**TY**: Tait–Bryan angle around *y*-axis. Tait–Bryan angle for *y*-axis relative to body element system.

**TZ**: Tait–Bryan angle around *z*-axis. Tait–Bryan angle for *z*-axis relative to body element system.

EXAMPLE:

---

```
#      name    nseg     type      crossgeomname  xo    yo    zo    tx    ty    tz
GEOM  gen1     1    GENERAL        boxsec      0.    0.    0.    0.    0.    0.
```

---

For **TYPE = FROM \_ FILE** the format is:

**GEOM NAME NSEG FROM \_ FILE FILENAME X0 Y0 Z0 TX TY TZ**

..... .....

where:

**FILENAME:** Name of the file where 3D geometry is stored

**X0:**  $x$ -coordinate for position vector relative to body element system.

**Y0:**  $y$ -coordinate for position vector relative to body element system.

**Z0:**  $z$ -coordinate for position vector relative to body element system.

**TX:** Tait–Bryan angle for  $x$ -axis. Tait–Bryan angle for  $x$ -axis relative to body element system.

**TY:** Tait–Bryan angle for  $y$ -axis. Tait–Bryan angle for  $y$ -axis relative to body element system.

**TZ:** Tait–Bryan angle for  $z$ -axis. Tait–Bryan angle for  $z$ -axis relative to body element system.

EXAMPLE: \_\_\_\_\_

```
#      name    nseg     type      filename xo   yo   zo   tx   ty   tz
GEO M  gen1     1    from_file    "vessel1.txt"  0.   0.   0.   0.   0.   0.
```

The file vessel1.txt starts with one line with 3 numbers:

**NEL** number of elements, **NENOD** number of element nodes  
(3 for triangles and 4 for rectangles) and **NNOD** number of  
nodes in the model. Then comes one line with the parameter **RB**  
that defines the radius of edges and corners used in relation to  
the **CONT153** element as described in Section 3.16. Then comes the  
topology and at last the  $x$ -,  $y$ - and  $z$ -coordinates. The example file  
content below defines a box wih 6 elements and 8 nodes:

```
6 4 8
0.05
1 2 3 4
1 2 6 5
2 3 7 6
3 7 8 4
1 4 8 5
5 6 7 8
-50 -15 0
-50 15 0
90 15 0
90 -15 0
-50 -15 10
-50 15 10
90 15 10
90 -15 10
```

### 3.29 HEAD - HEADING

The **HEAD** command is used to supply a descriptive text to the analysis. Up to six lines of text may be used to describe the analysis. Each line is started with **HEAD** followed by an arbitrary text string. The format is as follows:

```
HEAD TEXT
```

where:

**TEXT**: Descriptive analysis text.

EXAMPLE: \_\_\_\_\_

```
#  
HEAD This is a test example  
HEAD Elastic material model - 440 pipe elements  
#
```

---

### 3.30 HLA - lay simulation scenario definition

The following format is applied to define the **HLA** mode:

```
HLA HOST PORT FEDERATION FEDERATE [LOGFILE]
```

where:

**HOST**: Computer name.

**PORT**: Port number.

**FEDERATION**: Simulation set-up name.

**FEDERATE**: Numerical simulation name.

**LOGFILE**: Log file, optional. If a log file is specified the HLA simulation will run in dummy mode, and produce a log file with the specified name. The log file may later be played off in the SimVis visualization module.

EXAMPLE: \_\_\_\_\_

```
#      3.22 Lay simulation data  
#      host          port   federation   federate   vessel   logfile  
HLA  "127.0.0.1"  0       "test"        "bj-40"     "lb200"    "bj-oilt-04_msupp.ldat"
```

---

### 3.31 HLAPLOT - HLA time plot results

By the **HLAPLOT** card, user selected results related to soil and roller displacements and forces can be plotted in time domain in the Simvis user interface. If the **UNITS**-card (analysis units) is given the units of HLAPLOTS will be as given below. If not, the units will be corresponding to the units used in the analysis.

The following format is applied:

```
HLAPLOT TYPE ...
```

where

**TYPE**: Type of plot.

Depending on the value of **TYPE**, the following formats applies:

**Roller displacement results [unit: L]:**

```
HLAPLOT ROLLDISP DOF ELID1 ELID2
```

**DOF**: DOF number, max. 3, **1** = tangential, **2** = transverse and **3** = normal.

**ELID1**: Element 1 ID.

**ELID2**: Element 2 ID.

**Roller force results [unit: F]:**

```
HLAPLOT ROLLFORCE DOF ELID1 ELID2
```

**DOF**: DOF number, max. 3, **1** = tangential, **2** = transverse and **3** = normal.

**ELID1**: Element 1 ID.

**ELID2**: Element 2 ID.

**Mean transverse displacement for touch down region (deviation from theoretical route and displacement of soil-spring) [unit: L]:**

```
HLAPLOT ROUTEDEV L
```

**L**: Length over which the mean displacement is calculated (optional). If no value is given the characteristic length of the pipe is used.

**Touch down mobilization factor results [-]:**

```
HLAPLOT SOILMOB L
```

- L:** Length over which the mob. factor is calculated. Should not be taken larger than:  $L \leq \sqrt{\frac{12EI}{5qR}}$  where  $EI$  is the bending stiffness [ $\text{FL}^2$ ],  $R$  is the curve radius [L] and  $q$  is the transverse resistance [ $\text{FL}^{-1}$ ]

*Note:* Stability plots will be generated for all the lines given in the route-file, see Section 3.8. For additional lines the friction factor will be reported as zero if the transverse force component is acting towards the theoretical route, i.e. only cases where the pipe tends to slide away from the main route will be reported as critical. A safe configuration should hence have an acceptable friction factor for additional lines on both sides of the theoretical route.

#### Touch down and top tension results [kN]:

**HLAPLOT TENSION DS DD**

**DS:** Submerged distance from top node to sea level (unit: L)

**DD:** Dry distance from sea level to tensioner (unit: L). The reported tension will then be:  $T = T_{topnode} + w_s r3 + w_d r4$

#### Sagbend utilization results [-]:

**HLAPLOT SAGUTIL OPT VALUE**

**OPT:** Utilization option. Permitted values are:

**CURV :** Utilization factor is based on minimum allowed radius of curvature.

**STRESS :** Utilization factor is based on yield stress limit.

**VALUE:** Min. radius or yield stress, depending on which option is used. (Units: L or F/L<sup>2</sup>)

#### Departure angle results [deg]:

**HLAPLOT DEPANG**

#### Distance to specified roll [m]:

**HLAPLOT ROLLDIST IROLL**

**IROLL:** Roll element number.

#### Distance to nominal position in tower (rollerbox) [m]:

**HLAPLOT TOWERDIST**

Force acting on pipe (transverse and normal) between two given KP-values [kN]:

```
HLAPLOT SUPFORCE PLOTID KPSTART KPEND
```

**PLOTID:** Plot identifier.

**KPSTART:** KP-start.

**KPEND:** KP-end.

Horizontal distance from TDP to last node on pipe (connection to vessel) [m] :

```
HLAPLOT LAYBACK
```

**Distance to sea bed [m]:** Can be used under **STATIC-SDD** analysis, see Section 3.51.

```
HLAPLOT SEABDGAP NODID
```

**NODID:** Node number.

**Node position relative to tdp [m]** Position of a node relative to tdp, in a coordinate system where the x-axis is the route tangent at tdp, and the z-axis is vertical. Can be used under **STATIC-SDD** analysis, see Section 3.51.

```
HLAPLOT NODEPOS NODID
```

**NODID:** Node number.

**S-length [m]** Curvilinear distance of wire and the umbilical section from tdp to vessel. Can be used under **STATIC-SDD** analysis, see Section 3.51.

```
HLAPLOT S-LENGTH
```

**Torque moment in tensioner [kNm]:** In addition to torque moment in tensioner, torque moment in pipe at TDP is plotted.

```
HLAPLOT TENSTORQUE
```

**Accumulated rotation in tensioner [deg]:** In addition to accumulated rotation in tensioner, rotation of pipe (relative to tensioner exit) at TDP is plotted.

```
HLAPLOT TENSROT
```

Some examples are shown below.

EXAMPLE:

---

```
HLAPLOT SOILMOB 40
HLAPLOT TENSION 0 0
HLAPLOT ROUTEDEV
HLAPLOT ROLLFORCE 3 2301 2314
HLAPLOT SUPFORCE KP101570-580 101570 101580
```

---

### 3.32 HLAVIS - General HLA output

To obtain HLA output in the general case the objects intended for output to HLA must be specified.

If the **UNITS**-card (analysis units) is given the units of the HLA-results are:

tension :	kN	
submerged weight :	kg/m	If not, units are according to units used in analysis model.
stress:	MPa	
reaction force:	kN	

The following format is applied to define the object intended for output to HLA:

**HLAVIS OBJTYPE ...**

where:

**OBJTYPE**: Object type for HLA visualization. May have values **BODY**, **CPIPE** or **TPIPE**.

For **OBJTYPE=BODY** the format is as follows:

**HLAVIS BODY TYPE [NODID] NAME**

**TYPE**: Can have values **NODE** or **TDP**. The object will follow the specified node number or the touch down point location location.

For **TYPE=NODE**:

**NODID**: Node number.

**NAME**: Name of object, referred to in SIMVIS play file.

For **TYPE=TDP** only **NAME** must be given:

**NAME**: Name of object, referred to in SIMVIS play file.

For **OBJTYPE=CPIPE** (constant pipe) the format is as follows:

**HLAVIS CPIPE ELGRP NODE1 NODE2**

**ELGRP**: Element group name.

**NODE1**: First node ID number in visual model (optional).

**NODE2**: Last node ID number in visual model (optional).

For **OBJTYPE=TPIPE** (tailed pipe) the format is the same as for **CPIPE**:

**HLAVIS TPIPE ELGRP NODE1 NODE2**

**ELGRP**: Element group name.

**NODE1**: First node ID number in visual model (optional).

**NODE2**: Last node ID number in visual model (optional).

#### EXAMPLE:

---

```
#      Type    ID     Head
HLAVIS VESSEL 3001  2.437945
#      Type    Grp ID      node 1   node n
HLAVIS PIPE    ormpipe1    1       401
HLAVIS PIPE    ormpipe2    401     501
```

---

### 3.33 HYDROPRO - Hydrodynamic properties

If the BODY element is used to represent a trawl board, it may be necessary to let the hydrodynamic properties be a function of the orientation of the trawl board and the height above the seabed. The **HYDROPRO** enables the user to either substitute or add to the hydrodynamic properties defined in the **ELPROP** card.

Each element property type is defined by the following format:

**HYDROPRO ELGRP TYPE ...**

where

**ELGRP**: Element group name.

**TYPE**: Hydrodynamic property type as defined below.

The following type is allowed for:

**TRAWLBOARD** : Trawlboard properties (BODY502 element type)

The number of and definition of the following parameters depend on the choice of hydrodynamic property type.

### 3.33.1 TRAWLBOARD

```
HYDROPRO ELGRP TRAWLBOARD CONTSURF IREPLACE SCALEFACT  
MECCX MECCY MECCZ DECCX DECCY DECCZ NNOD NMASS NDRAG NODEI  
.... MASSFILEI ... DRAGFILEI [IROTDRAg CDN LX LY LZ]
```

**CONTSURF**: Name of the seabed contact surface

**IREPLACE**: Replace option for hydrodynamic mass coefficients in DOF 1-6 and hydrodynamic drag coefficients in DOF 1, 2 and 3, where allowable values are:

- 0** : The **ELPROP** hydrodynamic coefficients are added to the **HYDROPRO** coefficients
- 1** : The **ELPROP** hydrodynamic coefficients are replaced by the **HYDROPRO** coefficients

**SCALEFACT**: Trawlboard size scaling factor (unit: -)

**MECCX**: Local x-eccentricity from element origin to hydrodynamic mass center point (unit: L)

**MECCY**: Local y-eccentricity from element origin to hydrodynamic mass center point (unit: L)

**MECCZ**: Local z-eccentricity from element origin to hydrodynamic mass center point (unit: L)

**DECCX**: Local x-eccentricity from element origin to hydrodynamic drag center point (unit: L)

**DECCY**: Local y-eccentricity from element origin to hydrodynamic drag center point (unit: L)

**DECCZ**: Local z-eccentricity from element origin to hydrodynamic drag center point (unit: L)

**NNOD** : Number of nodes for seabed distance measurements

**NMASS**: Number of added mass files, max 6

**NDRAG**: Number of drag files, max 6

**NODEI**: Node ID numbers for measurement of height above seabed. The height will be taken as the minimum height based on these nodes

**MASSFILEI**: File name (max 6) of stored added mass coefficients for the DOFs surge, sway, heave, roll, pitch, yaw

**DRAGFILEI**: File name (max 6) of stored drag coefficients for the DOFs surge, sway, heave, roll, pitch, yaw

**IROTDRAg**: Rotational drag model type, optional, default value: 0. Allowable values:

- 0** : Standard rotational drag model based on the **ELPROP** rotational drag coefficient.
- 1** : Plate rotational drag model for DOF 5 and 6, while DOF 4 apply the same drag model as for **IROTDRAG=0**. The plate surface normal is directed along DOF 1, implying that the plate dimensions are defined by **LY** and **LZ** with **LX** being a dummy parameter.
- 2** : Plate rotational drag model for DOF 4 and 6, while DOF 5 apply the same drag model as for **IROTDRAG=0**. The plate surface normal is directed along DOF 2, implying that the plate dimensions are defined by **LX** and **LZ** with **LY** being a dummy parameter.
- 3** : Plate rotational drag model for DOF 4 and 5, while DOF 6 apply the same drag model as for **IROTDRAG=0**. The plate surface normal is directed along DOF 3, implying that the plate dimensions are defined by **LX** and **LY** with **LZ** being a dummy parameter.

**CDN**: Drag coefficient in plate surface normal direction applied for **IROTDRAG=1, 2, 3**, optional, default value: 0.0 (unit: -).

**LX**: Plate length in x-direction, optional, default value: 0.0 (unit: L).

**LY**: Plate length in y-direction, optional, default value: 0.0 (unit: L).

**LZ**: Plate length in z-direction, optional, default value: 0.0 (unit: L).

DOFs not included in a **MASSFILE** will be assigned the **ELPROP** added mass. Likewise, DOFs not included in a **DRAGFILE** will be assigned the drag coefficients from the **ELPROP** card.

The **IREPLACE** parameter does not apply for the drag moments in DOF 4, 5 and 6 that are modelled by **IROTDRAG=1, 2, 3** or the standard **ELPROP** drag coefficients. This is because these drag moments are based on angular velocity and are consequently not related to the **HYDROPRO** drag and lift coefficients based on linear velocity.

The **SCALEFACT** parameter  $\lambda$  scales the added mass  $m_{ai}$  and the drag coefficients  $C_i$  specified in the input files according to:

$$\bar{m}_{ai} = \lambda^3 m_{ai} \quad i = 1, 2, 3 \quad (3.20)$$

$$\bar{m}_{ai} = \lambda^5 m_{ai} \quad i = 4, 5, 6 \quad (3.21)$$

$$\bar{C}_i = \lambda^2 C_i \quad i = 1, 2, 3 \quad (3.22)$$

$$\bar{C}_i = \lambda^3 C_i \quad i = 4, 5, 6 \quad (3.23)$$

where  $i$  refers to the DOF, and  $\bar{m}_{ai}$  and  $\bar{C}_i$  represent the values that will be applied in the analysis. The above scaling is only valid for trawl boards that have the same aspect ratio.

### 3.33.2 Flat plate rotational drag model

The purpose of the rotational drag model for **IROTDRAG=1, 2, 3** is to account for increased rotational drag due to linear velocity normal to the plate. For this model, the external drag moments referred to the element node are computed by integration of drag force contributions over a thin flat rectangular plate surface according to,

$$\mathbf{M}_{ext} = -\frac{1}{2}\rho C_{dn} \iint_A |\dot{\mathbf{v}}_r \cdot \mathbf{n}| (\dot{\mathbf{v}}_r \cdot \mathbf{n}) \mathbf{r}_l \times \mathbf{n} dA \quad (3.24)$$

where  $\mathbf{n}$  is the plate surface unit normal vector,  $C_{dn} = \text{CDN}$ ,  $\rho$  is the water density,  $A$  is the plate area in the normal direction  $\mathbf{n}$  defined by **LX**, **, and  $\mathbf{r}_l$  is the position vector from the **ELPROP** hydrodynamic center to the plate surface point. Regarding  $\mathbf{r}_l$ , the plate area center is assumed to be located at the **ELPROP** hydrodynamic center as defined in Section 3.23.5. Further, the element node may be eccentric relative to the element origin as defined by the **ELECC** command in Section 3.18. The body linear velocity  $\dot{\mathbf{v}}_r$  at the plate surface relative to the fluid is computed as,**

$$\dot{\mathbf{v}}_r = \dot{\mathbf{v}}_0 + \boldsymbol{\omega} \times \mathbf{r} - \dot{\mathbf{v}}_f \quad (3.25)$$

where  $\dot{\mathbf{v}}_0$  is the body linear velocity at the element origin,  $\boldsymbol{\omega}$  is the body angular velocity and  $\mathbf{r}$  is the position vector from the element origin to the plate surface point. The fluid velocity  $\dot{\mathbf{v}}_f$  is taken to be constant over the body. See the SIMLA Theory Manual for further details.

Equation (3.24) gives zero moment for the component of  $\mathbf{M}$  that is parallel to  $\mathbf{n}$ . For this moment component, the model for **IROTDRAG=0** is applied instead. A resulting drag force is associated with Equation (3.24) provided that the body rotation point does not incidentally coincide with the **ELPROP** hydrodynamic center. This drag force is not included in the plate rotational drag model and must therefore be modelled by the **HYDROPRO** drag files or the **ELPROP** drag coefficients in DOFs 1, 2 or 3. This is also the reason for applying the position vector  $\mathbf{r}_l$  instead of  $\mathbf{r}$  in Equation (3.24), otherwise an additional drag moment would arise for a hydrodynamic center that is located eccentric relative to the element node. Such a drag moment must instead be modelled by the **HYDROPRO** drag files or the **ELPROP** drag coefficients in DOFs 1, 2 and 3 combined with an eccentric hydrodynamic center relative to the element node.

### 3.33.3 Interpolation of hydrodynamic mass

The hydrodynamic mass is interpolated in terms of Euler angles that describe the seabed proximity of the body together with the height  $h$  above the seabed. See the SIMLA Theory Manual for details about the applied Euler angle definition.

The following rules apply for the interpolation scheme used for hydrodynamic mass:

- The hydrodynamic mass is independent of rotation about the seabed surface unit normal vector  $\mathbf{N}$ . This is because the seabed proximity of the body is unaffected if the body rotates about  $\mathbf{N}$ . Hence, only two Euler angles are needed in the interpolation scheme.
- The seabed coordinate system is defined as follows: The  $Z$ -axis is directed along the seabed unit normal vector  $\mathbf{N}$  as defined by the contact surface data file, see the **COSURFPR** command in Section 3.8. The directions of the  $X$ -axis and  $Y$ -axis are set arbitrarily by SIMLA as the hydrodynamic mass is independent of rotation about the seabed surface normal vector.
- The Euler angles represent the motion that rotates the seabed coordinate system into the body element coordinate system. The rotation sequence is as follows: 1.rotation  $\theta_x$  about the unrotated  $X$ -axis of the seabed coordinate system, 2.rotation  $\theta_y$  about the unrotated  $Y$ -axis of the seabed coordinate system, and 3.rotation  $\theta_z$  about the unrotated  $Z$ -axis of the seabed coordinate system. The hydrodynamic mass is not a function of  $\theta_z$ , and hence this Euler angle is not involved in the interpolation scheme.
- The  $z$ -axis of the body system and the  $Z$ -axis of the seabed coordinate system coincide for  $\theta_x = 0$  degree and  $\theta_y = 0$  degree. Hence, the body must be defined such that the body element  $z$ -axis represents the upward axis of the trawl board.
- The trawl board geometry is assumed to obey the following symmetry properties for the hydrodynamic mass  $m_i(\theta_x, \theta_y)$  in terms of the full range of Euler angles  $\theta_x \in [-180 \text{ deg}, 180 \text{ deg}]$  and  $\theta_y \in [-90 \text{ deg}, 90 \text{ deg}]$ :
  - Symmetry about  $\theta_x = 0 \text{ deg}$ :  $m_i(\theta_x = -5 \text{ deg}, \theta_y) = m_i(\theta_x = 5 \text{ deg}, \theta_y)$
  - Symmetry about  $\theta_x = 90 \text{ deg}$ :  $m_i(\theta_x = 85 \text{ deg}, \theta_y) = m_i(\theta_x = 95 \text{ deg}, \theta_y)$
  - Symmetry about  $\theta_x = -90 \text{ deg}$ :  $m_i(\theta_x = -85 \text{ deg}, \theta_y) = m_i(\theta_x = -95 \text{ deg}, \theta_y)$
  - Symmetry about  $\theta_x = -180 \text{ deg}$ :  $m_i(\theta_x = -175 \text{ deg}, \theta_y) = m_i(\theta_x = 175 \text{ deg}, \theta_y)$
  - Symmetry about  $\theta_y = 0 \text{ deg}$ :  $m_i(\theta_x, \theta_y = 25 \text{ deg}) = m_i(\theta_x, \theta_y = -25 \text{ deg})$

The symmetry relations above are valid if the hydrodynamic mass is based on a box geometry.

- Due to symmetry, the hydrodynamic mass is specified on the definition range  $\theta_x \in [0 \text{ deg}, 90 \text{ deg}]$  and  $\theta_y \in [0 \text{ deg}, 90 \text{ deg}]$ . Angles outside these intervals are not allowed.
- If the hydrodynamic mass is not specified on the full definition range  $[0 \text{ deg}, 90 \text{ deg}]$ , the value at the end points will be applied if the computed interpolation variable

$\theta_x$  and  $\theta_y$  are outside of the user-specified definition ranges.

The computed Euler angles handle the case of seabed surface normal vectors that are not aligned with the global  $z$ -axis. This means that the hydrodynamic mass can be correctly modelled at for instance the shoulders of free spans, provided that the variation of the seabed normal vector is not too severe and that the seabed geometry is described with sufficient accuracy.

The third interpolation variable is the height  $h$  above the seabed. This height is taken to be the minimum height based on the **NODEI** nodes as follows,

$$h = z_n - z_s \quad (3.26)$$

where  $z_n$  is the global  $z$ -coordinate of the node and  $z_s$  is the global  $z$ -coordinate of the seabed point below the node.

Time history results of the interpolation variables  $h$ ,  $\theta_x$  and  $\theta_y$ , and the interpolated hydrodynamic mass are available from the **DYNRES\_E** card in Section 3.12.2.

### 3.33.4 Interpolation of drag and lift loads

The drag and lift coefficients are interpolated in terms of two angles and the height  $h$  above the seabed. The angles can be selected as hydrodynamic angles of attack based on the body velocity relative to the fluid, and/or as geometric angles describing the orientation of the body relative to the seabed unit normal vector. These angles are illustrated and defined in the SIMLA Theory Manual.

The hydrodynamic angles of attack are based on the body linear velocity relative to the fluid computed as,

$$\dot{\mathbf{v}}_{rd} = \dot{\mathbf{v}}_0 + \omega \times \mathbf{r}_d - \dot{\mathbf{v}}_f \quad (3.27)$$

where  $\dot{\mathbf{v}}_0$  is the body linear velocity at the element origin,  $\omega$  is the body angular velocity and  $\mathbf{r}_d$  is the position vector from the element origin to the drag hydrodynamic center defined by **DECCX**, **DECCY** and **DECCZ**. The fluid velocity  $\dot{\mathbf{v}}_f$  is taken to be constant over the body.

The body linear velocity vector relative to the fluid has the following components in the body element system,

$$\dot{\mathbf{v}}_{rd} = \begin{bmatrix} \dot{v}_x \\ \dot{v}_y \\ \dot{v}_z \end{bmatrix} \quad (3.28)$$

Further, the hydrodynamic angle of attack about the body element  $x$ -axis is defined as,

$$\alpha_x = \begin{cases} 2\pi - \cos^{-1} \left( \frac{\dot{v}_z}{\sqrt{\dot{v}_y^2 + \dot{v}_z^2}} \right) & \text{if } \dot{v}_y < 0 \\ \cos^{-1} \left( \frac{\dot{v}_z}{\sqrt{\dot{v}_y^2 + \dot{v}_z^2}} \right) & \text{if } \dot{v}_y \geq 0 \\ 0 & \text{if } \dot{v}_y = \dot{v}_z = 0 \end{cases} \quad (3.29)$$

and the hydrodynamic angle of attack about the body element  $y$ -axis,

$$\alpha_y = \begin{cases} 2\pi - \cos^{-1} \left( \frac{\dot{v}_x}{\sqrt{\dot{v}_x^2 + \dot{v}_z^2}} \right) & \text{if } \dot{v}_z < 0 \\ \cos^{-1} \left( \frac{\dot{v}_x}{\sqrt{\dot{v}_x^2 + \dot{v}_z^2}} \right) & \text{if } \dot{v}_z \geq 0 \\ 0 & \text{if } \dot{v}_x = \dot{v}_z = 0 \end{cases} \quad (3.30)$$

and the hydrodynamic angle of attack about the body element  $z$ -axis,

$$\alpha_z = \begin{cases} 2\pi - \cos^{-1} \left( \frac{\dot{v}_x}{\sqrt{\dot{v}_x^2 + \dot{v}_y^2}} \right) & \text{if } \dot{v}_y < 0 \\ \cos^{-1} \left( \frac{\dot{v}_x}{\sqrt{\dot{v}_x^2 + \dot{v}_y^2}} \right) & \text{if } \dot{v}_y \geq 0 \\ 0 & \text{if } \dot{v}_x = \dot{v}_y = 0 \end{cases} \quad (3.31)$$

The geometric angles describing the body orientation relative to the seabed surface unit normal vector  $\mathbf{N}$  are defined as follows,

$$\beta_x = \cos^{-1} (\mathbf{e}_x \cdot \mathbf{N}) \quad (3.32)$$

$$\beta_y = \cos^{-1} (\mathbf{e}_y \cdot \mathbf{N}) \quad (3.33)$$

$$\beta_z = \cos^{-1} (\mathbf{e}_z \cdot \mathbf{N}) \quad (3.34)$$

where  $\mathbf{e}_x$ ,  $\mathbf{e}_y$  and  $\mathbf{e}_z$  are the unit vectors along the body element  $x$ -,  $y$ - and  $z$ -axis, respectively. The seabed normal vector  $\mathbf{N}$  is defined by the contact surface data file, see the **COSURFP** command in Section 3.8.

The following rules apply for the interpolation scheme used for drag and lift coefficients:

- The coefficients can be specified on the following definition range in terms of the hydrodynamic angles of attack,  $\alpha_x \in [0 \text{ deg}, 360 \text{ deg}]$  and  $\alpha_y \in [0 \text{ deg}, 360 \text{ deg}]$  and  $\alpha_z \in [0 \text{ deg}, 360 \text{ deg}]$ . Angles outside these intervals are not allowed.
- The coefficients can be specified on the following definition range in terms of the geometric angles,  $\beta_x \in [0 \text{ deg}, 180 \text{ deg}]$  and  $\beta_y \in [0 \text{ deg}, 180 \text{ deg}]$  and  $\beta_z \in [0 \text{ deg}, 180 \text{ deg}]$ . Angles outside these intervals are not allowed.

- If the drag and lift coefficients are not specified on the complete definition ranges  $\alpha_i \in [0 \text{ deg}, 360 \text{ deg}]$  and/or  $\beta_i \in [0 \text{ deg}, 180 \text{ deg}]$ , the value at the end points will be applied if the computed angle is outside of the user-specified definition range.
- Only two of the angles  $\alpha_x, \alpha_y, \alpha_z, \beta_x, \beta_y$  and  $\beta_z$  can be selected as interpolation variables.

The third interpolation variable for the drag and lift coefficients is the height  $h$  above the seabed. This variable is described in Section 3.33.3.

Time history results of the interpolation variables  $h, \alpha_x, \alpha_y, \alpha_z, \beta_x, \beta_y$  and  $\beta_z$ , and the interpolated drag and lift loads are available from the **DYNRES\_E** card in Section 3.12.2.

### 3.33.5 File input format for hydrodynamic coefficients

The **MASSFILEI** and **DRAGFILEI** data files define the hydrodynamic coefficients in terms of the interpolation variables.

The hydrodynamic mass coefficients  $m_{ai}$  are specified in **MASSFILEI** as functions of the Euler angles  $\theta_x$  and  $\theta_y$ , and the height above the seabed  $h$  defined in Equation (3.26). The coefficients will be scaled by the sea water density  $\rho$  by SIMLA according to,

$$\bar{m}_{ai}(\theta_x, \theta_y, h) = \rho m_{ai}(\theta_x, \theta_y, h) \quad i = 1, 2, \dots, 6 \quad (3.35)$$

which means that the units of the coefficients are  $L^3$  for translation DOFs and  $L^5$  for rotation DOFs.

The example below shows the format of the hydrodynamic mass file for the surge DOF.

EXAMPLE:

---

```

ADDED MASS DATA
SURGE
HEIGHTS 0 0.5 1.5
ROWS EULER-X 0 30 45 90
COLUMNS EULER-Y 0 10 20 30 45 60 90
 0.9 0.8 0.5 0.5 0.5 0.5 0.5
 0.9 0.8 0.5 0.5 0.5 0.5 0.5
 0.9 0.8 0.5 0.5 0.5 0.5 0.5
 0.9 0.8 0.5 0.5 0.5 0.5 0.5

 0.7 0.8 0.5 0.5 0.5 0.5 0.5
 0.7 0.8 0.5 0.5 0.5 0.5 0.5
 0.7 0.8 0.5 0.5 0.5 0.5 0.5
 0.7 0.8 0.5 0.5 0.5 0.5 0.5

 0.5 0.5 0.5 0.5 0.5 0.5 0.5

```

---

```
0.5 0.5 0.5 0.5 0.5 0.5 0.5
0.5 0.5 0.5 0.5 0.5 0.5 0.5
0.5 0.5 0.5 0.5 0.5 0.5 0.5
```

---

The first block of data above defines the hydrodynamic mass at height  $h=0$ , where the four rows defines the hydrodynamic mass coefficient as a function of the Euler angle  $\theta_x$  and the seven columns defines the coefficient as a function of the Euler angle  $\theta_y$ .

The second block of data defines the hydrodynamic mass coefficient at height  $h = 0.5$  where the rows and columns define the coefficient as a function of the Euler angles  $\theta_x$  and  $\theta_y$ , respectively. Similarly, the third block of data defines the hydrodynamic mass coefficient at height  $h = 1.5$ .

The Euler angles  $\theta_x$  and  $\theta_y$  may be interchanged. The example below shows an equivalent description of the hydrodynamic mass coefficients as the above example, where the Euler angles for the row and column data have been interchanged,

**EXAMPLE:**

---

```
ADDED MASS DATA
SURGE
HEIGHTS 0 0.5 1.5
ROWS EULER-Y 0 10 20 30 45 60 90
COLUMNS EULER-X 0 30 45 90
0.9 0.9 0.9 0.9
0.8 0.8 0.8 0.8
0.5 0.5 0.5 0.5
0.5 0.5 0.5 0.5
0.5 0.5 0.5 0.5
0.5 0.5 0.5 0.5
0.5 0.5 0.5 0.5

0.7 0.7 0.7 0.7
0.8 0.8 0.8 0.8
0.5 0.5 0.5 0.5
0.5 0.5 0.5 0.5
0.5 0.5 0.5 0.5
0.5 0.5 0.5 0.5
0.5 0.5 0.5 0.5

0.5 0.5 0.5 0.5
0.8 0.5 0.5 0.5
0.5 0.5 0.5 0.5
0.5 0.5 0.5 0.5
0.5 0.5 0.5 0.5
0.5 0.5 0.5 0.5
```

0.5 0.5 0.5 0.5

---

The 6 DOF lift and drag external load vector is computed by SIMLA as follows,

$$\mathbf{F}_{ext}(\alpha_i, \alpha_j, \beta_k, \beta_m, h) = \begin{bmatrix} F_1(\alpha_i, \alpha_j, \beta_k, \beta_m, h) \\ F_2(\alpha_i, \alpha_j, \beta_k, \beta_m, h) \\ F_3(\alpha_i, \alpha_j, \beta_k, \beta_m, h) \\ F_4(\alpha_i, \alpha_j, \beta_k, \beta_m, h) \\ F_5(\alpha_i, \alpha_j, \beta_k, \beta_m, h) \\ F_6(\alpha_i, \alpha_j, \beta_k, \beta_m, h) \end{bmatrix} = \frac{1}{2}\rho \begin{bmatrix} C_1(\alpha_i, \alpha_j, \beta_k, \beta_m, h) \\ C_2(\alpha_i, \alpha_j, \beta_k, \beta_m, h) \\ C_3(\alpha_i, \alpha_j, \beta_k, \beta_m, h) \\ C_4(\alpha_i, \alpha_j, \beta_k, \beta_m, h) \\ C_5(\alpha_i, \alpha_j, \beta_k, \beta_m, h) \\ C_6(\alpha_i, \alpha_j, \beta_k, \beta_m, h) \end{bmatrix} ||\dot{\mathbf{v}}_{rd}|| \quad (3.36)$$

where  $\dot{\mathbf{v}}_{rd}$  is defined in Equation (3.27), and  $\rho$  is the sea water density. The drag and lift coefficients are denoted  $C_1$  for the surge DOF,  $C_2$  for the sway DOF,  $C_3$  for the heave DOF,  $C_4$  for the roll DOF,  $C_5$  for the pitch DOF and  $C_6$  for the yaw DOF. The coefficients are functions the hydrodynamic angles of attack  $\alpha_i$  and  $\alpha_j$  defined in Equations (3.29) - (3.31) and/or the geometric angles defined in Equations (3.32) - (3.34), and the height above the seabed  $h$  defined in Equation (3.26).

The drag and lift coefficients  $C_i$  are specified in **DRAGFILEI**. From Equation (3.36), the units of the coefficients are seen to be  $L^2$  for the translation DOFs and  $L^3$  for the rotation DOFs.

The example below shows the file format for the drag and lift coefficients for the sway DOF.

EXAMPLE:

---

```
DRAG DATA
SWAY
HEIGHTS 0 1.5
ROWS HYDROANGLE-Z 0 10 20 30 60 90
COLUMNS HYDROANGLE-X 0 20 40
0.7 0.5 0.5
0.7 0.5 0.5
0.7 0.5 0.5
0.7 0.5 0.5
0.7 0.5 0.5
0.7 0.5 0.5
0.6 0.5 0.5
0.6 0.5 0.5
0.6 0.5 0.5
0.6 0.5 0.5
0.6 0.5 0.5
0.6 0.5 0.5
```

The first block of data above defines the drag and lift coefficient at height  $h=0$ , where the six rows defines the coefficient as a function of the hydrodynamic angle of attack  $\alpha_z$  about the  $z$ -axis and the three columns defines the coefficient as a function of the hydrodynamic angle of attack  $\alpha_x$  about the  $x$ -axis.

The second block of data defines the hydrodynamic mass coefficient at height  $h = 1.5$  where the rows and columns define the coefficient as a function of the hydrodynamic angles of attack  $\alpha_z$  and  $\alpha_x$ , respectively.

The hydrodynamic angles of attack  $\alpha_z$  and  $\alpha_x$  may be interchanged. The example below shows an equivalent description of the drag and lift coefficients as the above example, where the angles of attack for the row and column data have been interchanged,

#### EXAMPLE:

---

```
DRAG DATA
SWAY
HEIGHTS 0 1.5
ROWS HYDROANGLE-X 0 20 40
COLUMNS HYDROANGLE-Z 0 10 20 30 60 90
0.7 0.7 0.7 0.7 0.7 0.7
0.5 0.5 0.5 0.5 0.5 0.5
0.5 0.5 0.5 0.5 0.5 0.5

0.6 0.6 0.6 0.6 0.6 0.6
0.5 0.5 0.5 0.5 0.5 0.5
0.5 0.5 0.5 0.5 0.5 0.5
```

Further, the coefficients may also be defined in terms of the hydrodynamic angle of attack  $\alpha_y$  about the  $y$ -axis as shown in the example below for the pitch DOF.

#### EXAMPLE:

---

```
DRAG DATA
PITCH
HEIGHTS 0 1.5
ROWS HYDROANGLE-Y 0 20 40
COLUMNS HYDROANGLE-Z 0 10 20 30 60 90
0.7 0.7 0.7 0.7 0.7 0.7
0.5 0.5 0.5 0.5 0.5 0.5
0.5 0.5 0.5 0.5 0.5 0.5

0.6 0.6 0.6 0.6 0.6 0.6
0.5 0.5 0.5 0.5 0.5 0.5
```

0.5 0.5 0.5 0.5 0.5 0.5

---

Alternatively, one can use the geometric angles  $\beta_i$  in Equations (3.32) - (3.34) as shown below,

EXAMPLE: \_\_\_\_\_

```
DRAG DATA
PITCH
HEIGHTS 0 1.5
ROWS GEOMANGLE-Y 0 20 40
COLUMNS GEOMANGLE-X 0 10 20 30 60 90
0.7 0.7 0.7 0.7 0.7 0.7
0.5 0.5 0.5 0.5 0.5 0.5
0.5 0.5 0.5 0.5 0.5 0.5

0.6 0.6 0.6 0.6 0.6 0.6
0.5 0.5 0.5 0.5 0.5 0.5
0.5 0.5 0.5 0.5 0.5 0.5
```

---

or alternatively a combination of hydrodynamic angles of attack  $\alpha_i$  and geometric angles  $\beta_i$  as shown below.

EXAMPLE: \_\_\_\_\_

```
DRAG DATA
PITCH
HEIGHTS 0 1.5
ROWS GEOMANGLE-Y 0 20 40
COLUMNS HYDROANGLE-Z 0 10 20 30 60 90
0.7 0.7 0.7 0.7 0.7 0.7
0.5 0.5 0.5 0.5 0.5 0.5
0.5 0.5 0.5 0.5 0.5 0.5

0.6 0.6 0.6 0.6 0.6 0.6
0.5 0.5 0.5 0.5 0.5 0.5
0.5 0.5 0.5 0.5 0.5 0.5
```

---

### 3.34 INISTR - initial strain loading

The initial element strain loading is applied by the following format:

**INISTR HIST DIR IEL1 ISTR1 [IEL2 ISTR2]**  
**[REPEAT N ELINC]**

where

**HIST**: Load history number.

**DIR**: Element load direction relative to the element axis (**1** = axial strain , **4-6** = torsion and curvatures about x-z).

**IEL1**: First element ID.

**ISTR1**: Initial strain for the first element.

**IEL2**: Last element ID, optional.

**ISTR2**: Initial strain for the last element, optional. Linear interpolation is applied for intermediate elements.

If the **REPEAT** command is used then the previous sequence of load generation is repeated:

**N**: Number of repetitions.

**ELINC**: Element increment.

N times with element increment m. To summarise:

<b>DIR</b> has value	<b>ISTR1/ISTR2</b> is
<b>1</b>	axial strain along local x axis (unit: -)
<b>4</b>	the torsion about local x axis (unit: L <sup>-1</sup> )
<b>5</b>	curvature about local y axis (unit: L <sup>-1</sup> )
<b>6</b>	curvature about local z axis (unit: L <sup>-1</sup> )

EXAMPLE:

---

```
#      hist  dir  iel1  torsion
#
INISTR 200  4    100   20.0
                  103   20.0
#
#      n  m
REPEAT 4  4
```

---

### 3.35 JOINTPR\_APPLY - Apply pipeline joint properties

To make a joint table active, see Section 3.36, the **JOINTPR\_APPLY** must be given.

The **JOINTPR\_APPLY** command is defined by the following format:

**JOINTPR\_APPLY JOINTABNAME ELGRP1 ...**

**JOINTABNAME:** Name of joint table.

**ELGRP1:** Name of element group for which **JOINTNAME** is to be used.

The card can have an arbitrary number of **ELGPR** names. If several joint tables are to be applied in the same simulation, each joint table of them must have its own **JOINTPR\_APPLY** card.

EXAMPLE: \_\_\_\_\_

```
#          kptable elgrps
JOINTPR_APPLY umba    pipegr1  pipegr2  pipegr3
```

### 3.36 JOINTPR\_DEFINE - Define pipeline joint properties

The joint property table is relevant for all scenarios where (SIMLA) has knowledge about kp-values. That is whenever a **COSURFPR** card, see Section 3.8, is present in the input file. The kp-tables defied by the **JOINTPR\_DEFINE** card will only be active if a **JOINTPR\_APPLY** card, see Section 3.35, is given.

The **JOINTPR\_DEFINE** command is defined by the following format:

**JOINTPR\_DEFINE NAME TYPE ....**

**NAME:** Name of joint table. To be refered in the **JOINTPR\_APPLY** card, see Section 3.35 to connect the table to one or more element groups.

**TYPE:** Joint type, which can have two values:

**PIPE :** Bending and torsional stiffness is calculated based on given radius and thickness of pipe. Convenient for simple pipes.

**GENERAL :** Bending and torsional stiffness is given in the following. Convenient for complex structures, as umbilicals.

For **PIPE** the format is as follows:

<b>JOINTPR_DEFINE NAME PIPE</b>
<b>KP1 KP2 RAD TH CDR CDT CMR CMT WD WS DO DOW FRAC LAB</b>
.....

**KP1:** KP start for this line of the table.

- KP2:** KP stop for this line of the table.
- RAD:** Structural radius i.e. the mean radius of the pipe wall (unit: L).
- TH:** Structural thickness, i.e. the wall thickness (unit: L).
- CDR:** Radial drag coefficient (unit: -).
- CDT:** Tangential drag coefficient (unit: -).
- CMR:** Radial added mass coefficient (unit: -).
- CMT:** Tangential added mass coefficient (unit: -).
- WD:** Dry mass (unit:  $ML^{-1}$ ).
- WS:** Dry mass - buoyancy mass (unit:  $ML^{-1}$ ).
- DO:** Outer diameter, Do (unit: L).
- DOW:** External wrapping outer diameter, Dow (unit: L).
- FRAC:** External wrapping fraction  $\eta(0 - 1)$ . Hence the diameter that will be applied to calculate drag and mass forces will be:  $D = (1 - \eta)D_o + \eta D_{ow}$
- LAB:** Label for the joints described in this line of the JOINT-table.

For **GENERAL** a few more parameters are needed:

```
JOINTPR_DEFINE NAME GENERAL
  KP1 KP2 RAD TH CDR CDT CMR CMT WD WS DO DOW FRAC EA
    EIY EIZ GIT LAB
```

...   ...   ...   .....

- KP1:** KP start for this line of the table.
- KP2:** KP stop for this line of the table.
- RAD:** Structural radius i.e. the mean radius of the pipe wall (unit: L).
- TH:** Structural thickness, i.e. the wall thickness (unit: L).
- CDR:** Radial drag coefficient (unit: -).
- CDT:** Tangential drag coefficient (unit: -).
- CMR:** Radial added mass coefficient (unit: -).
- CMT:** Tangential added mass coefficient (unit: -).
- WD:** Dry mass (unit:  $ML^{-1}$ ).
- WS:** Dry mass - buoyancy mass (unit:  $ML^{-1}$ ).
- DO:** Outer diameter, Do (unit: L).
- DOW:** External wrapping outer diameter, Dow (unit: L).
- FRAC:** External wrapping fraction  $\eta(0 - 1)$ . Hence the diameter that will be applied to calculate drag and mass forces will be:  $D = (1 - \eta)D_o + \eta D_{ow}$
- EA:** Axial stiffness (unit: F)
- EIY:** Bending stiffness about y axis (unit:  $FL^2$ )

**EIZ:** Bending stiffness about z axis (unit:  $FL^2$ )

**GIT:** Shear modulus (unit:  $FL^{-2}$ )

**LAB:** Label for the joints described in this line of the JOINT-table.

EXAMPLE:

---

```
#-----
#
# Joint property input
#           name type
JOINTPR_DEFINE umba PIPE
#
#   kp1    kp2    rad      th CDr Cdt CMr CMt    wd      ws    ODp    ODw rks lab
    1    94  0.525 0.0341 1.0 0.1 2.0 0.2 1.291 0.161 1.096 1.096 0.5 ''p1''
   95   111  0.525 0.0341 1.0 0.1 2.0 0.2 1.661 0.454 1.096 1.096 0.5 ''p2''
  112   769  0.525 0.0341 1.0 0.1 2.0 0.2 1.291 0.161 1.096 1.096 0.5 ''p2''
  770   786  0.525 0.0341 1.0 0.1 2.0 0.2 1.661 0.454 1.096 1.096 0.5 ''p2''
  787  1771  0.525 0.0341 1.0 0.1 2.0 0.2 1.291 0.161 1.096 1.096 0.5 ''p2''
#
#-----
```

---

### 3.37 MATERIAL - MATERIAL properties

The following material types are available in version 3.24.0 of the program:

**LINEAR :** Linear material properties for the elastic pipe elements. Relevant for PIPE31, PIPE34 and CABLE111 .

**ELASTOPLASTIC :** Elastoplastic strain-stress material behaviour with kinematic/isotropic hardening due to the elastoplastic pipe elements. Relevant for PIPE33 and PIPE39

**EPCURVE :** Elastoplastic material behaviour with kinematic/isotropic hardening.

**HYCURVE :** Hyperelastic (non-linear elastic) material behaviour.

**FRICTION :** Various friction models with user defined elastic stiffness.

**SEA :** Sea properties, for the SEA150 element type only.

**RESULTANT :** User defined material property based on describing the material curves in each direction for the COMPIPE42 element type.

**R\_CONTACT :** User defined material property based on describing the material curves in the x-, y- and z-directions and the x-rotation for the seabed contact element type CONT126. The purpose of the model is to enable modelling of x-moment induced by pure x-rotation with zero contribution from the y-displacement. As default, the x- and y-direction material curves are to be specified as displacements versus unit force curves. By specifying **USERDEFINED** at the end of the contact card, the x- and y-curves are to be specified as displacement versus force. Then

the x- and y-forces will be independent of the z-force. The element length is taken into account so the material curves are per unit length.

**CONTACT :** User defined material surface property based on describing the material curve in the x-, y- and z- directions for the contact element types. As default, the force x and y components are multiplied with the z-force reaction and a friction coefficient. This means that the x- and y-material curves are to be specified as displacements versus unit force curves. By specifying **USERDEFINED** at the end of the card, the x- and y- curves are to be specified as displacement versus force curves. Then the x- and y-forces will be independent of the z-force. For element type 125, 126, 130 and 152 the element length is taken into account, i.e. the diagrams is given per element length. For element types 124 and 170 the diagram need to be given as lumped properties.

**ISOCOMPRESSIVE :** Isotropic compressive material for element types 124, 130, 152 and 164.

**ISOKXYCONTACT :** Isotropic contact material with constant stick stiffness for element types 153 and 164.

**CLAY\_Y :** Elasto-plastic clay soil model for on-bottom stability analysis. The model is tailor-made for cyclic lateral response and accounts for penetration-dependent lateral resistance. Only available for the CONT126 element type.

**SAND\_Y :** Elasto-plastic sand soil model for on-bottom stability analysis. The model is tailor-made for cyclic lateral response and accounts for penetration-dependent lateral resistance. Only available for the CONT126 element type.

**BREAKOUT\_Y :** Elasto-plastic breakout model for undrained and drained soil.

**UNDRAIN\_Z :** Hyper-elastic model for undrained soil in vertical direction.

**DRAIN\_Z :** Hyper-elastic model for drained soil in vertical direction.

**HYPERELASTIC :** Hyperelastic strain-stress material behaviour (non-linear elastic).  
No relevant element types at the moment.

**LINSPRING :** User defined material property using linear stiffness in 6 DOFs. The stiffness must be defined in terms of lumped properties. Only available for element type 136.

**SPRING :** User defined spring property based on describing material curves for the 3 translation DOFs and the x-rotation DOF. Relevant for modelling of rock dumps. No coupling between the DOFs is allowed for. The material curves are described per unit length, i.e. the program calculates the element length and multiplies all forces with this length. Only available for element type 128.

**GENSEP :** User defined material property based on describing material curves in 6 DOFs. The material curves may be activated at an arbitrary load step in the different DOFs, and allow for various coupling between the DOFs relevant for pipe-in-pipe and friction modelling, see Sections 3.23.10 and 3.37.20. The material curves must be defined in terms of lumped properties. Only available for element

type 137.

**CONCRETE** : Concrete material, only for element type HSHEAR342.

Some of the parameters may be dummy, as the input is organized in the same manner for different materials. The user should specify a value of correct type, but the specified value will not be applied.

The format is as follows:

**MATERIAL MNAME MTYPE ...**

where

**MNAME**: Material name.

**MTYPE**: Material type as defined above.

The following parameters depend on the material type as specified in the following.

### 3.37.1 LINEAR (PIPE31, PIPE34)

**MATERIAL MNAME LINEAR POISS TALFA TECOND HEATC BETA EA EIY EIZ  
GIT EM GM**

**POISS**: Poisson's ratio (-)

**TALFA**: Temperature elongation coeff. (unit  $K^{-1}$ )

**TECOND**: Thermal conductivity (dummy) (unit:  $W/mC^\circ$ )

**HEATC**: Heat capacity (dummy) (unit: unit:  $J/kgC^\circ$ )

**BETA**: Axial-torsion coupling parameter  $\beta$  (dummy) (unit:L)

**EA**: Axial stiffness (unit: F)

**EIY**: Bending stiffness about y axis (unit:  $FL^2$ )

**EIZ**: Bending stiffness about z axis (unit:  $FL^2$ )

**GIT**: Torsion stiffness (unit:  $FL^2$ )

**EM**: Young's modulus (unit:  $FL^{-2}$ )

**GM**: Shear modulus (unit:  $FL^{-2}$ )

EXAMPLE:

---

```
#      name      type    pois  talfa   tecond  heatc  beta  ea     eiy     eiz   git   em   gm
MATERIAL pipemat1 linear  0.3  1.1e-5 50       800    0     1.4e4 7.6e2 7.6e2 5.8e2 2e5 8e4
```

---

### 3.37.2 ELASTOPLASTIC (PIPE33, PIPE39)

**MATERIAL MNAME ELASTOPLASTIC IHARD POISS RHO0 TALFA TECOND**

**HEATC EPS SIGMA**

... ...

**IHARD:** Hardening parameter 0.0-1.0 or 2.0-3.0; 0.0 = isotropic hardening; 1.0 = kinematic hardening with accumulated plastic strains. 2.0 and 3.0 is isotropic and kinematic hardening respectively, but with no accumulation of plastic strains for reverse axial loading. Value can also be between 2.0-3.0, corresponding to values 0.0-1.0 in terms of hardening.

**POISS:** Poisson's ratio (-)

**RHO0:** Density (dummy) (unit:  $\text{ML}^{-3}$ )

**TALFA:** Temperature elongation coeff. (unit  $K^{-1}$ )

**TECOND:** Thermal conductivity (dummy) (unit:  $\text{EL}^{-1}\text{TE}^{-1}$ )

**HEATC:** Heat capacity (dummy) (unit:  $\text{E}^{-1}\text{M}^{-1}\text{TE}^{-1}$ )

**EPS:** Strain always starting at 0

**SIGMA:** Stress (unit:  $\text{FL}^{-2}$ ) always starting with 0 and then only positive paired values of strain and stress.

Any number of pairs of **EPS** and **SIGMA** can be given, and linear interpolation is applied between the supplied data. The first non-zero strain-stress pair is taken as the proportionality limit. The hardening parameter (**IHARD**) controls the behaviour of the yield surface. Kinematic hardening (i.e. elastic range is constant equal to  $2\sigma_y$ ) is specified by a hardening parameter equal to 1.0. Isotropic hardening (i.e. elastic range is extended as function of strain experienced by the material) is specified by a hardening parameter equal to 0.0. An illustration for a simply supported beam exposed to repeated rotations in one end is shown below in Fig. 3.33. Note that the first rotation has smaller amplitude than the following.

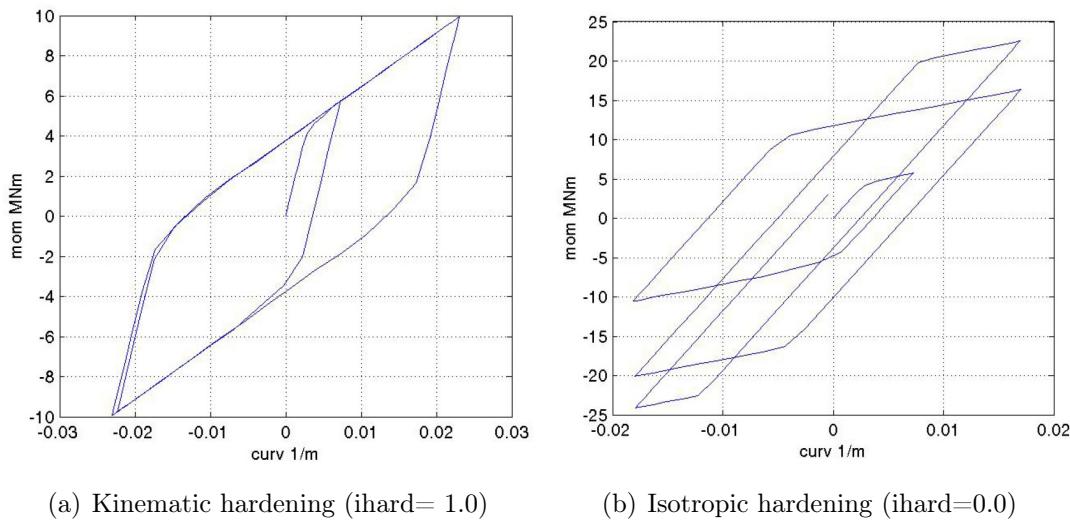


Figure 3.33: Illustration of material hardening effect.

EXAMPLE:

```

#
#      name      type          ihard poiss   ro    talpha   tecond heatc
MATERIAL pipemat2 elastoplastic 1.0      0.3    7850 1.17e-5 50      800
#
#      eps        sigma ..1,n
0            0
1.691E-03  3.50E+02
2.000E-03  3.90E+02
2.500E-03  4.15E+02
3.000E-03  4.27E+02
3.500E-03  4.35E+02
4.000E-03  4.41E+02
5.010E-03  4.50E+02
6.000E-03  4.57E+02

```

---

### 3.37.3 EPCURVE

<b>MATERIAL MNAME EPCURVE IHARD EPS SIGMA</b>
...    ...

**IHARD:** Hardening parameter; **0** = isotropic hardening; **1** = kinematic hardening.

**EPS:** Strain, displacement, torsion or curvature depending on which quantity to be described, starting at 0.

**SIGMA:** Stress, unit resultant or resultant values depending on which quantity to be described. The material curve should always start at 0 and then contain only

positive paired values.

Between specified points, linear interpolation is applied. Values outside the specified range are extrapolated.

Table 3.2 gives an overview of which dimensions to apply for **EPS** and **SIGMA**, depending on the setting in which the **EPCURVE** is applied.

Table 3.2: Units for **EPS** and **SIGMA**.

Element type	Material type	Option	DOF	EPS-unit	SIGMA-unit
COMPIPE42	RESULTANT		axial	- (strain)	F (force)
	RESULTANT		torsion	$L^{-1}$ (torsion)	FL (torsion moment)
	RESULTANT		bending	$L^{-1}$ (curvature)	FL (bending moment)
CABLE111			axial	- (strain)	F (force)
CONT124	CONTACT		z-dir	L (displacement)	F (force)
	ISOCONTACT				
	CONTACT	COULOMB (default)	x- and y-dir	L (displacement)	- (unit force)
CONT125	CONTACT	USERDEFINED	x- and y-dir	L (displacement)	F (force)
	ISOCONTACT		x- and y-dir	L (displacement)	- (unit force)
	CONTACT		z-dir	L (displacement)	$FL^{-1}$ (force pr. length)
CONT126	CONTACT		x- and y-dir	L (displacement)	$L^{-1}$ (unit force pr. length)
	R_CONTACT		z-dir	L (displacement)	$FL^{-1}$ (force pr. length)
	CONTACT	COULOMB (default)	x- and y-dir	L (displacement)	$FL^{-1}$ (unit force pr. length)
CONT128	CONTACT	USERDEFINED	x- and y-dir	L (displacement)	$FL^{-1}$ (force pr.length)
	R_CONTACT		x-rot	Rad (rotation)	$L^{-1}$ (unit y-force pr. length)
	R_CONTACT	COULOMB (default)	x-rot	Rad (rotation)	F (force pr. length)
CONT130	SPRING		z-dir	L (displacement)	$FL^{-1}$ (force pr.length)
	SPRING		x- and y-dir	L (displacement)	$FL^{-1}$ (force pr.length)
	SPRING		x-rot	Rad (rotation)	F (moment pr.length)
CONT130	CONTACT		z-dir	L (displacement)	$FL^{-1}$ (force pr. length)
	ISOCONTACT				
	ISOKXYCONTACT				
SPRING137	CONTACT	COULOMB (default)	x- and y-dir	L (displacement)	$L^{-1}$ (unit force pr. length)
	ISOCONTACT				
	ISOKXYCONTACT				
CONT152	CONTACT	USERDEFINED	x- and y-dir	L (displacement)	$FL^{-1}$ (force pr. length)
	GENSESPRING		x-, y-, and z- dir	L (displacement)	F (force)
	GENSESPRING		x-, y- and z-rot	Rad (rotation)	FL (moment)
CONT152	ISOCONTACT		z-dir	L (displacement)	$FL^{-1}$ (force pr. length)
	ISOKXYCONTACT				

*Continued on next page*

Element type	Material type	Option	DOF	EPS-unit	SIGMA-unit
	CONTACT ISOCONTACT ISOKXYCONTACT	COULOMB (default)	x- and y-dir	L (displacement)	$L^{-1}$ (unit force pr. length)
	CONTACT	USERDEFINED	x- and y-dir	L (displacement)	$FL^{-1}$ (force pr. length)
CONT153	ISOKXYCONTACT		z-dir	L (displacement)	F (force)
CONT164	CONTACT ISOCONTACT ISOKXYCONTACT		z-dir	L (displacement)	F (force)
	CONTACT ISOCONTACT ISOKXYCONTACT	COULOMB (default)	x- and y-dir	L (displacement)	- (unit force)
	CONTACT	USERDEFINED	x- and y-dir	L (displacement)	F (force)
CONT170	CONTACT		z-dir	L (displacement)	F (force)
	CONTACT	COULOMB (default)	x- and y-dir	L (displacement)	- (unit force)
	CONTACT	USERDEFINED	x- and y-dir	L (displacement)	F (force)
HSHEAR342	-	-	axial rela- tive (shear)	L (relative displacement)	$FL^{-2}$ (force pr. area)

## EXAMPLE:

---

```
#      name   type   ihard   eps   sig
MATERIAL soilx epcurve   1     0.000  0.0
                           0.005  1.0
                           1.000  1.0
```

---

### 3.37.4 HYCURVE

**MATERIAL MNAME HYCURVE EPS SIGMA**

... ...

**EPS:** Strain, displacement, torsion or curvature depending on which quantity to be described

**SIGMA:** Stress or resultant values depending on which quantity to be described

Between specified points, linear interpolation is applied. Values outside the specified range are extrapolated.

To help in convergence of the CONT152 element, a smooth stiffness transition has been implemented. The stiffness goes from zero to the full value over a length which stretches 0.001 times the length from the origin to the **EPS**-coordinate of the first material point on the negative side. This modification only affects the stiffness, and not the force. Consequently, the final equilibrium state will not be influenced by the

stiffness modification. For CONT152, the first material point on the negative side is suggested to be less than 1 percent of the last material point as can be seen in the example below.

Refer to Table 3.2 for an overview of which dimensions to apply for **EPS** and **SIGMA**, depending on the setting in which the **HYCURVE** is applied.

EXAMPLE:

---

```
#      name   type     eps      sig
MATERIAL soilx  hycurve -1000.000 -10.0
                  1000.000 10.0
#      name   type     eps      sig
MATERIAL bm_cont_z  hycurve -1000.0 -100000.0
                  -10.0    -1000.0
                  0.0     0.0
                  1000.0 100000.0
```

---

### 3.37.5 ISOHYCURVE

<b>MATERIAL MNAME ISOHYCURVE EPS SIGMA</b>
...

**EPS**: Strain, displacement, torsion or curvature depending on which quantity to be described

**SIGMA**: Stress or resultant values depending on which quantity to be described

**ISOHYCURVE** is used for applications which need symmetry about origo. The first eps sigma pair must thus be in origo.

Refer to Table 3.2 for an overview of which dimensions to apply for **EPS** and **SIGMA**, depending on the setting in which the **ISOHYCURVE** is applied.

EXAMPLE:

---

```
#      name   type     eps      sig
MATERIAL soilx  isohycurve 0.0     0.0
                  1000.000 10.0
```

---

### 3.37.6 SEA

<b>MATERIAL MNAME MTYPE SEADEN</b>
------------------------------------

**SEADEN:** Sea density (unit:  $\text{ML}^{-3}$ )

EXAMPLE: \_\_\_\_\_

#	name	type	density
MATERIAL	seamat	sea	1000e-6

### 3.37.7 RESULTANT (COMPIPE42)

**MATERIAL MNAME RESULTANT POISS RHO TALFA TECOND HEATC BETA  
FQX FFX FKY FKZ FF REFP IPDER EPDER REFT TDER AXNAME TORNAME  
BENDNAME**

**POISS:** Apparent Poisson's number  $\nu_a$  (-)

**RHO:** Density (Dummy) (unit:  $\text{ML}^{-3}$ )

**TALFA:** Temperature elongation coeff.  $\alpha$  (-)  $\varepsilon = \alpha\Delta T$

**TECOND:** Thermal conductivity (Dummy) (unit:  $\text{EL}^{-1}\text{TE}^{-1}$ )

**HEATC:** Heat capacity (Dummy) (unit:  $\text{E}^{-1}\text{M}^{-1}\text{TE}^{-1}$ )

**BETA:** Axial-torsion coupling parameter  $\beta$  (unit: L)

**FQX:** Axial stress factor  $f_{qx}$  (-) Axial stress = Axial force  $\times$  factor

**FFX:** Torsion stress factor  $f_{kx}$  (-) Shear stress = Torsion  $\times$  factor

**FKY:** y-curvature elastic stress factor  $f_{ky}$  (-) Bending stress = y-curvature  $\times$  factor

**FKZ:** z-curvature elastic stress factor  $f_{kz}$  (-) Bending stress = z-curvature  $\times$  factor

**FF:** friction stress conversion factor (-) Bending stress = Bending moment  $\times$  factor

**IPDER:** Currently dummy float number

**EPDER:** Currently dummy float number

**REFT:** Currently dummy float number

**TDER:** Currently dummy float number

**AXNAME:** Material name defining axial force vs. axial strain (epcurve or hycurve)

**TORNAME:** Material name defining torque vs. torsion (epcurve or hycurve)

**BENDNAME:** Material name defining bending moment vs. curvature (epcurve or isohycurve)

The apparent Poisson's number  $\nu_a$  may be calculated for a load case involving axial and radial loads. Flexible pipe manufacturers typically provide tables in terms of axial strain  $\epsilon_x$  versus internal pressure  $p_i$ . The following formula can then be applied to estimate  $\nu_a$ ,

$$\epsilon_x = \frac{(1 - 2\nu_a) p_i \pi R_i^2}{EA} \quad (3.37)$$

where  $EA$  is the axial stiffness and  $R_i$  is the internal radius.

For a non-zero axial-torsion coupling parameter  $\beta$ , the **AXNAME** and **TORNAME** material curves must define a constant axial stiffness  $EA$  and a constant torsion stiffness  $GJ$  for all strain ranges, and the material curves must intersect the origin. The axial force  $F_x$  and the torque  $M_x$  for a given axial strain  $\epsilon_x$  and torsion  $\kappa_x$  are then calculated as follows,

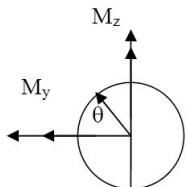
$$\begin{bmatrix} F_x \\ M_x \end{bmatrix} = \begin{bmatrix} EA & \beta EA \\ \beta EA & GJ + \beta^2 EA \end{bmatrix} \begin{bmatrix} \epsilon_x \\ \kappa_x \end{bmatrix} \quad (3.38)$$

where it is noted that the total torsion stiffness is equal to:  $GJ + \beta^2 EA$ . The user-specified torsion stiffness  $GJ$  defined by the **TORNAME** material curve shall therefore not include the coupling contribution  $\beta^2 EA$ .

Equation (3.38) may be applied to estimate the  $\beta$ -parameter based on laboratory testing or by using an axi-symmetric load analysis software.

If the **BENDNAME** material is assigned the **EPCURVE** material type, the hardening for the elasto-plastic bending response will be kinematic regardless of other user-defined input data. In that case, the hardening parameter **IHARD** in Section 3.37.3 is regarded as a dummy parameter.

*Note:*



$$\sigma_{xx} = \epsilon_x f_{ax} + \kappa_y f_{ky} \cos \theta + \kappa_z f_{kz} \sin \theta + M_y f_f \sin \theta + M_z f_f \cos \theta$$

and

$$\sigma_{xy} = \kappa_x f_{kx}$$

EXAMPLE: \_\_\_\_\_

```
#      mname    mtype      poiss   rho   talfa   tecond   heatc   beta   fqx      fkx
MATERIAL pipemat resultant  0.2     0.0   1.17e-5  0.0      0.0     0.1   2.07e5  27152
#
#      fky     fkz     ff   refp   ipder   epder   reft   tder   axname   torname   bendname
#                  7025    7025   81     0.0     0.0     0.0     0.0     0.0   axmat    tomat    bendmat
#
#      name     type      eps   force
MATERIAL axmat   hycurve   -1000  -0.5e6
#                                         0.0     0.0
#                                         1000   1.5e6
#
#      name     type      tors   torque
MATERIAL tomat   hycurve   -1000  -0.5e5
#                                         0.0     0.0
```

```

          1000    2.0e5
#
#      name     type     ihard     curv     moment
MATERIAL  bendmat  epcurve   1        0.0       0.0
                      0.001     120.0
                     10.00    130000.0

```

---

### 3.37.8 CONTACT (CONT124-126, 130, 152, 164, 170)

**MATERIAL MNAME CONTACT MUX MUY XNAME YNAME ZNAME**  
**[ [XYCHOICE] [ NX NY PX\_1 PXNAME\_1 ... PX\_NX PXNAME\_NX PY\_1**  
**PYNAME\_1 ... PY\_NY PYNAME\_NY ] ]**

**MUX**: Friction coefficient in local *x*-direction (unit: -).

**MUY**: Friction coefficient in local *y*-direction. Negative value will provide isotropic friction based on **MUX** for element type CONT126 (unit: -).

**XNAME**: Local *x*-direction material curve name.

**YNAME**: Local *y*-direction material curve name.

**ZNAME**: Local *z*-direction material curve name.

**XYCHOICE**: Enables overruling of the default Coulomb friction model with user-defined characteristics for the *x*- and *y*-directions. Optional. Allowable values:

**COULOMB** : The **XNAME** and **PXNAME\_I** curves are scaled by the product of the local *z*-direction force and **MUX**, while the **YNAME** and **PYNAME\_I** curves are scaled by the product of the local *z*-direction force and **MUY**. This is the default behaviour.

**USERDEFINED** : The **XNAME**, **YNAME**, **PXNAME\_I** and **PYNAME\_I** curves as defined by the user are applied.

**COULOMB-COULOMB-COULOMB** : Coulomb friction is applied for all material curves in the local *x*- and *y*-directions. Equivalent to specifying **COULOMB** only.

**COULOMB-USERDEFINED-USERDEFINED** : Coulomb friction for **XNAME** and **YNAME**, and userdefined characteristics for **PXNAME\_I** and **PYNAME\_I**.

**COULOMB-USERDEFINED-COULOMB** : Coulomb friction for **XNAME**, **YNAME** and **PYNAME\_I**, and userdefined characteristics for **PXNAME\_I**.

**COULOMB-COULOMB-USERDEFINED** : Coulomb friction for **XNAME**, **YNAME** and **PXNAME\_I**, and userdefined characteristics for **PYNAME\_I**.

**USERDEFINED-USERDEFINED-USERDEFINED** : The curves as defined by the user are applied for all material curves in the local *x*- and *y*-directions.

Equivalent to specifying **USERDEFINED** only.

**USERDEFINED-COULOMB-USERDEFINED** : Coulomb friction for **PXNAME\_I**, and userdefined characteristics for **XNAME**, **YNAME** and **PYNAME\_I**.

**USERDEFINED-USERDEFINED-COULOMB** : Coulomb friction for **PYNAME\_I**, and userdefined characteristics for **XNAME**, **YNAME** and **PXNAME\_I**.

**USERDEFINED-COULOMB-COULOMB** : Coulomb friction for **PXNAME\_I** and **PYNAME\_I**, and userdefined characteristics for **XNAME** and **YNAME**.

**NX**: Number of penetration-dependent penetration-dependent material curves in local *x*-direction. Optional.

**NY**: Number of penetration-dependent penetration-dependent material curves in local *y*-direction. Optional.

**PX\_1**: Penetration value at which the first penetration-dependent material curve in local *x*-direction is valid. Not specified for **NX=0**. Optional. (unit: L).

**PXNAME\_1**: Local *x*-direction first penetration-dependent material curve name. The curve must be hyper-elastic. Not specified for **NX=0**. Optional.

**PX\_NX**: Penetration value at which the last penetration-dependent material curve in local *x*-direction is valid. Not specified for **NX=0,1**. Optional. (unit: L).

**PXNAME\_NX**: Local *x*-direction last penetration-dependent material curve name. The curve must be hyper-elastic. Not specified for **NX=0,1**. Optional.

**PY\_1**: Penetration value at which the first penetration-dependent material curve in local *y*-direction is valid. Not specified for **NY=0**. Optional. (unit: L).

**PYNAME\_1**: Local *y*-direction first penetration-dependent material curve name. The curve must be hyper-elastic. Not specified for **NY=0**. Optional.

**PY\_NY**: Penetration value at which the last penetration-dependent material curve in local *y*-direction is valid. Not specified for **NY=0,1**. Optional. (unit: L).

**PYNAME\_NY**: Local *y*-direction last penetration-dependent material curve name. The curve must be hyper-elastic. Not specified for **NY=0,1**. Optional.

For other element types than CONT126, the only optional parameter that will be applied is **XYCHOICE** with value equal to either **COULOMB** or **USERDEFINED** as shown in the below example.

EXAMPLE:

---

#	name	type	mux	muy	xname	yname	zname	xychoice
MATERIAL	bellm1	contact	0.5	1.0	matx	maty	matz1	coulomb
MATERIAL	body2	contact	0.5	1.0	matx	maty	matz2	
MATERIAL	roller1	contact	0.3	0.3	matx	maty	matz3	userdefined
MATERIAL	roller2	contact	0.3	0.3	matx	maty	matz4	userdefined
MATERIAL	roller3	contact	0.3	0.3	matx	maty	matz5	

---

For the CONT126 element type, the optional parameters must either consist of only **XYCHOICE** or all of the optional parameters as shown in the examples below. Regarding **XYCHOICE**, the first word governs the **XNAME** and **YNAME** curves, the second word governs the **PXNAME\_I** curves and the third word governs the **PYNAME\_I** curves. This is convenient when describing soil resistance where one part is related to **COULOMB** and the second part is related to earth pressure resistance better described by a **USERDEFINED** characteristic depending on the penetration. For such cases it is recommended to use two curves as illustrated in Fig. 3.34.

An arbitrary number of penetration-dependent curves **PXNAME\_I** and **PYNAME\_I** may be given and SIMLA will interpolate the force and the stiffness based on the actual penetration. Zero force is always assumed at zero penetration. Extrapolation of the force and stiffness is done if the actual penetration exceeds the largest user-defined penetration value. Hence, very large force and stiffness magnitudes may result if the actual penetration exceeds the largest user-defined penetration value significantly. The sign of the user-defined penetration values are of no importance because SIMLA applies the absolute value in the computations. The actual penetration includes the contributions from the soil elastic penetration, the element initial displacement defined by the **INISTR** card in Section 3.34, the KP-based soil embedment defined by the **ELPROP** card in Section 3.23.11 and the soil embedment modelled according to Section 3.8.3.

---

#### EXAMPLE:

---

```
#      name   type   mux   muy   xname  yname  zname  xychoice
MATERIAL soil1  contact 0.5  1.0  soilx  soily  soilz  coulomb-userdefined-userdefined
#
#  Nx   Ny   px1   pxname1   py1   pyname1   py2   pyname2
    1     2   0.1   soilx_1   0.1   soily_1   0.2   soil_y2
```

---



---

#### EXAMPLE:

---

```
#      name   type   mux   muy   xname  yname  zname  xychoice
MATERIAL soil1  contact 0.5  1.0  soilx  soily  soilz  userdefined
```

---

If the optional parameter **XYCHOICE=USERDEFINED** is omitted, the material curves in the *x*- and *y*-directions are to be defined as a consecutive number of points defining displacement versus unit force per length or unit force. This is because SIMLA will scale the unit force value by  $\mu F_z$  or  $\mu F_z L$ , depending on the element type. Here,  $\mu$  is the friction factor,  $F_z$  is the *z*-direction contact force (unit: F), and  $L$  is the length of the pipe element in contact.

For **XYCHOICE=USERDEFINED**, the  $x$ - and  $y$ -direction curves are to be defined as a consecutive number of points defining displacement versus force per unit length or force, i.e. no scaling with  $\mu F_z$  is applied.

If the **YNAME** curve is assigned the **CLAY\_Y** and **SAND\_Y** models in Sections 3.37.12 and 3.37.13, the  $y$ -direction force will not be scaled with  $\mu F_z$  regardless of the value specified for **XYCHOICE**.

For the standard case, the material curves **XNAME**, **YNAME** and **ZNAME** need to be defined using the **HYCURVE** or **EPCURVE** options. The latter option should be applied in modelling of friction, otherwise the friction force reversal will not be correctly modelled. The material curve units depend on the element type, see Table 3.2.

The contact point  $y$ -displacement  $u_y^{cp}$  contains contributions from both pure  $y$ -translation  $u_y$  and  $y$ -translation induced by  $x$ -rotation  $\theta_x$  according to,

$$u_y^{cp} = u_y + r\theta_x \quad (3.39)$$

where  $r$  is the pipe external radius.

The local  $y$ -force due to pipe-soil interaction is calculated as,

$$F_y(u_y^{cp}) = \mu_y \bar{f}_y(u_y^{cp}) F_z L \quad \text{if } \text{XYCHOICE}=\text{COULOMB} \quad (3.40)$$

$$F_y(u_y^{cp}) = f_y(u_y^{cp}) L \quad \text{if } \text{XYCHOICE}=\text{USERDEFINED} \quad (3.41)$$

where  $\mu_y=\text{MUY}$ ,  $\bar{f}_y$  is the dimensionless force per unit length defined by **YNAME**,  $F_z$  is the force in the local  $z$ -direction (unit: F),  $L$  is the length of the pipe element in contact,  $f_y$  is the force per unit length defined by **YNAME**, and  $u_y^{cp}$  is defined in Eq. (3.39). The local  $x$ -moment due to pipe-soil interaction is calculated as,

$$M_x = rF_y(u_y^{cp}) \quad (3.42)$$

which means that  $x$ -moment will be induced for rotation about the  $x$ -axis and  $y$ -displacement such that rolling contact interaction will be correctly handled.

Note that for the element type CONT126, the  $x$ -moment will be set to zero and the second term in Eq. (3.39) will be removed if the option **IGAP=2** is applied for the **CONTINT** card in Section 3.5.

The local  $x$ -direction for seabed contact elements is directed along the current  $x$ -axis of the element system, and will undergo the same rotations as the attached pipe node during the analysis. Hence, the element initial  $x$ -axis as defined by the **ELORIENT** card must be directed along the pipe element initial  $x$ -axis in order to get the local  $x$ - and  $y$ -force directed along and transverse to the pipeline, respectively, and to get the local  $x$ -moment acting about the pipeline  $x$ -axis. The local  $z$ -direction is adjusted to

coincide with the radial direction of the pipe cross-section provided that the element initial  $x$ -axis and the pipe element initial  $x$ -axis coincide, and the local  $y$ -direction is computed by the cross product of local  $z$ -direction unit vector and the local  $x$ -direction unit vector.

For other contact element types, the local  $x$ -direction coincides with the current pipe axial direction, and the local  $y$ -direction is determined by the cross product of the contact normal direction unit vector and the local  $x$ -direction unit vector.

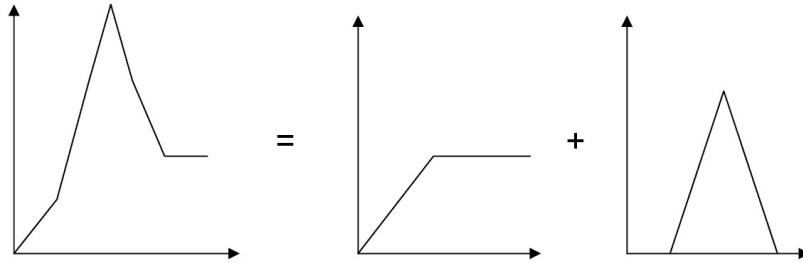


Figure 3.34: Material curve combination.

### 3.37.9 ISOCONTACT (CONT130, CONT152, CONT164)

**MATERIAL MNAME ISOCONTACT MUXY XYNAME ZNAME**

**MUXY:** Friction coefficient in local xy-direction

**XYNAME:** xy-plane material curve name

**ZNAME:** z-direction material curve name

EXAMPLE:

---

```
#      name      type      rmyx      xyname    zname
MATERIAL rollmat  isocontact  1.0      rollxy   rollz
#
#      name      type      ihard     eps      sig
MATERIAL rollxy epcurve    1       0.0     0.0
                           1.0    2.5e7
```

---

The **ZNAME** material curve can be defined by the **EPCURVE** or **HYCURVE** options, while only the **EPCURVE** option is allowed for the **XYNAME** material curve.

The **XYNAME** material is to be defined as a consecutive number of points defining displacement versus unit force per length. This is because SIMLA will scale the unit force value with  $\mu N$ , where  $\mu$  is the friction factor and  $N$  is the normal direction force.

### 3.37.10 ISOKXYCONTACT (CONT153, CONT164)

**MATERIAL MNAME ISOKXYCONTACT MUXY KSTICK ZNAME**

**MUXY:** Friction coefficient in local xy-direction (unit: -).

**KSTICK:** Stick stiffness in local xy-direction (unit:  $FL^{-1}$ ) for element types **CONT153** and **CONT164**,  $FL^{-2}$  for element types **CONT130** and **CONT152**

**ZNAME:** z-direction material curve name

**EXAMPLE:**

---

```
#      name    type     rmyx     kstick   zname
MATERIAL rollmat  isocontact  1.0      2.5e7    rollz
```

---

The **ZNAME** material curve must be defined by the **HYCURVE** option for **CONT153**, while both **EPCURVE** and **HYCURVE** can be applied for other elements.

A standard Coulomb friction model is applied where the change of friction force in the stick state is equal to the product of the tangential displacement increment and **KSTICK**. As opposed to the **ISOCONTACT** option, the stick stiffness is constant and independent of the normal direction contact force. This may enhance the convergence properties in situations where large normal direction contact forces occur, such as for contact between hard materials or for large contact force spikes at initial contact.

### 3.37.11 R\_CONTACT (CONT126)

**MATERIAL MNAME R\_CONTACT MUX MUY MUTX XNAME YNAME ZNAME TXNAME [ XYCHOICE ] [ NX NY PX\_1 PXNAME\_1 ... PX\_NX PXNAME\_NX PY\_1 PYNAME\_1 ... PY\_NY PYNAME\_NY ]**

**MUX:** Friction coefficient in local *x*-direction (unit: -).

**MUY:** Friction coefficient in local *y*-direction. Negative value will provide isotropic friction based on **MUX** (unit: -).

**MUTX:** Friction coefficient in local *y*-direction due to *x*-rotation (unit: -).

**XNAME:** Local *x*-direction material curve name.

**YNAME:** Local *y*-direction material curve name.

**ZNAME:** Local *z*-direction material curve name.

**TXNAME:** Material curve defining local *x*-rotation versus the local *y*-direction force applied for calculating the *x*-moment.

**XYCHOICE**: Enables overruling of the default Coulomb friction model with user-defined characteristics for the local  $x$ -direction, the local  $y$ -direction and the local  $x$ -rotation. Optional. Allowable values:

**COULOMB** : The **XNAME** and **PXNAME\_I** curves are scaled by the product of the local  $z$ -direction force and **MUX**, while the **YNAME** and **PYNAME\_I** curves are scaled by the product of the local  $z$ -direction force and **MUY**. The **TXNAME** curve is scaled by the product of the local  $z$ -direction force, the friction coefficient **MUTX** and the pipe external radius. This is the default behaviour.

**USERDEFINED** : The **XNAME**, **YNAME**, **TXNAME**, **PXNAME\_I** and **PYNAME\_I** curves as defined by the user are applied.

**COULOMB-COULOMB-COULOMB** : Coulomb friction is applied for all material curves for the local  $x$ - and  $y$ -directions and the local  $x$ -rotation. Equivalent to specifying **COULOMB** only.

**COULOMB-USERDEFINED-USERDEFINED** : Coulomb friction for **XNAME**, **YNAME** and **TXNAME**, and userdefined characteristics for **PXNAME\_I** and **PYNAME\_I**.

**COULOMB-USERDEFINED-COULOMB** : Coulomb friction for **XNAME**, **YNAME**, **TXNAME** and **PYNAME\_I**, and userdefined characteristics for **PXNAME\_I**.

**COULOMB-COULOMB-USERDEFINED** : Coulomb friction for **XNAME**, **YNAME**, **TXNAME** and **PXNAME\_I**, and userdefined characteristics for **PYNAME\_I**.

**USERDEFINED-USERDEFINED-USERDEFINED** : The curves as defined by the user are applied for all material curves for the local  $x$ -direction, the local  $y$ -direction and the local  $x$ -rotation. Equivalent to specifying **USERDEFINED** only.

**USERDEFINED-COULOMB-USERDEFINED** : Coulomb friction for **PXNAME\_I**, and userdefined characteristics for **XNAME**, **YNAME**, **TXNAME** and **PYNAME\_I**.

**USERDEFINED-USERDEFINED-COULOMB** : Coulomb friction for **PYNAME\_I**, and userdefined characteristics for **XNAME**, **YNAME**, **TXNAME** and **PXNAME\_I**.

**USERDEFINED-COULOMB-COULOMB** : Coulomb friction for **PXNAME\_I** and **PYNAME\_I**, and userdefined characteristics for **XNAME**, **YNAME** and **TXNAME**.

**NX**: Number of penetration-dependent material curves in local  $x$ -direction. Optional.

**NY**: Number of penetration-dependent material curves in local  $y$ -direction. Optional.

**PX\_1**: Penetration value at which the first penetration-dependent material curve in local  $x$ -direction is valid. Positive for penetration into the seabed. Not specified for **NX=0**. Optional. (unit: L).

**PXNAME\_1:** Local  $x$ -direction first penetration-dependent material curve name. The curve must be hyper-elastic. Not specified for  $\text{NX}=0$ . Optional.

**PX\_NX:** Penetration value at which the last penetration-dependent material curve in local  $x$ -direction is valid. Positive for penetration into the seabed. Not specified for  $\text{NX}=0,1$ . Optional. (unit: L).

**PXNAME\_NX:** Local  $x$ -direction last penetration-dependent material curve name. The curve must be hyper-elastic. Not specified for  $\text{NX}=0,1$ . Optional.

**PY\_1:** Penetration value at which the first penetration-dependent material curve in local  $y$ -direction is valid. Positive for penetration into the seabed. Not specified for  $\text{NY}=0$ . Optional. (unit: L).

**PYNAME\_1:** Local  $y$ -direction first penetration-dependent material curve name. The curve must be hyper-elastic. Not specified for  $\text{NY}=0$ . Optional.

**PY\_NY:** Penetration value at which the last penetration-dependent material curve in local  $y$ -direction is valid. Positive for penetration into the seabed. Not specified for  $\text{NY}=0,1$ . Optional. (unit: L).

**PYNAME\_NY:** Local  $y$ -direction last penetration-dependent material curve name. The curve must be hyper-elastic. Not specified for  $\text{NY}=0,1$ . Optional.

The purpose of **R\_CONTACT** is to remove the  $x$ -moment dependency on the  $y$ -displacement. This is done by introducing the following modifications as compared to the **CONTACT** material type in Section 3.37.8:

- A separate curve **TXNAME** defines the  $y$ -direction force applied for calculating the  $x$ -moment.
- The local  $x$ -rotation is applied as the only displacement variable for the  $x$ -moment. The curve **TXNAME** is therefore defined in terms of the local  $x$ -rotation as the argument.
- The local  $x$ -rotation does not contribute to contact point displacement  $u_y^{cp}$  in the local  $y$ -direction. This means that the second term on the right-hand side of Eq. (3.39) is removed.

The above modifications enable modelling of pipe-soil skin friction effects in situations with non-zero  $x$ -rotation and zero  $y$ -displacement. For instance, consider a pipe that is penetrated into the seabed and subjected to a  $x$ -rotation without displacing in the  $y$ -direction. Here, the skin friction effect will induce a  $x$ -moment and zero force in the local  $y$ -direction, with non-zero  $x$ -rotation and zero  $y$ -displacement. This effect is not possible to model with the **CONTACT** model in Section 3.37.8.

The **R\_CONTACT** model gives zero  $x$ -moment for a case with zero  $x$ -rotation and non-zero  $y$ -displacement, however, the  $y$ -force will still be correctly modelled by the

**YNAME** curve. The model shall therefore not be used for modelling of the  $x$ -moment for a pipe that displaces in the  $y$ -direction. In that case the **CONTACT** model in Section 3.37.8 shall be applied.

The local  $y$ -force due to pipe-soil interaction is calculated as,

$$F_y = \mu_y \bar{f}_y (u_y) F_z L \quad \text{if } \mathbf{XYCHOICE}=\mathbf{COULOMB} \quad (3.43)$$

$$F_y = f_y (u_y) L \quad \text{if } \mathbf{XYCHOICE}=\mathbf{USERDEFINED} \quad (3.44)$$

where  $\mu_y=\mathbf{MUY}$ ,  $\bar{f}_y$  is the dimensionless force per unit length defined by **YNAME**,  $F_z$  is the force in the local  $z$ -direction (unit: F),  $L$  is the length of the pipe element in contact,  $f_y$  is the force per unit length defined by **YNAME**, and  $u_y$  is the local  $y$ -displacement of the pipe node. Note that the  $y$ -displacement consists only of pure translation, i.e. the  $x$ -rotation coupling term in Eq. (3.39) is not included. The  $x$ -moment only depends on the  $x$ -rotation  $\theta_x$  and is calculated as follows,

$$M_x = \mu_{tx} \bar{f}_{ty} (\theta_x) F_z r L \quad \text{if } \mathbf{XYCHOICE}=\mathbf{COULOMB} \quad (3.45)$$

$$M_x = f_{ty} (\theta_x) r L \quad \text{if } \mathbf{XYCHOICE}=\mathbf{USERDEFINED} \quad (3.46)$$

where  $\mu_{tx}=\mathbf{MUTX}$ ,  $\bar{f}_{ty}$  is the dimensionless  $y$ -force per unit length defined by **TXNAME**,  $F_z$  is the force in the local  $z$ -direction (unit: F),  $r$  is the pipe external radius,  $L$  is the length of the pipe element in contact, and  $f_{ty}$  is the  $y$ -force per unit length defined by **TXNAME**. Note that the  $x$ -moment will be set to zero if the option **IGAP=2** is applied for the **CONTINT** card in Section 3.5.

The optional parameters must either consist of only **XYCHOICE** or all of the optional parameters as shown in the examples below. Regarding **XYCHOICE**, the first word governs the **XNAME**, **YNAME** and **TXNAME** curves, the second word governs the **PXNAME\_I** curves and the third word governs the **PYNAME\_I** curves. This is convenient when describing soil resistance where one part is related to **COULOMB** and the second part is related to earth pressure resistance better described by a **USERDEFINED** characteristic depending on the penetration. For such cases it is recommended to use two curves as illustrated in Fig. 3.34.

An arbitrary number of penetration-dependent curves **PXNAME\_I** and **PYNAME\_I** may be given and SIMLA will interpolate the force and the stiffness based on the actual penetration. Zero force is always assumed at zero penetration. Extrapolation of the force and stiffness is done if the actual penetration exceeds the largest user-defined penetration value. Hence, very large force and stiffness magnitudes may result if the actual penetration exceeds the largest user-defined penetration value significantly. The sign of the user-defined penetration values are of no importance because SIMLA applies the absolute value in the computations.

The actual penetration includes the contributions from the soil elastic penetration, the element initial displacement defined by the **INISTR** card in Section 3.34, the KP-based soil embedment defined by the **ELPROP** card in Section 3.23.11 and the soil embedment modelled according to Section 3.8.3.

---

**EXAMPLE:**


---

```
#      name   type      mux  muy  mutx  xname  yname  zname  txname
MATERIAL soil1  r_contact 0.5  1.0  0.7  soilx  soily  soilz  soiltx
#
# xychoice
# coulomb-userdefined-userdefined  Nx  Ny  px1  pxname1  py1  pyname1  py2  pyname2
coulomb-userdefined-userdefined  1    2    0.1  soilx_1  0.1  soily_1  0.2  soil_y2
```

---

**EXAMPLE:**


---

```
#      name   type      mux  muy  mutx  xname  yname  zname  txname  xychoice
MATERIAL soil1  r_contact 0.5  1.0  0.5  soilx  soily  soilz1  soiltz
MATERIAL soil2  r_contact 0.5  1.0  0.5  soilx  soily  soilz2  soiltz  coulomb
MATERIAL roller1 r_contact 0.3  0.3  0.5  soilx  soily  soilz3  soiltz
MATERIAL roller2  r_contact 0.3  0.3  0.5      fx      fy  soilz4      mtz  userdefined
MATERIAL roller3  r_contact 0.3  0.3  0.5      fx      fy  soilz5      mtz  userdefined
```

---

If the optional parameter **XYCHOICE=USERDEFINED** is omitted, the material curves for the local  $x$ -direction and the local  $y$ -direction are to be defined as a consecutive number of points defining displacement versus unit force per length. This is because SIMLA will scale the unit force by  $\mu_x F_z L$  and  $\mu_y F_z L$  for the local  $x$ - and  $y$ -directions, respectively. Likewise, the  $x$ -rotation curve is to be defined as a consecutive number of points defining  $x$ -rotation versus unit force per length. SIMLA will then scale the unit force by  $\mu_{tx} F_z L r$  to give the  $x$ -moment. Here,  $\mu_x$ ,  $\mu_y$  and  $\mu_{tx}$  are the friction factors for the local  $x$ -direction,  $y$ -direction and  $x$ -rotation,  $F_z$  is the local  $z$ -direction contact force (unit: F),  $r$  is the pipe external radius, and  $L$  is the length of the pipe element in contact.

For **XYCHOICE=USERDEFINED**, the  $x$ - and  $y$ -direction curves are to be defined as a consecutive number of points defining displacement versus force per unit length, i.e. no scaling with  $\mu F_z$  is applied. Likewise, the  $x$ -rotation curve is to be defined as a consecutive number of points defining  $x$ -rotation versus a  $y$ -direction force per unit length that SIMLA will multiply by the pipe external radius to give the  $x$ -moment.

If the **YNAME** curve is assigned the **CLAY\_Y** and **SAND\_Y** models in Sections 3.37.12 and 3.37.13, the  $y$ -direction force will not be scaled with  $\mu F_z$  regardless of the value specified for **XYCHOICE**.

For the standard case, the material curves **XNAME**, **YNAME**, **ZNAME** and **TXNAME** need to be defined using the **HYCURVE** or **EPCURVE** options. The latter option should be applied in modelling of friction, otherwise the friction force reversal will not be correctly modelled. The material curve units are defined in Table 3.2. In particular, it should be noted that the **TXNAME** material curve is to be defined as a consecutive number of points defining *x*-rotation (unit: radian) versus a *y*-direction force or unit force which SIMLA multiplies by the pipe external radius to give the *x*-moment.

The local *x*-direction is directed along the current *x*-axis of the element system, and will undergo the same rotations as the attached pipe node during the analysis. Hence, the element initial *x*-axis as defined by the **ELORIENT** card must be directed along the pipe element initial *x*-axis in order to get the local *x*- and *y*-force directed along and transverse to the pipeline, respectively, and to get the local *x*-moment acting about the pipeline *x*-axis. The local *z*-direction is adjusted to coincide with the radial direction of the pipe cross-section provided that the element initial *x*-axis and the pipe element initial *x*-axis coincide, and the local *y*-direction is computed by the cross product of local *z*-direction unit vector and the local *x*-direction unit vector.

### 3.37.12 CLAY\_Y (CONT126)

**MATERIAL MNAME CLAY\_Y KE GAMD SU TIMEON IZSOIL SU\_Z3**

**KE**: Elastic soil stiffness (unit:  $\text{FL}^{-2}$ ).

**GAMD**: Soil dry weight per volume (unit :  $\text{FL}^{-3}$ ).

**SU**: Undrained shear strength (unit:  $\text{FL}^{-2}$ ).

**TIMEON**: Activation time for the **CLAY\_Y** model (unit: T).

**IZSOIL**: Force model in z-direction (unit: -)

**1** : Clay hyper-elastic force-penetration relation based on the initial embedment model described in the SIMLA Theory Manual.

**2** : Force-penetration relation as defined by the **ZNAME** curve in Sections 3.37.8 and 3.37.11.

**SU\_Z3**: The undrained shear strength value  $s_{uz3}$  used for defining the residual penetration reduction factor in Eq. (3.47). Recommended value is 4 kPa. (unit:  $\text{FL}^{-2}$ ).

The model is applicable for on-bottom stability analysis on clay seabeds. The pipe-soil interaction force consists of a Coulomb friction contribution and a soil penetration-dependent passive resistance contribution. The model is implemented for the **CONTACT** and **R\_CONTACT** material options, in which **YNAME** must refer to the name

of the **CLAY\_Y** model and the Coulomb friction coefficient must be defined by the parameter **MUY**.

The elastic soil penetration will be equal to zero when the z-direction contact force becomes equal to zero. This may be unwanted in on-bottom stability analysis since less hydrodynamic load reduction then results from the penetration-dependent reduction factors **YPENFACNAME** and **ZPENFACNAME** in Section 3.14. To avoid this effect, it is recommended to use the option **IZSOIL=2** and apply a **HYCURVE** model in Section 3.37.4 with relatively high stiffness such that the elastic soil penetration becomes negligible. In that case, the initial soil embedment predicted by using the option **IZSOIL=1** in static analysis must be included either by using KP-based initial soil embedment or by using initial displacement for the seabed contact element, see Sections 3.23.11 and 3.34.

A reduction factor for the residual penetration  $z_3$  is applied to account for stiff clay according to,

$$\begin{aligned} z_3 &= \hat{z}_3 \left( \frac{s_{uz3}}{s_u} \right)^4 && \text{if } s_u > s_{uz3} \\ z_3 &= \hat{z}_3 && \text{if } s_u \leq s_{uz3} \end{aligned} \quad (3.47)$$

where  $\hat{z}_3$  is the theoretical residual penetration and  $z_3$  is the actual residual penetration applied for calculating the residual force, see further details in the SIMLA Theory Manual. The reduction factor for  $z_3$  can be de-activated simply by setting **SU\_Z3** to a larger value than **SU**.

The soil lateral stiffness is equal to zero prior to **TIMEON**. Convergence difficulties due to zero stiffness in static analysis can be avoided by using the time-controlled spring option described in Section 3.23.11.

The passive resistance force is computed based on the total soil penetration consisting of contributions from the soil elastic penetration, the soil penetration developed by lateral pipeline displacement, the element initial displacement defined by the **INISTR** card in Section 3.34, the KP-based soil embedment defined by the **ELPROP** card in Section 3.23.11 and the soil embedment modelled according to Section 3.8.3, see the SIMLA Theory Manual for further details. The penetration developed by lateral pipeline displacement is however not included in the computed pipe motion in order to avoid unwanted oscillations in dynamic analysis.

Be cautious in modelling of pipe-in-pipe systems or models where more than one pipe element group is involved. In such cases, the total submerged weight shall be applied to the pipe element group that is linked to the CONT126 element group. This is because the pipe-soil vertical contact force is normalized with respect to the pipe submerged weight, see SIMLA Theory Manual. The other pipe element group(s) involved must then

be assigned zero submerged weight.

The result types available for the model are defined in Section 3.12.3.

Further details of the model are given in Ref. (Verley and Lund, 1995b) and in the SIMLA Theory Manual.

EXAMPLE:

---

```
#      name   type     rmyx   rmyy   xname   yname   zname
MATERIAL soil   contact    0.2     0.2   soilx   soily   soilz
#
#      name   type     ke     gamd   su   timeon   izsoil
MATERIAL soily  clay_y   65000  18000  800   1.0       1
```

---

### 3.37.13 SAND\_Y (CONT126)

**MATERIAL MNAME SAND\_Y KE GAMSAT TIMEON IZSOIL**

**KE**: Elastic soil stiffness (unit:  $\text{FL}^{-2}$ ).

**GAMSAT**: Soil saturated weight per volume (unit :  $\text{FL}^{-3}$ ).

**TIMEON**: Activation time for the **SAND\_Y** model (unit: T).

**IZSOIL**: Force model in z-direction (unit: -)

**1** : Sand hyper-elastic force-penetration relation based on the initial embedment model described in the SIMLA Theory Manual.

**2** : Force-penetration relation as defined by the **ZNAME** curve in Sections 3.37.8 and 3.37.11.

The model is applicable for on-bottom stability analysis on sandy seabeds. The pipe-soil interaction force consists of a Coulomb friction contribution and a soil penetration-dependent passive resistance contribution. The model is implemented for the **CONTACT** and **R\_CONTACT** material options, in which **YNAME** must refer to the name of the **SAND\_Y** model and the Coulomb friction coefficient must be defined by the parameter **MUY**.

The elastic soil penetration will be equal to zero when the z-direction contact force becomes equal to zero. This may be unwanted in on-bottom stability analysis since less hydrodynamic load reduction then results from the penetration-dependent reduction factors **YPENFACNAME** and **ZPENFACNAME** in Section 3.14. To avoid this effect, it is recommended to use the option **IZSOIL=2** and apply a **HYCURVE** model in Section 3.37.4 with relatively high stiffness such that the elastic soil penetration becomes negligible. In that case, the initial soil embedment predicted by using the option **IZSOIL=1**

in static analysis must be included either by using KP-based initial soil embedment or by using initial displacement for the seabed contact element, see Sections 3.23.11 and 3.34.

Note that **GAMSAT** is the soil saturated weight per volume  $\gamma_{sat} = \gamma_s + \rho_w g$ , where  $\gamma_s$  is the soil submerged weight per volume,  $\rho_w$  is the water density and  $g$  is the gravity acceleration.

The soil lateral stiffness is equal to zero prior to **TIMEON**. Convergence difficulties due to zero stiffness in static analysis can be avoided by using the time-controlled spring option described in Section 3.23.11.

The passive resistance force is computed based on the total soil penetration consisting of contributions from the soil elastic penetration, the soil penetration developed by lateral pipeline displacement, the element initial displacement defined by the **INISTR** card in Section 3.34, the KP-based soil embedment defined by the **ELPROP** card in Section 3.23.11 and the soil embedment modelled according to Section 3.8.3, see the SIMLA Theory Manual for further details. The penetration developed by lateral pipeline displacement is however not included in the computed pipe motion in order to avoid unwanted oscillations in dynamic analysis.

Be cautious in modelling of pipe-in-pipe systems or models where more than one pipe element group is involved. In such cases, the total submerged weight shall be applied to the pipe element group that is linked to the CONT126 element group. This is because the pipe-soil vertical contact force is normalized with respect to the pipe submerged weight, see SIMLA Theory Manual. The other pipe element group(s) involved must then be assigned zero submerged weight.

The result types available for the model are defined in Section 3.12.3.

Further details of the model are given in Ref. (Verley and Sotberg, 1994) and in the SIMLA Theory Manual.

#### EXAMPLE:

---

```
#      name   type      rmyx   rmyy   xname   yname   zname
MATERIAL soil   contact    0.2     0.2   soilx   soily   soilz
#
#      name   type      ke      gamsat   timeon   izsoil
MATERIAL soily sand_y   65000   18000    1.0       1
```

---

#### 3.37.14 BREAKOUT\_Y (CONT126)

The input card has the following format:

**MATERIAL MNAME BREAKOUT\_Y KE TIMEON MODTYP ...**

**KE:** Elastic soil stiffness (unit:  $\text{FL}^{-2}$ ).

**TIMEON:** Model activation time and reference time for constant vertical force and constant penetration (unit: T).

**MODTYP:** Available model types:

**UNDRAIN\_NGI :** NGI model for undrained soil.

**UNDRAIN\_DNV2 :** DNV-RP-F114 model 2 for undrained soil.

**DRAIN\_NGI :** NGI model for drained soil.

**USERDEF :** Userdefined model.

The lateral characteristic forces and the mobilization displacements are calculated based on the soil penetration and the vertical contact force at time=**TIMEON**. Hence, the above break-out models are applicable for cases where the potential development of soil penetration is small.

The calculated characteristic forces and mobilization displacements are printed to the SIMLA output file (prefix.sof) at the first time step after time=**TIMEON**.

The lateral stiffness is equal to zero prior to time=**TIMEON**. Convergence difficulties due to zero stiffness in static analysis can be avoided by using the time-controlled spring option described in Section 3.23.11.

### Undrained soil

The input card has the following format for **MODTYP** equal to **UNDRAIN\_NGI** and **UNDRAIN\_DNV2**:

**MATERIAL MNAME BREAKOUT\_Y KE TIMEON MODTYP GAMS SU SOFTYP  
[C D E F G H I J K M A1 B1 C1 A2 B2 C2]**

with the following mandatory input parameters:

**GAMS:** Soil submerged weight per volume (unit :  $\text{FL}^{-3}$ ).

**SU:** Soil undrained shear strength. Penetration-dependent values will be applied if a table name is specified, see Section 3.49. The parameter will be constant if a floating number or integer is specified (unit :  $\text{FL}^{-2}$ ).

**SOFTYP:** Option for softening of characteristic forces. The following options can be selected:

**ANTISYM :** The characteristic forces will be softened for the current displacement direction only, i.e. anti-symmetric softening. The break-out force will

undergo reduction towards the residual resistance force when the displacement exceeds the breakout mobilization displacement. Further, when the residual resistance mobilization displacement is exceeded, the passive residual resistance is set to zero and a Coulomb model with friction coefficient **J** will be applied for subsequent displacement half-cycles in that displacement direction.

**SYM** : Same softening as for **ANTISYM**, however, the softening occurs simultaneously for the opposite displacement direction as well, i.e. symmetric softening. This is the most conservative option for lateral stability assessments.

**NONE** : The characteristic forces will not undergo softening and the passive residual resistance is always included. This option may be non-conservative for lateral stability assessments.

The optional input parameters are as follows:

- C**:  $c$ , optional, default value: 0.78 for UNDRAIN\_NGI and 1.7 for UNDRAIN\_DNV2 (unit: -).
- D**:  $d$ , optional, default value: 0.61 for UNDRAIN\_NGI and 0.61 for UNDRAIN\_DNV2 (unit: -).
- E**:  $e$ , optional, default value: 0.12 for UNDRAIN\_NGI and 0.23 for UNDRAIN\_DNV2 (unit: -).
- F**:  $f$ , optional, default value: 0.83 for UNDRAIN\_NGI and 0.83 for UNDRAIN\_DNV2 (unit: -).
- G**:  $g$ , optional, default value: 0.71 for UNDRAIN\_NGI and 0.6 for UNDRAIN\_DNV2 (unit: -).
- H**:  $h$ , optional, default value: 1.0 for UNDRAIN\_NGI and 1.0 for UNDRAIN\_DNV2 (unit: -).
- I**:  $i$ , optional, default value: 2.0 for UNDRAIN\_NGI and 2.0 for UNDRAIN\_DNV2 (unit: -).
- J**:  $j$ , optional, default value: 0.3 for UNDRAIN\_NGI and 0.32 for UNDRAIN\_DNV2 (unit: -).
- K**:  $k$ , optional, default value: 0.76 for UNDRAIN\_NGI and 0.8 for UNDRAIN\_DNV2 (unit: -).
- M**:  $m$ , optional, default value: 0.8 for UNDRAIN\_NGI and 0.8 for UNDRAIN\_DNV2 (unit: -).
- A1**:  $a_1$ , optional, default values: 0.4 for UNDRAIN\_NGI and 0.02 for UNDRAIN\_DNV2 (unit: -).
- B1**:  $b_1$ , optional, default values: 0.1 for UNDRAIN\_NGI and 0.25 for UNDRAIN\_DNV2 (unit: -).

- C1:**  $c_1$ , optional, default values: 0.41 for UNDRAIN\_NGI and 1.5 for UNDRAIN\_DNV2 (unit: -).
- A2:**  $a_2$ , optional, default value: 1.85 for UNDRAIN\_NGI and dummy for UNDRAIN\_DNV2 (unit: -).
- B2:**  $b_2$ , optional, default value: 0.2 for UNDRAIN\_NGI and dummy for UNDRAIN\_DNV2 (unit: -).
- C2:**  $c_2$ , optional, default value: 1.5 for UNDRAIN\_NGI and dummy for UNDRAIN\_DNV2 (unit: -).

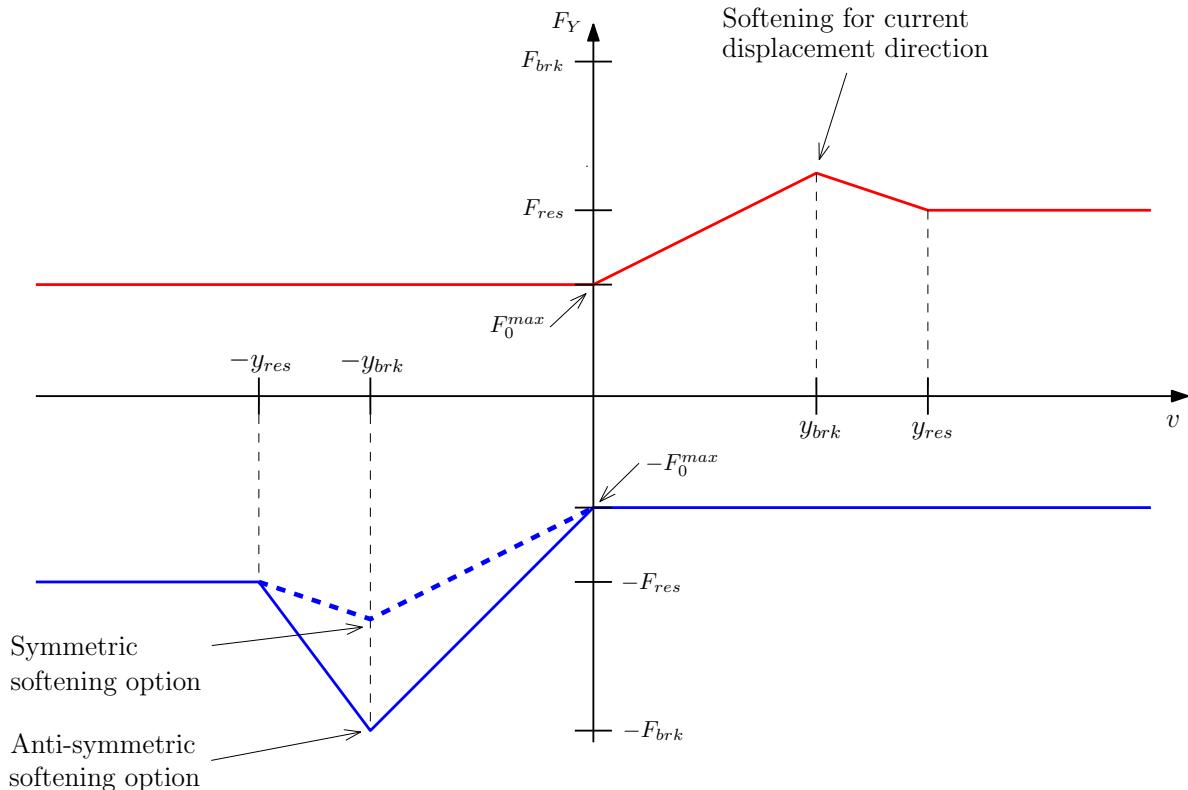


Figure 3.35: Illustration of symmetric and anti-symmetric softening options. Here, the break-out displacement  $y_{brk}$  has been exceeded in the positive displacement direction. This results in softening of the positive red yield curve. For the symmetric softening option, the blue negative yield force curve for the opposite displacement direction will undergo the same softening. In contrast, if anti-symmetric softening is selected, there will be no softening for the opposite displacement direction.

The anti-symmetric and symmetric softening behaviors governed by **SOFTYP** is illustrated in Fig. 3.35. Please see the SIMLA Theory Manual for further information about the applied yield curves, the hardening behavior and the softening behavior.

The optional parameters above represent model calibration factors which are defined through the model equations presented in the SIMLA Theory Manual.

If a **TABLE** from Section 3.49 is applied for **SU**, the penetration values are to be defined

in the first column and the undrained shear strength values in the second column. The penetration has dimension length and is negative when the pipe penetrates into the soil. See example below.

Be aware that the friction coefficient **MUY** specified in the **CONTACT** or **R\_CONTACT** input cards in Sections 3.37.8 and 3.37.11 will be a dummy parameter. The friction force associated with the break-out resistance is governed by the parameters **E** and **F**. The Coulomb friction coefficient applied for the residual resistance is defined by the parameter **J**.

The characteristic forces and the mobilization displacements shown in Fig. 3.35 are printed to the SIMLA output file (prefix.sof) at time=**TIMEON**.

---

EXAMPLE:

---

```
#      name   type     rmyx   rmyy   xname   yname   zname
MATERIAL soil    contact  0.4     0.0   soilx   soily   soilz
#
#      name   type       ke     timeon   modtyp        gams   su      softyp
MATERIAL soily  breakout_y  65000  1.0     undrain_dnv2  8000   1200   antisym
```

---



---

EXAMPLE:

---

```
#      name   type     rmyx   rmyy   xname   yname   zname
MATERIAL soil    contact  0.2     0.0   soilx   soily   soilz
#
#      name   type       ke     timeon   modtyp        gams   su      softyp
MATERIAL soily  breakout_y  65000  1.0     undrain_ngi  6000   su_intact   antisym
#
#      name      ncol
TABLE su_intact  2
-1.0  2000.0
-0.3  1200.0
-0.1  800.0
-0.0   0.0
```

---

## Drained soil

The input card has the following format for **MODTYP=DRAIN\_NGI**:

```
MATERIAL MNAME BREAKOUT_Y KE TIMEON MODTYP GAMS SOFTYP [B  
C D E F A1 B1 C1 A2 B2 C2]
```

with the following mandatory input parameters:

**GAMS:** Soil submerged weight per volume (unit :  $\text{FL}^{-3}$ ).

**SOFTYP:** Option for softening of characteristic forces. The following options can be selected:

**ANTISYM :** The characteristic forces will be softened for the current displacement direction only, i.e. anti-symmetric softening. The break-out force will undergo reduction towards the residual resistance force when the displacement exceeds the breakout mobilization displacement. Further, when the residual resistance mobilization displacement is exceeded, the passive residual resistance is set to zero and a Coulomb model with friction coefficient **F** will be applied for subsequent displacement half-cycles in that displacement direction.

**SYM :** Same softening as for **ANTISYM**, however, the softening occurs simultaneously for the opposite displacement direction as well, i.e. symmetric softening. This is the most conservative option for lateral stability assessments.

**NONE :** The characteristic forces will not undergo softening and the passive residual resistance is always included. This option may be non-conservative for lateral stability assessments.

The optional input parameters are as follows:

- B:**  $b$ , optional, default value: 1.61 (unit: -).
- C:**  $c$ , optional, default value: 0.1 (unit: -).
- D:**  $d$ , optional, default value: 0.9 (unit: -).
- E:**  $e$ , optional, default value: 0.42 (unit: -).
- F:**  $f$ , optional, default value: 0.42 (unit: -).
- A1:**  $a_1$ , optional, default value: 0.4 (unit: -).
- B1:**  $b_1$ , optional, default value: 0.1 (unit: -).
- C1:**  $c_1$ , optional, default value: 0.41 (unit: -).
- A2:**  $a_2$ , optional, default value: 1.85 (unit: -).
- B2:**  $b_2$ , optional, default value: 0.2 (unit: -).
- C2:**  $c_2$ , optional, default value: 1.5 (unit: -).

The anti-symmetric and symmetric softening behaviors governed by **SOFTYP** is illustrated in Fig. 3.35. Please see the SIMLA Theory Manual for further information about the applied yield curves, the hardening behavior and the softening behavior.

The optional parameters represent model calibration factors which are defined through the model equations presented in the SIMLA Theory Manual.

Be aware that the friction coefficient **MUY** specified in the **CONTACT** or **R\_CONTACT** input cards in Sections 3.37.8 and 3.37.11 will be a dummy parameter. The Coulomb

friction coefficient associated with the break-out resistance and the residual resistance are defined by the parameters **E** and **F**, respectively.

The characteristic forces and the mobilization displacements shown in Fig. 3.35 are printed to the SIMLA output file (prefix.sof) at time=**TIMEON**.

EXAMPLE:

---

```
#      name   type     rmyx   rmyy   xname   yname   zname
MATERIAL soil    contact  0.4     0.0    soilx   soily   soilz
#
#      name   type        ke      timeon   modtyp      gams   softyp
MATERIAL soily  breakout_y 65000    1.0     drain_ngi  8800   antisym
```

---

## Userdefined model

The input card has the following format for **MODTYP=USERDEF**:

```
MATERIAL MNAME BREAKOUT_Y KE TIMEON MODTYP F0_MAX FBRK
FRES_PASS YBRK YRES SOFTYP
```

with the following input parameters:

**F0\_MAX**: Maximum force at zero mobilization displacement. See  $F_0^{max}$  in Fig. 3.35.  
(unit :  $FL^{-1}$ ).

**FBRK**: Total break-out force. See  $F_{brk}$  in Fig. 3.35. (unit :  $FL^{-1}$ ).

**FRES\_PASS**: Passive part of residual resistance force (unit :  $FL^{-1}$ ).

**YBRK**: Mobilization displacement for break-out force (unit : L).

**YRES**: Mobilization displacement for residual resistance force (unit : L).

**SOFTYP**: Option for softening of characteristic forces. The following options can be selected:

**ANTISYM** : The characteristic forces will be softened for the current displacement direction only, i.e. anti-symmetric softening. The break-out force will undergo reduction towards the residual resistance force when the displacement exceeds the breakout mobilization displacement. Further, when the residual resistance mobilization displacement is exceeded, the passive residual resistance is set to zero and a Coulomb model with friction coefficient **MUY** as defined in Sections 3.37.8 and 3.37.11 will be applied for subsequent displacement half-cycles in that displacement direction.

**SYM** : Same softening as for **ANTISYM**, however, the softening occurs simultaneously for the opposite displacement direction as well, i.e. symmetric softening. This is the most conservative option for lateral stability assessments.

**NONE** : The characteristic forces will not undergo softening and the passive residual resistance is always included. This option may be non-conservative for lateral stability assessments.

The anti-symmetric and symmetric softening behaviors governed by **SOFTYP** is illustrated in Fig. 3.35. This figure also shows the characteristic forces and the mobilization displacements defined by the above input parameters. Please see the SIMLA Theory Manual for further information about the applied yield curves, the hardening behavior and the softening behavior.

The friction part of the residual resistance force is calculated based on the friction coefficient **MUY** as specified in the **CONTACT** or **R\_CONTACT** input cards in Sections 3.37.8 and 3.37.11. The residual resistance force depicted in Fig. 3.35 is then given by,

$$F_{res} = \text{MUY} \cdot F_z + \text{FRES_PASS} \quad (3.48)$$

The characteristic forces and the mobilization displacements shown in Fig. 3.35 are printed to the SIMLA output file (prefix.sof) at time=**TIMEON**.

EXAMPLE:

---

```
#      name   type      rmyx   rmyy   xname   yname   zname
MATERIAL soil    contact   0.4     0.3   soilx   soily   soilz
#
#      name   type          ke      timeon   modtyp   F0_max   Fbrk
MATERIAL soily  breakout_y  90000    2.0    userdef   5.5     13.0
#
#      Fres_pass   ybrk    yres   softyp
        7.0       0.01   0.17   antisym
```

---

### 3.37.15 UNDRAIN\_Z (CONT126)

**MATERIAL MNAME UNDRAIN\_Z GAMS SU [A B C D E F]**

**GAMS**: Soil submerged weight per volume (unit :  $\text{FL}^{-3}$ ).

**SU**: Soil undrained shear strength. Penetration-dependent values will be applied if a table name is specified, see Section 3.49. The parameter will be constant if a floating number or integer is specified (unit :  $\text{FL}^{-2}$ ).

**A**:  $a$ , optional, default value: 6.0 (unit: -).

- B:**  $b$ , optional, default value: 0.25 (unit: -).
- C:**  $c$ , optional, default value: 3.4 (unit: -).
- D:**  $d$ , optional, default value: 0.5 (unit: -).
- E:**  $e$ , optional, default value: 1.5 (unit: -).
- F:**  $f$ , optional, default value: 1.0 (unit: -).

The optional parameters represent model calibration factors which are defined through the model equations presented in the SIMLA Theory Manual.

If a **TABLE** from Section 3.49 is applied for **SU**, the penetration values are to be defined in the first column and the undrained shear strength values in the second column. The penetration has dimension length and is negative when the pipe penetrates into the soil. See example below.

The elastic soil penetration will be equal to zero when the z-direction contact force becomes equal to zero. This may be unwanted in on-bottom stability analysis since less hydrodynamic load reduction then results from the penetration-dependent reduction factors **YPENFACNAME** and **ZPENFACNAME** in Section 3.14. To avoid this effect, it is recommended to de-activate the **UNDRAIN\_Z** model and instead apply a **HY-CURVE** model in Section 3.37.4 with relatively high stiffness such that the elastic soil penetration becomes negligible. In that case, the initial soil embedment predicted by the **UNDRAIN\_Z** model in static analysis must be included either by using KP-based initial soil embedment or by using initial displacement for the seabed contact element, see Sections 3.23.11 and 3.34.

---

#### EXAMPLE:

---

```
#      name   type     rmyx   rmyy   xname   yname   zname
MATERIAL soil    contact  0.4     0.0   soilx   soily   soilz
#
#      name   type       gams   su
MATERIAL soilz  undrain_z 8000    1200
```

---



---

#### EXAMPLE:

---

```
#      name   type     rmyx   rmyy   xname   yname   zname
MATERIAL soil    contact  0.4     0.0   soilx   soily   soilz
#
#      name   type       gams   su
MATERIAL soilz  undrain_z 8000    suitable
#
#      name   ncol
TABLE suitable  2
-1.0  9000.0
```

---

```
-0.3 2400.0
-0.1 800.0
-0.0 0.0
```

---

### 3.37.16 DRAIN\_Z (CONT126)

**MATERIAL MNAME DRAIN\_Y GAMS [A B]**

**GAMS:** Soil submerged weight per volume (unit :  $\text{FL}^{-3}$ ).

**A:**  $a$ , optional, default value: 0.09 (unit: -).

**B:**  $b$ , optional, default value: 0.75 (unit: -).

The optional parameters represent model calibration factors which are defined through the model equations presented in the SIMLA Theory Manual.

The elastic soil penetration will be equal to zero when the z-direction contact force becomes equal to zero. This may be unwanted in on-bottom stability analysis since less hydrodynamic load reduction then results from the penetration-dependent reduction factors **YPENFACNAME** and **ZPENFACNAME** in Section 3.14. To avoid this effect, it is recommended to de-activate the **DRAIN\_Z** model and instead apply a **HYCURVE** model in Section 3.37.4 with relatively high stiffness such that the elastic soil penetration becomes negligible. In that case, the initial soil embedment predicted by the **DRAIN\_Z** model in static analysis must be included either by using KP-based initial soil embedment or by using initial displacement for the seabed contact element, see Sections 3.23.11 and 3.34.

EXAMPLE:

---

```
#      name   type     rmyx   rmyy   xname   yname   zname
MATERIAL soil    contact  0.4     0.0   soilx   soily   soilz
#
#      name   type     gams
MATERIAL soilz  drain_z  6500
```

---

### 3.37.17 HYPERELASTIC

**MATERIAL MNAME HYPERELASTIC POISS RHO0 TALFA TECOND**

**HEATC EPS SIGMA**

... ...

**POISS:** Poisson's ratio (-)  
**RHO0:** Density (dummy) (unit:  $\text{ML}^{-3}$ )  
**TALFA:** Temperature elongation coeff. (unit  $K^{-1}$ )  
**TECOND:** Thermal conductivity (dummy) (unit:  $\text{ET}^{-1}\text{L}^{-1}\text{TE}^{-1}$ )  
**HEATC:** Heat capacity (dummy) (unit:  $\text{E}^{-1}\text{M}^{-1}\text{TE}^{-1}$ )  
**EPS:** Strain  
**SIGMA:** Stress (unit:  $\text{FL}^{-2}$ ) values describing the material curve for both positive and negative values of strain and stress.

EXAMPLE: \_\_\_\_\_

```
#      name      type      poiss  ro    talfa   tecond  heatc  eps   sigma
MATERIAL pipemat3 hyperelastic 0.3    7850 1.17e-5 50     800    0       0
                                         0.005 450
                                         1.0    700
#
```

---

### 3.37.18 LINSPRING (SPRING136)

**MATERIAL MNAME LINSPRING KX KY KZ KTX KTY KTZ**

**KX:** Stiffness in local x-direction ( $\text{FL}^{-1}$ )  
**KY:** Stiffness in local y-direction ( $\text{FL}^{-1}$ )  
**KZ:** Stiffness in local z-direction ( $\text{FL}^{-1}$ )  
**KTX:** Stiffness about local x-axis (FL)  
**KTY:** Stiffness about local y-axis (FL)  
**KTZ:** Stiffness about local z-axis (FL)

### 3.37.19 SPRING (CONT128)

**MATERIAL MNAME SPRING XNAME YNAME ZNAME MXNAME**

**XNAME:** Axial x-direction material curve name  
**YNAME:** Transverse y-direction material curve name  
**ZNAME:** Vertical z-direction material curve name  
**MXNAME:** Torsion moment material curve name

EXAMPLE: \_\_\_\_\_

```
#      name      type      xname   yname   zname
MATERIAL rockdump1 spring    dumpx  dumpy   dumpz
```

It is noted that the material curves need to be defined using the **HYCURVE** or **EPCURVE** options. For both cases the curve is to be defined as a consecutive number of points defining displacement versus force per unit length.

### 3.37.20 GENSPRING (SPRING137)

```
MATERIAL MNAME GENSPRING XNAME YNAME ZNAME RXNAME RY-
NAME RZNAME [XYCHOICE]
```

**XNAME**: *x*-direction material curve name

**YNAME**: *y*-direction material curve name

**ZNAME**: *z*-direction material curve name

**RXNAME**: *x*-rotation material curve name

**RYNAME**: *y*-rotation material curve name

**RZNAME**: *z*-rotation material curve name

**XYCHOICE**: Friction force scaling is introduced by specifying **COULOMB**. For this option, the parameter **ICOULCNTR** in Section 3.23.10 must be set, which governs whether the friction force is activated in the *xy*-plane or only in the *x*-direction. By selecting **USERDEFINED**, the forces in the *x*- and *y*-directions are taken from the userdefined material curves without any scaling, which also is default behavior when the optional parameter **XYCHOICE** is omitted.

EXAMPLE:

---

```
#  
#      name   type    spr1    spr2    spr3    spr4    spr5    spr6    xychoice  
MATERIAL vessel1 genspring surgesp2 yawsp heavesp rollsp pitchsp swaysp userdefined
```

---

It is noted that the material curves need to be defined using the **EPCURVE** or **HYCURVE** options in Sections 3.37.3 and 3.37.4, respectively. For both cases the curve is to be defined as a consecutive number of points defining displacement/rotation versus force/moment. As explained in Section 3.23.10, material curves that are scaled according to the **COULOMB** option should be defined by the **EPCURVE** option, otherwise reversal of the friction force will not be correctly modelled.

### 3.37.21 CONCRETE (HSHEAR342)

```
MATERIAL MNAME CONCRETE POISS RHO TALFA TECOND HEATC EPS SIGMA  
... ...
```

**POISS:** Poisson's ratio.

**RHO:** Density (dummy) (unit:  $\text{ML}^{-3}$ )

**TALFA:** Temperature elongation coeff.  $\alpha$  (unit  $K^{-1}$ )

**TECOND:** Thermal conductivity (dummy) (unit:  $\text{EL}^{-1}\text{TE}^{-1}$ )

**HEATC:** Heat capacity (dummy) (unit:  $\text{E}^{-1}\text{M}^{-1}\text{TE}^{-1}$ )

**EPS:** Strains, always starting at the compressive side and ends at the tensile side.

**SIGMA:** Stresses, always starting at the compressive side and ends at the tensile side (unit:  $\text{FL}^{-2}$ ).

The concrete material model is based on dividing the material curve given into two regions, the tensile region and the compressive region. Within each region separate material models are assumed. On the compressive side the material is taken to behave elastoplastic with full kinematic hardening, whereas on the tensile side, the material is taken to be hyperelastic. The stress and strain characteristic must contain the strain and stress values 0 0.

EXAMPLE:

---

```
#  
#      name    type    poiss  ro   talfa   tecond  heatc   eps   sigma ..1,n  
MATERIAL concmat concrete 0.3    7850 1.17e-5  50       800     -0.05   -17  
                           -0.002   -15  
                           0         0  
                           0.0002   2  
                           0.05     3
```

---

### 3.38 MOVE \_ GROUP - move group during autostart

In order to enable complex structures to be moved by the autostart option, the **MOVE \_ GROUP** command enables the user to move the strucure equal to a given node as part of the autostart. The following format is applied.

**MOVE \_ GROUP NOD1 ELGR [.. ELGR]**

where

**NOD1:** Node ID. If a negative **NOD1** is given, then the nodes are incremented by 1 for each element node. The autostart logic is working such that all nodes having same coordinates as the autostart ones will be moved accordingly. The negative **NOD1** can be used if the user wants to force another node system being eccentric relative to the autostart node system to be moved accordingly.

**ELGR:** Element group name.

EXAMPLE: \_\_\_\_\_

```
#  
# Contact interface data:  
#-----  
#           NODEID   Group name 1   Group name 2  
MOVE_GROUP    101       Tee-structure Templatestructure
```

### 3.39 NOCOOR - Nodal coordinates of model

By the **NOCOOR** command, the user can specify the initial coordinates of the elements. The nodal coordinates are defined by a card starting as follows:

**NOCOOR TYPE ...**

where

**TYPE:** Type of coordinate definition. Three values are valid:

**COORDINATES:** Nodal points are given in global coordinates system.

**POLAR:** Nodal points are given in a polar coordinates system . Local origo and orientation of the polar system must be specified. Nodal coordinates will be described along local x-axis by defining radius, angles and distance along the x-axis for each node.

**ROTDISP:** Enables the user to rotate and displace a given set of coordinates

If option **COORDINATES** is selected, the **NOCOOR** card has the following elements:

**NOCOOR COORDINATES NODE XCOR YCOR ZCOR**

[ .. .. .. .. ]

**[REPEAT N NODINC XINC YINC ZINC]**

**NODE:** Node ID number of a node.

**XCOR:** x-coordinate.

**YCOR:** y-coordinate.

**ZCOR:** z-coordinate.

Under option **COORDINATES**, the **REPEAT** card has the following parameters:

**N:** Number of repeats (total number of times the nodal list defined above is repeated, including the original nodal list).

**NODINC:** Nodal increment.

**XINC:** Increment in x-coordinate.

**YINC:** Increment in y-coordinate.

**ZINC:** Increment in z-coordinate.

If option **POLAR** is selected, the **NOCOOR** card has the following elements:

```
NOCOOR POLAR X0 Y0 Z0 BETA1 BETA2 BETA3 R NODE XCOR THETA
[ .. .. .. ]
[REPEAT N NODEINC XINC THETAINC]
```

**X0:** x-coordinate of local system.

**Y0:** y-coordinate of local system.

**Z0:** z-coordinate of local system.

**BETA1:** Rotation about x for local system.

**BETA2:** Rotation about y for local system.

**BETA3:** Rotation about z for local system.

**R:** Distance from local x-axis.

**NODE:** Node ID number of a node.

**XCOR:** x-coordinate along local x-axis.

**THETA:** Angle of node coordinate.

Under option **POLAR** the **REPEAT** sequence has the following parameters:

**N:** Number of repeats (total number of times the nodal list defined above is repeated, including the original nodal list).

**NODINC:** Nodal increment.

**XINC:** Increment along local x-axis.

**THETAINC:** Increment in angle.

If option **ROTDISP** is selected, the **NOCOOR** card has the following elements.

```
NOCOOR ROTDISP TX TY TZ XINC YINC ZINC
```

**NODE:** Node ID number of a node.

**TX:** Tait–Bryan angle around x axis.

**TY:** Tait–Bryan angle around the intrinsic y axis.

**TZ:** Tait–Bryan angle around the intrinsic z axis.

**XINC:** x-increment.

**YINC:** y-increment.

**ZINC:** z-increment.

An arbitrary sequence of nodes may be given, both for **POLAR** and **COORDINATES**, as indicated by the dots. However, the value of **NODE** must be constantly increasing. If **NODE** increases with more than 1, linear interpolation is used to create intermediate nodal coordinates.

By including the **REPEAT** command the **NOCOOR** sequence is repeated so that a total of **N** sequences is defined. The **REPEAT** command can be nested. Each instance will repeat all nodes which are defined so far in the **NOCOOR** card, including the nodes generated with **REPEATs** above.

The **ROTDISP** causes all nodes from the previous **ROTDISP** card or from the start of the **NOCOOR** sequence until the present **ROTDISP** card to be rotated and displaced with the given quantity.

EXAMPLE:

---

```
#  
#  
#  
NOCOOR COORDINATES  nodnr  x      y      z  
1           0       0       0  
11          0       0     -10  
21          0       0     -15  
41          0       0     -20
```

---

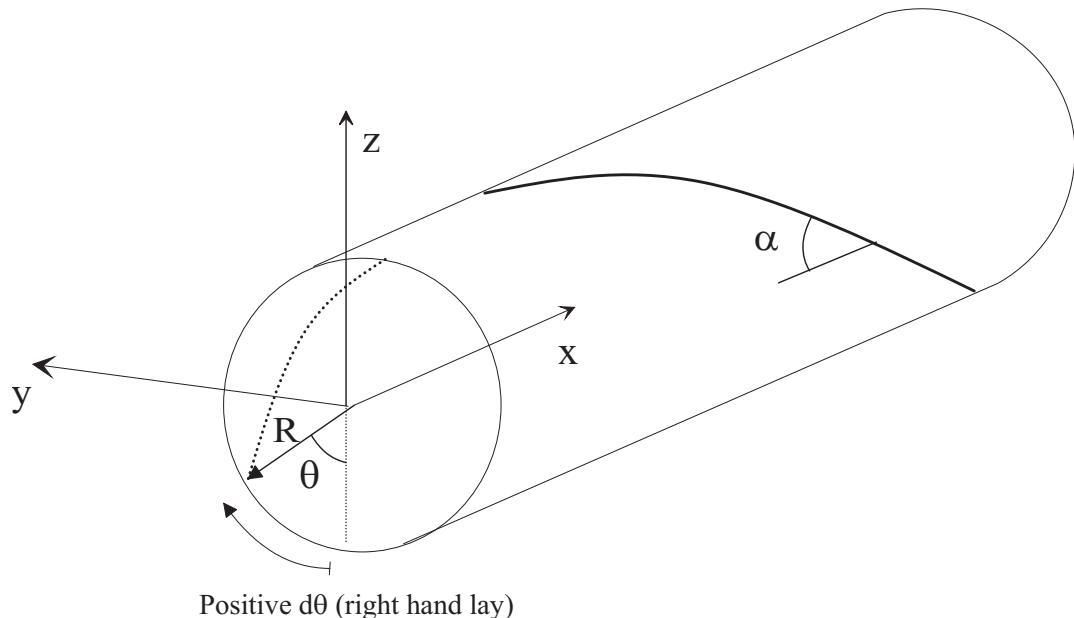


Figure 3.36: Polar coordinate definition.

### 3.40 NODPROP - node interpolated element properties

The **NODPROP** command allows the user to specify bellmouth or bend stiffener geometries, overruling the concept of constant geometry properties per group. The element properties in terms of radius and thickness is taken as the mean between the element nodes. The following format is applied:

```
NODPROP NODID RO RI
```

```
.. .. ..
```

where

**NODID**: Node ID number j.

**RO**: Outer radius at node j (unit: L).

**RI**: Inner radius at node j (unit: L).

The **NODPROP** command allows adjusting the element geometry properties of PIPE31 - PIPE39 and COMPIPE42 element types which are connected to the nodes at which **NODPROP** is specified.

The sequence of the three last numbers may be repeated for as many node numbers as defined. Values for missing node numbers will be determined by linear interpolation.

*Note:* The outer pipe diameter  $D_{op}$  of an element is calculated by averaging the radiiuses from the **NODPROP** card according to  $D_{op} = 0.5(2R_1 + 2R_2)$ , where  $R_1$  and  $R_2$  are the radiiuses at the element nodes. The outer diameter **ODP** specified in the **ELPROP** card will be over-ruled by  $D_{op}$ .

*Note:* The wrap diameter **ODW** specified in the **ELPROP** card in Sections 3.23.1, 3.23.2 and 3.23.6 is not adjusted. The hydrodynamic diameter is calculated as  $D = (1 - \eta)D_{op} + \eta D_{ow}$  where the wrap diameter  $D_{ow}$  is taken from **ODW** in the **ELPROP** input card and  $D_{op}$  is taken from the **NODPROP** card. Hence, **RKS=0.0** must be applied in the **ELPROP** card to get a hydrodynamic diameter corresponding to **RO** specified above.

*Note:* Regarding contact search, the inner diameter of the pipe is  $2 \cdot \text{RI}$ . The outer diameter is adjusted in the following manner: The difference between the outer diameter and the structural diameter is calculated as  $\Delta_p = \text{ODP} - 2 \cdot \text{RAD} + \text{TH}$  (refer to **ELPROP** card). Thereafter, the difference between the wrap diameter and the structural diameter is calculated as  $\Delta_w = \text{ODW} - 2 \cdot \text{RAD} + \text{TH}$  (refer to **ELPROP** card). The maximum value is given by  $\Delta_{\max} = \max(\Delta_p, \Delta_w)$ . With this, the outer diameter at an element node applied in the contact search is set equal to  $2 \cdot \text{RO} + \Delta_{\max}$ .

EXAMPLE: \_\_\_\_\_

#	nodid	Ro	Ri
NODPROP	1001	255	240
	1041	255	240
	1042	255	240
	1043	256	241
	1044	257	242
	1045	260	245
	1046	263	248

### 3.41 NOORIENT - orientation of nodes

The **NOORIENT** command is used to define the initial orientation of the nodes. As default these are oriented parallel to the global coordinate frame, i.e. if no **NOORIENT** card is present this will be the case. However, if constraint equations is to be introduced in a local coordinate system, the user is allowed to change the orientation to e.g. parallel to the element coordinate system, see Section 3.22. The orientation of the node may be defined by specifying a set of three consecutive Tait–Bryan angles ( $\theta_x$ ,  $\theta_y$ ,  $\theta_z$ ) that rotates the default coordinate system relative to the global coordinate system in the defined sequence, see Fig. 3.37. The format is similar to the **NOCOOR** command:

```
NOORIENT TYPE NODE X Y Z
[..  
 .. . . .  
[REPEAT N INC XINC YINC ZINC]
```

**TYPE**: Type of orientation. This parameter can only have the value.

The remaining parameters have the following meaning:

**NODE**: id of node.

**X**: Tait–Bryan angle around x-axis.

**Y**: Tait–Bryan angle around intrinsic y-axis.

**Z**: Tait–Bryan angle around intrinsic z-axis.

An arbitrary sequence may be given, as long as **NODE** is increasing. If the difference between two consecutive nodes is  $> 1$ , then linear interpolation is used to create intermediate nodal coordinates.

By including the **REPEAT** command the **NOORIENT** sequence is repeated **N** times:

**N**: Number of repetitions.

**INC**: Node increment.

**XINC**: Tait–Bryan angle increment.

**YINC**: Tait–Bryan angle increment.

**ZINC:** Tait–Bryan angle increment.

EXAMPLE: \_\_\_\_\_

```
#  
#      type          node      tx      ty      tz  
NOORIENT EULERANGLE      3001      0       0       0
```

---

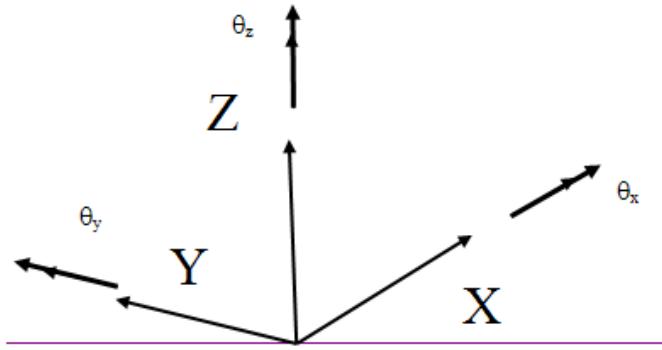


Figure 3.37: **NOORIENT** command.

### 3.42 PELOAD - external pressure and gravity loading

The external pressure and gravity loads are specified by the following format:

**PELOAD PRESHIST GRAVHIST**

where

**PRESHIST:** External pressure and buoyancy mass history number.

**GRAVHIST:** Dry mass history number.

Refer to the concepts of dry mass and dry buoyancy mass concepts applied in the **ELPROP** and **JOINTPR\_DEFINE** cards, see Sections 3.23 and 3.36. This means at a given time t:

$$\begin{aligned} f_z(t) &= m_{d0} T_d(t) g + (m_{d-b0} - m_{d0}) T_b(t) g \\ p_e &= \rho g (z_s - z) T_b(t) \\ m_d &= m_{d0} T_d(t) \end{aligned}$$

where:

$$f_z = \text{gravity load (unit: FL}^{-1}\text{)}$$

$m_{d0}$	= dry mass reference value as defined in <b>ELPROP</b> card (unit: ML-1)
$T_d(t)$	= dry mass history factor at time t, as defined in the <b>THIST</b> refered to by <b>GRAVHIST</b>
$m_{d-b0}$	= dry mass - buoyancy mass reference values as defined in <b>ELPROP</b> card
	= submerged weight/gravity (unit: ML <sup>-1</sup> )
$T_b(t)$	= buoyancy mass history factor at time t, as defined in the <b>THIST</b> refered to by <b>PRESHIST</b>
$m_d(t)$	= dry mass at a given time t
$g$	= gravity constant (unit: LT <sup>-2</sup> )
$\rho$	= seawater density (unit: ML <sup>-3</sup> )
$p_e$	= external pressure (unit: FL <sup>-2</sup> )

For a typical installation analysis, the masses normally refers to the empty pipe condition, hence only one history is required to handle buoyancy and dry masses. However, if a free-span analysis is to be carried out for empty, waterfilled and operation conditions, the dry mass change whereas the buoyancy is constant. In that case two separate histories will be needed (the dry mass change whereas the buoyancy is constant).

EXAMPLE: \_\_\_\_\_

```
#  
#           (gravity loading)  
#      externpreshistnr      gravloadhistnr  
PELOAD 100          100
```

---

### 3.43 PILOAD - internal pressure load

The internal pressure is specified by the following format:

**PILOAD HIST ELNR1 P1 [ELNR2 P2]**

where

**HIST**: Load history number.

**ELNR1**: First element ID number.

**P1**: Internal pressure of first element (unit: FL<sup>-2</sup>).

**ELNR2**: Last element ID number.

**P2**: Internal pressure of last element (unit: FL<sup>-2</sup>). Linear interpolation is applied between intermediate elements.

EXAMPLE: \_\_\_\_\_

```
#      (internal pressure load)
```

```
#      loadhistnr elnr1   p1      elnr2   p2
PILOAD 400        16001  69.0    16280  69.0
PILOAD 400        17001  69.0    17280  69.0
```

---

### 3.44 RAOPROP - RAO properties

Transfer functions that relates the amount of prescribed displacements for the dofs defined in the **CONSTR** card to the wave elevation is defined by the **RAOPROP** card where the format is as follows, including NDIR directions:

**RAOPROP NAME TYPE ...**

where

**NAME**: Name of the transfer function referred to in the **CONSTR** card.

**TYPE**: Type of RAO properties given, allowable values are **DEF** (initiation) or **CURVE**.

All transfer functions need to be initiated by a **DEF** option card.

If **TYPE = DEF** then the following format is applied:

**RAOPROP NAME DEF DIR1 [...] DIRN NAME1 [...] NAMEN**

**DIR1**: The direction angles of the first transfer function.

**DIRN**: The direction angles of the n-th transfer function.

**NAME1**: Name of the first transfer function.

**NAMEN**: Name of the n-th transfer function.

An arbitrary number of directions **DIR** can be given, and a corresponding set of transfer functions must be defined by use of the **CURVE** option defined below:

If **TYPE = CURVE** then the following format is applied:

**RAOPROP NAME CURVE FREQ A PHI**

...      ...    ...

**FREQ**: Frequency (unit: R).

**A**: Unit amplitude (unit: L/L or R/L).

**PHI**: Phase angle  $\phi_r$  (unit: R).

*Note:* List of frequencies must be given in increasing order.

The sequence **FREQ A PHI** is repeated for all frequencies. Note that for each direction an arbitrary number of points may be given, however, within each direction no variation of the number of points is allowed.

With reference to Fig. 3.1 a right-handed Cartesian coordinate system is used in SIMLA and so also for the RAO functions which refers to a right handed Cartesian coordinate system positioned at the vessel COG, see Fig. 3.38. Rotations are taken to be positive according to the right hand rule (placing the right hand thumb along one axis then the positive rotation about that axis follows the fingers). The RAO function is an unit function that is scaled with the wave elevation. Hence for the displacement DOFs the unit must be  $L/L$  = displacement per m wave elevation and for the rotation dofs the unit must be  $R/L$  = rotation in radians per m wave elevation. It is important to note that in SIMLA there is no vessel concept. The RAO is implemented as a special feature related to prescribed displacements, see Section 3.4. Hence, there is no assumption made regarding symmetry and the RAO's need to be given for all directions. The RAO directions are assumed to be defined relative to a local vessel system with heading according to the **CONSTR** card, see Section 3.4. In order to allow vessel eccentricities to be introduced in the vessel coordinate system it is convenient to use element type SPRING137 and define separate nodes for the vessel that are orientated by the EULERANGLE concept, see Section 3.39.

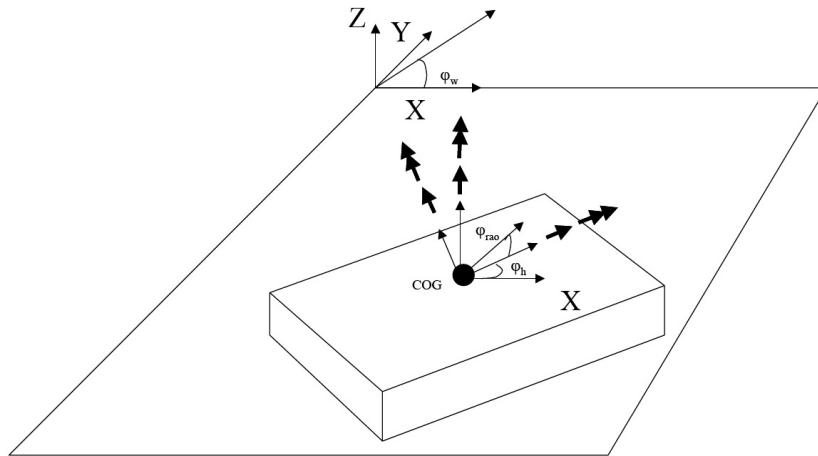


Figure 3.38: Vessel RAO coordinate system in relation to global system.

For a given wave frequency the response quantity is calculated according to the following procedure:

1. Calculate the angle  $\varphi_{rao} = \varphi_w - \varphi_h$  which is the wave heading angle in the rao vessel system, see Fig. 3.38, and Section 3.4.  $\varphi_w$  is the angle between the wave direction and the global x-axis.  $\varphi_h$  is the vessel heading angle relative to the global x-axis corresponding to the angle **VHEAD** in Section 3.4.1.

2. If necessary interpolate for each response direction defined in the **CONSTR PDISP** card (surge, pitch etc.) to find the relevant rao values defined by curves for each direction angle number i, i.e. the unit response amplitude and associated phase as:

$$\beta_{rao} = f_{amp}(\omega, a_i)(1 - \xi) + f_{amp}(\omega, a_{i+1})\xi \quad (3.49)$$

$$\phi_{rao} = f_{phase}(\omega, \phi_i)(1 - \xi) + f_{phase}(\omega, \phi_{i+1})\xi \quad (3.50)$$

where  $\xi$  is a non-dimensional interpolation scaling parameter =  $\frac{\varphi_{rao} - \varphi_i}{\varphi_{i+1} - \varphi_i}$ .

3. The response is then calculated as:

$$r(x, y, t) = \frac{H}{2} \sin(\omega t - k(x - x_0) \cos \varphi_w - k(y - y_0) \sin \varphi_w + \phi_{rao}) \quad (3.51)$$

Note that if the vessel need to shift heading during the analysis, the procedure will be as follows:

1. Prescribe the vessel heading by introducing **LOCAL** prescribed rotation about the vessel node z-axis, see Section 3.4.1.
2. Replace the **LOCAL** prescribed displacement above with an RAO prescribed displacement, see Section 3.4.1. The userdefined heading angle **VHEAD** in Section 3.4.1 must be set equal to the new heading angle obtained after the rotation in item 1.

#### EXAMPLE:

---

Example of DEF type RAOPROP:

```
#      name   type fi1 fi2 fi3 fi4 fi5   curve1  curve2  curve3  curve4  curve5
#
RAOPROP surge DEF  0   1.57 3.14 4.71 6.28  surge1   surge2   surge3   surge4   surge5
RAOPROP sway  DEF  0   1.57 3.14 4.71 6.28  sway1    sway2    sway3    sway4    sway5
RAOPROP heave DEF  0   1.57 3.14 4.71 6.28  heave1   heave2   heave3   heave4   heave5
RAOPROP roll   DEF  0   1.57 3.14 4.71 6.28  roll1    roll2    roll3    roll4    roll5
RAOPROP pitch  DEF  0   1.57 3.14 4.71 6.28  pitch1   pitch2   pitch3   pitch4   pitch5
RAOPROP yaw    DEF  0   1.57 3.14 4.71 6.28  yaw1     yaw2     yaw3     yaw4     yaw5
#
```

Example of CURVE type RAOPROP:

```
#      curvnam type      circfreq (rad)   amplitude (per m wave ampl)  phase (rad)
#
RAOPROP surge1  CURVE  0                  0                      0.0000000000
                                         0.314159265          1.1908                3.129375349
```

0.34906585	1.0798	3.127630020
0.369599136	1.0253	3.127630020
0.392699082	0.9693	3.124139361
0.41887902	0.909	3.122394032

### 3.45 READTRF - READ TRAnsfer function data

The **READTRF** card enables the user to read transfer functions from a separate file. The format is as follows:

**READTRF NAME TYPE [PAR1]**

where

**NAME**: Name of file with transfer function data.

**TYPE**: File type.

**SIM** : : SIMLA format, transfer function is defined on separate file using the RAOPROP card.

**WAM** : : \*.4 result file from WAMIT

**MOS** : : Transfer function file in MOSES format

**PAR1**: Additional parameter to describe imported transfer functions.

For **TYPE = WAM**, **PAR1** is the characteristic length used in WAMIT.

For **TYPE = MOS**, **PAR1** is number of periods used in imported transfer function.

*Note:* When WAMIT data is imported, the transfer functions must be defined as a function of the period.

*Note:* When transfer functions in MOSES-format are imported, the local x-axis of the node where the RAO is applied should point towards the bow of the vessel and local z-axis should be upwards.

The read transfer function will be printed to an ASCII file named '\*-sim.rao' in the analysis folder. This file contains the transfer function defined according to the **RAOPROP** input card in Section 3.44.

The read transfer functions will be assigned the names **SURGE**, **SWAY**, **HEAVE**, **ROLL**, **PITCH** and **YAW** by SIMLA. These names must be referred to by the character string **RAONAME** in Section 3.4.1, as shown in the examples below.

EXAMPLE: \_\_\_\_\_

```

#      filename   type [par]
READTRF vessel.rao  SIM
#
#      pdisp  pdtype  nodid  dof  vhead  waveno  raoname
CONSTR PDISP  RAO    5001   1   0.785  100    SURGE
CONSTR PDISP  RAO    5001   2   0.785  100    SWAY
CONSTR PDISP  RAO    5001   3   0.785  100    HEAVE
CONSTR PDISP  RAO    5001   4   0.785  100    ROLL
CONSTR PDISP  RAO    5001   5   0.785  100    PITCH
CONSTR PDISP  RAO    5001   6   0.785  100    YAW
#

```

---

EXAMPLE: \_\_\_\_\_

```

#      filename   type [par]
READTRF octopus.4  WAM    1.0
#
#      pdisp  pdtype  nodid  dof  vhead  waveno  raoname
CONSTR PDISP  RAO    5001   1   0.785  100    SURGE
CONSTR PDISP  RAO    5001   2   0.785  100    SWAY
CONSTR PDISP  RAO    5001   3   0.785  100    HEAVE
CONSTR PDISP  RAO    5001   4   0.785  100    ROLL
CONSTR PDISP  RAO    5001   5   0.785  100    PITCH
CONSTR PDISP  RAO    5001   6   0.785  100    YAW
#

```

---

EXAMPLE: \_\_\_\_\_

```

#      filename   type [par]
READTRF vessel-rao.txt  MOS    35
#
#      pdisp  pdtype  nodid  dof  vhead  waveno  raoname
CONSTR PDISP  RAO    5001   1   0.785  100    SURGE
CONSTR PDISP  RAO    5001   2   0.785  100    SWAY
CONSTR PDISP  RAO    5001   3   0.785  100    HEAVE
CONSTR PDISP  RAO    5001   4   0.785  100    ROLL
CONSTR PDISP  RAO    5001   5   0.785  100    PITCH
CONSTR PDISP  RAO    5001   6   0.785  100    YAW
#

```

---

### 3.46 REEL - reeling and straightening simulation scenario definition

The **REEL** command is used to define the components needed in a reeling/straightening simulation. In addition to the pipe intended for reeling such an analysis must contain a spool and possible rolls to model a straightener.

Master-Slave technique is utilized in the modelling of reeling and hence constraint equations between the different components must be defined in addition to the **REEL**-command. This is done by the **CONSTR CONEQ**-command, see Sections 3.4 and 3.4.2.

The principle of the constraint formulation is illustrated in (a) in Fig. 3.39. The pipe group intended for reeling contains only master nodes, except for the first node, which is a slave of the spool axis. Each of the master nodes in the pipe has a corresponding slave node in the spool. This slave follows the rotation of the master in the spool and is always located “under” its master in the pipe. The local coordinate system is always normal to the spool as indicated in the figure. When contact between a pipe node and the spool occurs the displacement normal to the spool is locked to the slave node. In order to monitor the contact force and have a criterion for disconnecting, the slave in the spool is connected to another slave by a stiff element. The second slave node of this “contact element” is locked to a spring (ordinary beam-element), as indicated in (b) in Fig. 3.39. The contact force for each point can then be found as the shear force (local y-direction) of the beam. For a given negative contact force limit (or zero) the master node in the pipe disconnects from the slave node and the slave node connects to the master of the spring. The location of the slave node underneath the master node in the pipe is governed by a constraint equation between the slave node of the “spring” and the master node of the spool axis.

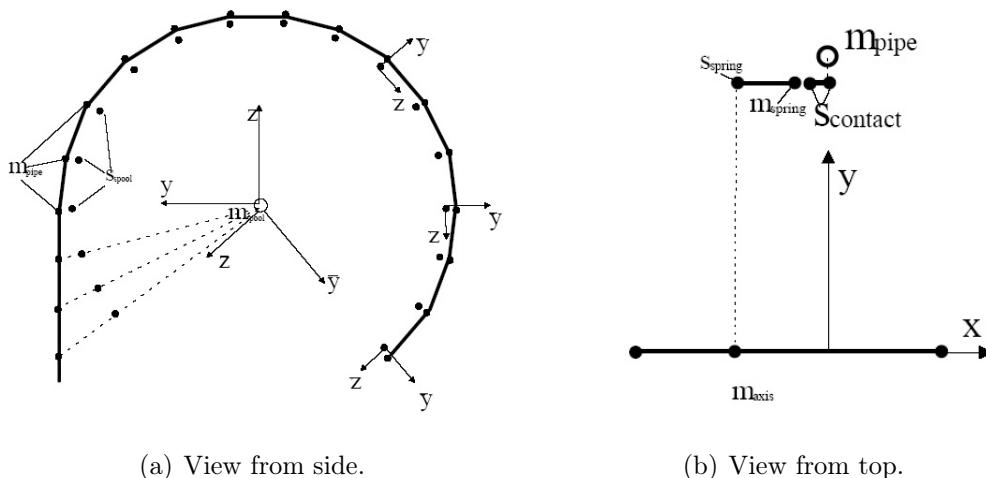


Figure 3.39: Reeling.

The principle for the rolls is similar, but in this case only one master node in the pipe connects to one slave node in the roll. Hence only one “slave-element” needs to be defined, but all nodes in the pipe must be masters to this slave. The spring element may be the roll itself.

It should be noted that all constraint equations related to contact between pipe and spool/rolls are applied in the local coordinate system of the slave.

The **REEL** command has the following format:

```
REEL    REELGRP    ROPIPE    SPOOLSPRGRP    SPOOLCOGRP    RSPOOL
FCLIMSPPOOL [ROLLGRP ROLLCOGRP RROLL FCLIMROLL]
```

where:

**REELGRP**: Name of the pipe group intended for REELING.

**ROPIPE**: Outer radius of the reeled pipe.

**SPOOLSPRGRP**: Name of group containing “spring elements” in spool.

**SPOOLCOGRP**: Name of group containing “contact elements” in spool.

**RSPOOL**: Radius of spool.

**FCLIMSPPOOL**: Contact-force cut-off limit for spool.

**ROLLGRP**: Name of group for roll.

**ROLLCOGRP**: Name of group for “contact element” in roll.

**RROLL**: Radius of roll.

**FCLIMROLL**: Contact-force cut-off limit for spool.

The sequence **ROLLGRP ROLLCOGRP RROLL FCLIMROLL** can be repeated to model multiple rolls.

EXAMPLE: \_\_\_\_\_

```
#  
#   Reeling pipe:  
# Pipe ID      Radius  
REEL    centerpipe  0.25  
#  
#       Spool:  
#     Spr ID      Cont ID      Radius      fclim  
#           spring1    slave1      7.5        0.0  
#  
#       Straightner:  
#     Spr ID      Cont ID      Radius      fclim  
#           roll11     droll1     0.5        0.0  
#           roll12     droll12    0.5        0.0  
#           roll13     droll13    0.5        0.0
```

---

```
roll14      droll14      0.5      0.0
```

---

### 3.47 SEALO - Sea load specification

The **SEALO** command allows the user to define current loading that depends on the position relative to a curvilinear path in the SIMLA coordinate system. The following format is applied:

```
SEALO SEAGR X1 Y1 X2 Y2 CURRNO THIST
      ...   ...   ...   ...   ...       ..
```

where:

**SEAGR**: Sea element group name.

**X1**: x-coordinate point 1 (unit: L).

**Y1**: y-coordinate point 1 (unit: L).

**X2**: x-coordinate point 2 (unit: L).

**Y2**: y-coordinate point 2 (unit: L).

**CURRNO**: Current profile number.

**THIST**: Current load history number.

An arbitrary sequence of current loads may be given by repeating **X1 Y1 X2 Y2 CURRNO THIST**. The curvilinear path consists of piecewise straight segments spanning between (**X1**, **Y1**) and (**X2**, **Y2**). The direction change between adjacent straight segments must be small. This is because only pipe elements located perpendicular to the straight segments are included in the search algorithm used for assigning the sea load data.

EXAMPLE:

---

```
#  
# Current and wave loads:  
#-----  
#      groupname    x1      y1      x2      y2      icur      ihist  
SEALO  sea1      -4000     0       0       0      100      400  
          0       0    24000     0      200      500
```

---

### 3.48 SIMLA - lay simulation scenario definition

The **SIMLA** card is part of the lay scenario definition. In addition, the relevant analysis type must be selected in the **TIMECO** card, and for some types of analysis one or more **FEED** cards are also required.

The **SIMLA** card contains information about lay rate and which criterion the vessel should be steered according to. This may be departure angle of pipe, distance to roller, tension at TDP, layback distance or position of pipe relative to tower position. In addition to the steering criteria a tolerance is given. This parameter controls the strictness of the steering criteria used. Full correction of the vessel will be given when the deviation is equal to the tolerance.

For lay analysis of types other than **SIMLA-FEED** (see Section 3.51) the element length of the pipe element group must be constant, at least for the **NELPST** elements closest to the tail. See the **NOCOOR** command. Prescribed displacements must to be defined in the longitudinal and transverse directions of both the vessel and bottom end nodes, see the **CONSTR** command Section 3.4. In some cases a concentrated load is required for the vessel in longitudinal direction.

The lay simulation scenario is defined along the curvilinear s-coordinates (established from  $\sum \sqrt{dx^2 + dy^2} = KP$ ) of the contact surface, ie. the given route. When defining the **SIMLA** data group, the constraint coefficients in the longitudinal and transverse directions are automatically updated for vessel and tail of pipe, in order to follow the route and fulfil the steering criterion.

The following format is applied:

```
SIMLA IVESSEL NPIPE PIPEGRP1 SEABDGRP1
      [...          ..
      PIPEGRP2 SEABEDGRP2]
FILE NODES SIGF UTIL TYPE ....
```

where:

**IVESSEL**: Node ID number for prescribed vessel motions.

**NPIPE**: Number of pipe groups in pipe to be layed.

**PIPEGRP1**: Name of pipe group number 1, beeing the group starting at the pipe tail.

**SEABDGRP1**: Name of seabed group number 1, beeing the seabed contact group of **PIPGRP1**.

**PIPEGRP2**: Name of the last pipe group, which is group number **NPIPE**.

**SEABDGRP2**: Name of the last seabed group, beeing the seabed contact group of **PIPGRP2**.

**FILE:** File name at which soil reaction forces at the TDP is to be stored. Coordinate , Force vector and TDP tension is written to file. .

**NODES:** Number of nodal points from and including tdp, where the soil reaction forces are to be stored (only points in contact)

**SIGF:** Yield stress in steel material (unit: F/L<sup>2</sup>).

**UTIL:** Maximum allowable sagbend utilization, if this level is exceeded the steering parameter will be adjusted (only for J and S type in present version).

**TYPE:** Type of steering, where one of the following can be selected:

**J :** Steering to obtain constant departure angle. The pipe is hinged at the top.

**S :** Steering to obtain constant distance from a specified roll. The pipe is fixed at the top.

**TENSION :** Steering to obtain constant tension at TDP.

**LAYBACK :** Steering to obtain constant layback.

**T :** Steering to maintain nominal position relative to tower.

For **TYPE=J** (J-lay), the following format is applied:

```
SIMLA IVESSEL NPIPE PIPEGRP1 SEABDGRP1
[...          ..
PIPEGRP2 SEABEDGRP2]
FILE NODES SIGF UTIL J NELPST DEPANG TOL
```

**NELPST:** Number of elements per step. The model will then in each step be moved along the route a distance corresponding to a feed of **NELPST** elements.

**DEPANG:** Required departure angle.

**TOL:** Tolerance.

For **TYPE=S** (S-lay), the following format is applied:

```
SIMLA IVESSEL NPIPE PIPEGRP1 SEABDGRP1
[...          ..
PIPEGRP2 SEABEDGRP2]
FILE NODES SIGF UTIL S NELPST ROLLDIST TOL STINGERGRP
```

**NELPST:** Number of elements per step. The model will then in each step be moved along the route a distance corresponding to a feed of **NELPST** elements.

**ROLLDIST:** Required distance to roller.

**TOL:** Tolerance.

**STINGERGRP:** Name of stinger group.

For **TYPE=TENSION**, the following format is applied:

```
SIMLA IVESSEL NPIPE PIPEGRP1 SEABDGRP1
[...      ..
PIPEGRP2 SEABEDGRP2]
FILE NODES SIGF UTIL TENSION NELPST TENS TOL [THIST]
```

**NELPST**: Number of elements per step. The model will then in each step be moved along the route a distance corresponding to a feed of **NELPST** elements.

**TENS**: Required tension at TDP.

**TOL**: Tolerance.

**THIST**: Time history, optional.

For **TYPE=LAYBACK**, the following format is applied:

```
SIMLA IVESSEL NPIPE PIPEGRP1 SEABDGRP1
[...      ..
PIPEGRP2 SEABEDGRP2]
FILE NODES SIGF UTIL LAYBACK NELPST LAYBK TOL [THIST]
```

**NELPST**: Number of elements per step. The model will then in each step be moved along the route a distance corresponding to a feed of **NELPST** elements.

**LAYBK**: Required layback.

**TOL**: Tolerance.

**THIST**: Time history, optional.

For **TYPE=T**, (Tower steering), the following format is applied:

```
SIMLA IVESSEL NPIPE PIPEGRP1 SEABDGRP1
[...      ..
PIPEGRP2 SEABEDGRP2]
FILE NODES SIGF UTIL T NELPST TLENGTH OFFSET TOL
```

**NELPST**: Number of elements per step. The model will then in each step be moved along the route a distance corresponding to a feed of **NELPST** elements.

**TLENGTH**: Distance from top of tower(tensioner exit) to steering position of pipe.

**OFFSET**: Offset relative to nominell position. Positive values correspond to larger tension in pipe, negative values to lower tension in pipe.

**TOL**: Tolerance.

NOTE: **NELPST** may be negative (retrieval) when a **STATIC-FEED** type analysis is carried out.

## EXAMPLE:

---

```
#      ivessel   npipe  pipegrp1    seabdgrp1
SIMLA    7001      1     umbilical1  seabed
#
#      file          nodes
"supp-lay-static.txt"      5
#
#      sigf       util   type        nelpst  TDPT   tol
450000  1.0     TENSION      5        7.0    1.0
```

---

**3.49 TABLE - Table data**

The table command is used for defining data arrays. The array must be rectangular and must contain at least two rows and two columns. Tables are for instance used when defining DROPS hydrodynamic properties for pipe elements in Section 3.14, and for initial soil embedment in Section 3.23.11.

The format is as follows:

<b>TABLE NAME NCOL</b>
<b>T11 T12 .. T1_NCOL</b>
..
<b>TN1 TN2 .. TN_NCOL</b>

where

**NAME:** Name of the table.

**NCOL:** Number of columns in table. Must be larger than or equal to 2.

**T11:** Value of entry (1,1), first argument value

**T12:** Value of entry (1,2), first function value of column 2.

**T1\_NCOL:** Value of entry (1,ncol), first function value of last column.

**TIJ:** Value of entry (i,j).

**TN1:** Value of entry (n,1), last argument value.

**TN2:** Value of entry (n,2), last function value of column 2.

**TN\_NCOL:** Value of entry (n,ncol), last function value of last column.

The first column defines the argument values, and the other columns define the function values. Linear interpolation of the function values are applied in between the argument values. If the argument is less than **T11**, the first function value of the column(s) will

be applied. If the argument is larger than **TN1**, the last function value of the column(s) will be applied.

EXAMPLE:

---

```
#-----
#      name      ncol
#-----
TABLE  masscoef   2
#
# gap  cm
#
0.0    3.28
0.4    2.0
20.0   2.0
#-----
#
#-----
#      name      ncol
#-----
TABLE  loadcoef   4
#
# arg  f1     f2     f3
#
0.0    3.2    1.2    0.7
1.0    2.0    1.0    0.6
20.0   2.0    1.0    0.5
#-----
```

---

### 3.50 THIST - time histories

By the **THIST** command the user is allowed to describe the load histories related to the different loads specified. The following format is applied:

```
THIST NO T1 FAC1
          T2 FAC2
          .. ..
```

where

**NO**: History number.

**T1**: Time t1.

**FAC1**: Load factor for time t1.

**T2**: Time t2.

**FAC2:** Load factor for time t2

An arbitrary sequence may be given, and linear interpolation is applied between the time steps defined by the **TIMECO** card above. For  $t \in [T1, T2]$  the load factor will then be:

$$F = \frac{\text{FAC2}(t - T1) + \text{FAC1}(T2 - t)}{T2 - T1} \quad (3.52)$$

The load factor is multiplied with the load level given in the relevant load definition.

### 3.50.1 THIST\_F - history from file

A special version of **THIST** can be applied to obtain user defined load histories from file. The following format applies:

**THIST\_F NO TFILE**

where

**NO:** History number.

**TFILE:** file name containing the parameters t1,fac1,t2,fac2...,defined above

### 3.50.2 THIST\_H - harmonic load

A special version of **THIST** can be applied to obtain harmonic loads. The following format applies:

**THIST\_H NO OMEGA PHASE**

where

**NO:** History number.

**OMEGA:** Frequency,  $\omega$ .

**PHASE:** Phase displacement,  $\varphi$ .

The load factor at any time  $t$ , is then defined by:

$$F = \sin(\omega t + \varphi) \quad (3.53)$$

### 3.50.3 THIST\_R - ramping loads smoothly on or off

A special version of **THIST** can be applied if loads should be ramped on or off in a smooth manner. This is in particular recommended for wave loads. The following format applies:

**THIST\_R NO START STOP RAMPTYPE FAC**

.. .. .. ..

**NO:** History number.

**START:** Start time for ramping.

**STOP:** Stop time for ramping.

**RAMPTYPE:** Type of ramping, only **RAMPCOS** allowed.

**FAC:** Load factor after the ramping.

The shape of the ramping function is a half cosine period, starting at zero slope. Between ramping periods, the load factor is kept constant at the value obtained in the last ramping period. Before first ramping, the load factor is zero.

EXAMPLE: \_\_\_\_\_

```

#           weight and external pressure
#
#       no      time    fac empty pipe = 0.421
THIST   100      0      0.4210
          1      0.4210
          1.05    0.4210
          2      1.0000
          3      1.0000
          4      0.4210
          5      0.5376
         32.5    0.5376
#
#           temperature
#       no      time    fac
THIST   200      0      0.0
          5      0.0
          25     131.9
         32.5    131.9
#
#           no      filename
THIST_F 100     file100
#:
#
#           no      omega   phase
THIST_H 100     0.628   0.0
#
#
#           no      start   stop    Ramptype   fac
THIST_R 100     10      20      RAMPCOS    1.0
          30      40      RAMPCOS    0.0

```

### 3.51 TIMECO - time control data

By the **TIMECO** command the user defines the analysis as a function of time. The format is as follows:

```
TIMECO T DT DTVI DTDY DT0 TYPE HLAFLAG [STEPTYPE ITERCO ITCRIT  
MAXIT MAXDIV CONR]
```

where:

**T**: Time to simulate to.

**DT**: Time increment to be used to reach the required time.

**DTV**: Time increment between each restart/visual storage to the .raf file.

**DTDY**: Time increment between each storage to the .ldat file.

**DT0**: Time increment between each zero setting of the accumulated convergence control vectors.

**TYPE**: The analysis type which may be **STATIC**, **DYNAMIC**, **STATIC-SIMLA**, **STATIC-SDD** or **STATIC-FEED**.

**HLAFLAG**: HLA control parameter which may be **HLA** or **NOHLA**.

**STEPTYPE**: Type of step control, optional parameter If **STEPTYPE** is given, **ITERCO**, **ITCRIT**, **MAXIT**, **MAXDIV** and **CONR** must also be given. **STEPTYPE** can have two different values:

**MANUAL** : Time stepping is carried out as normal, but given **MAXIT** overrules the one given in the **CONTROL** card.

**AUTO** : If required tolerance is not met within **MAXIT**, the step is subdivided in two parts. This is repeated until required tolerance is met within **MAXIT** or the maximum number of allowed sub-divisions **MAXDIV** is reached. SIMLA will proceed with the new, smaller time step until next ordinary time step is reached. If, during the sub-stepping, the tolerance is not met the time step will be split and iteration start over from the last ordinary time step.

**ITERCO**: Iteration control parameter. Must be given if **STEPTYPE** is given.

**NONE** : usual

**GO-ON** : the program proceeds after the maximum number of iterations is reached

**ITCRIT**: Iteration criterion parameter, enables the user to select between alternative incremental norms scaled with respect to the associated accumulated norms. Must be given if **STEPTYPE** is given.

**DISP** : displacement norm is used

**FORC** : force norm is used

**ENER** : energy norm is used

**ALL** : all norms are used

**MAXIT**: Maximum number of iterations. Must be given if **STEPTYPE** is given.

**MAXDIV**: Maximum number of sub-divisions. Must be given if **STEPTYPE** is given.

**CONR**: Convergence radius. Recommended value is  $10^{-5}$ – $10^{-7}$ . Overrules the global convergence radius from the **CONTROL** card in Section 3.6. Must be given if **STEPTYPE** is given.

EXAMPLE: \_\_\_\_\_

```

#
#-----
# Analysis time control:
#-----
#      t      dt      dtvi dtdy  dt0      type    hla?
TIMECO  1      1      1.0  1.0   201.0  STATIC  NOHLA
#
#      Waterfilling
#
#      t      dt      dtvi dtdy  dt0      type    hla?
TIMECO  2      0.05  1.0  1.0   201.0  STATIC  NOHLA
#
#      Hydrotest
#
#      t      dt      dtvi dtdy  dt0      type    hla?
TIMECO  3      0.05  1.0  1.0   201.0  STATIC  NOHLA
#
#      Dewatering
#
#      t      dt      dtvi dtdy  dt0      type    hla?
TIMECO  4      0.05  1.0  1.0   201.0  STATIC  NOHLA
#
#      Gas filling and rock dumping
#
#      t      dt      dtvi dtdy  dt0      type    hla?
TIMECO  5      0.05  1.0  1.0   201.0  STATIC  NOHLA
#
#      Operation
#
#      t      dt      dtvi dtdy  dt0      type    hla?
TIMECO  29.5  0.1   0.5  1.0   201.0  STATIC  NOHLA
#
#      Operation + trawling
#
#      t      dt      dtvi dtdy  dt0      type    hla?
TIMECO  32.5  0.1   0.5  1.0   201.0  DYNAMIC NOHLA

```

#

---

### 3.52 TLOAD - temperature loading

The temperature loading is specified by the following format:

<b>TLOAD HIST ELNR1 T1 [ELNR2 T2]</b>
---------------------------------------

where

**HIST**: Load history number.

**ELNR1**: First element ID number.

**T1**: Temperature of first element (unit: TE).

**ELNR2**: Last element ID number.

**T2**: Temperature of last element (unit: TE). Linear interpolation is applied for the intermediate elements.

EXAMPLE:

---

```
#          (temperature loading)
#      loadhistnr  elnr1   t1      elnr2   t2
TLOAD    300        16001  100.0  16280  100.0
```

---

### 3.53 UNITS - SIMVIS unit conversion factors

This card is used to specify SIMVIS unit conversion factors relative to the units applied in the analysis. If this card is given, HLA-results and plots will be written with appropriate units and range. If not given, HLA results will be corresponding to model units used. The card does not affect the calculations, only scales and labels for the HLA plots available in SIMVIS will be affected.

<b>UNITS UMASS ULENGTH UTIME</b>
----------------------------------

where:

**UMASS**: Mass unit relative to kg.

**ULENGTH**: Length unit relative to m.

**UTIME**: Time unit relative to sec.

For an analysis where kN, m and s have been used for forces, lengths and times respectively, the card would be as follows:

EXAMPLE: \_\_\_\_\_

```
# mass      length   time
UNITS    1.0e-3   1.0     1.0
#
```

---

### 3.54 VISRES - visual results

By including the **VISRES** card, visual presentation in XPOST is enabled. Only one **VISRES** card can be given, and all results can be listed in that card. If no **VISRES** card is present, only restart info will be stored on the .raf file. However, SIMPost can still be applied.

Most structural elements are visualized as tubulars, whereas most contact elements are visualized as lines.

However, CONT126 additionally includes shell elements to visualize the sea bottom. HSHEAR342 is further visualized as a rectangle.

The format is as follows:

**VISRES MODE FACTOR RESULT [...]**

where

**MODE**: Result presentation mode:

**INTEGRATION** : FEM results according to the list defined below is enabled including consistent representation of all numerical elements of the FEM model in XPOST including the applied element and node numbers. In addition to the list below finite element resultant force results of all pipe elements is allowed for. Other options are deprecated, but this value are kept for backwards compatibility.

**FACTOR**: Scaling factor for all radial quantities (scale pipe radius).

**RESULT**: Result types that will be stored, can be a list. Options are:

**SIGMA-XX** :  $\sigma_{xx}$  Longitudinal stress (cable elements) at mean surface (pipe elements).

**SIGMA-YY** :  $\sigma_{yy}$  Hoop stress at mean surface- thinwalled theory (pipe elements).

**SIGMA-XY** :  $\sigma_{xy}$  Shear stress at mean surface (pipe elements).

**STRAIN-XX** :  $\varepsilon_{xx}$  Longitudinal strain at mean surface - (pipe elements).

**STRAIN-YY** :  $\varepsilon_{yy}$  Hoop strain at mean surface (pipe elements).

**STRAIN-XY** :  $\varepsilon_{xy}$  Shear strain at mean surface (pipe elements).

**ACCSTRAIN** :  $\varepsilon_{tot}$  Accumulated (equivalent) plastic strain (elastoplastic pipe elements).

**SIGMA-XXI** :  $\sigma_{xxi}$  Longitudinal stress at inner surface (elastic pipe elements).

**SIGMA-YYI** :  $\sigma_{yyi}$  Hoop stress at inner surface-thickwalled theory (elastic pipe elements).

**SIGMA-XYI** :  $\sigma_{xyi}$  Shear stress at inner surface (elastic pipe elements).

**SIGMA-XXO** :  $\sigma_{xxo}$  Longitudinal stress at outer surface (elastic pipe elements).

**SIGMA-YYO** :  $\sigma_{yyo}$  Hoop stress at outer surface-thickwalled theory (elastic pipe elements).

**SIGMA-XYO** :  $\sigma_{xyo}$  Shear stress at outer surface (elastic pipe elements).

**VCONDIS-X** : Contact element local x-displacement.

**VCONDIS-Y** : Contact element local y-displacement.

**VCONDIS-Z** : Contact element local z-displacement.

**VCONFOR-X** : Contact element local x-force (unit: See Table 3.1)

**VCONFOR-Y** : Contact element local y-force (unit: See Table 3.1)

**VCONFOR-Z** : Contact element local z-force (unit: See Table 3.1)

EXAMPLE:

---

#	type	scaling factor	result list
VISRES	integration	1	sigma-xx sigma-yy

---

### 3.55 WAVELO - Regular and irregular wave loading

The waveload applies to the cable and pipe element types, assuming a circular cross-section. It is also noted that even in cases with no waveloads, the waveload card is needed to activate the relative velocity kinematics (drag forces). The hydrodynamic forces are calculated as follows with reference to the local element coordinate frame.

$$\begin{aligned}
 F_1 &= C_{M,t}\rho \frac{\pi D^2}{4} \dot{u}_1 - (C_{M,t} - 1)\rho \frac{\pi D^2}{4} \ddot{x}_2 + \frac{1}{2}\rho DC_{D,t}|v_1|v_1 \\
 F_2 &= C_{M,n}\rho \frac{\pi D^2}{4} \dot{u}_2 - (C_{M,n} - 1)\rho \frac{\pi D^2}{4} \ddot{x}_2 + \frac{1}{2}\rho DC_{D,n}|\mathbf{v}_n|v_2 \\
 F_3 &= C_{M,n}\rho \frac{\pi D^2}{4} \dot{u}_3 - (C_{M,n} - 1)\rho \frac{\pi D^2}{4} \ddot{x}_3 + \frac{1}{2}\rho DC_{D,n}|\mathbf{v}_n|v_3
 \end{aligned} \tag{3.54}$$

The local coordinate system and velocity vectors are shown in Fig. 3.40, and the terms in the above equation are described below:

- $C_{M,t}$ : Inertia coefficient (tangential to pipe) from Morison's equation.  $C_{M,n}$ : Inertia coefficient (normal to pipe) from Morison's equation.
- $C_{D,t}$ : Tangential drag coefficient from Morison's equation.  $C_{D,n}$ : Normal drag coefficient from Morison's equation.
- $C_v$ : Dimensionless coefficient representing the magnitude of the vortex shedding force.
- $\dot{u}_i$ : Water particle acceleration in direction i to the cylinder.
- $\ddot{x}_n$ : Cylinder acceleration normal to the cylinder.
- $\mathbf{v}_n$ : Relative flow velocity normal to the cylinder, see Fig. 3.40. The relative velocity is found as  $\mathbf{v} = \mathbf{u} - \dot{\mathbf{x}}$  in the two normal directions (2 and 3).  $v_1$ : Relative flow velocity in the tangential direction of the cylinder, see Fig. 3.40. The relative velocity is found as  $v_1 = u_1 - \dot{x}_1$  in the tangential direction.

*Note:* Hydrodynamic forces are not applied for pipe nodes that are connected to the CONT128 element type. These nodes are assumed to be buried into the seabed and should consequently not be subjected to hydrodynamic forces.

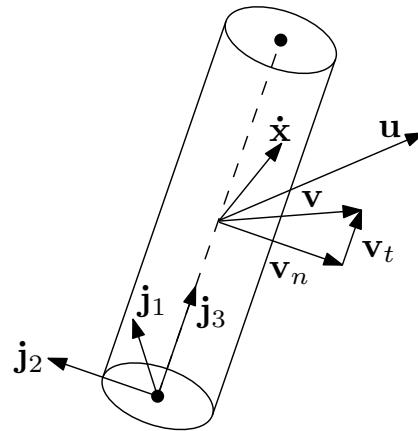


Figure 3.40: A cylinder segment with local coordinate system and velocity vectors.

The wave loading is assumed to occur as a result of a wave generator positioned at a point  $x_0, y_0$  in the xyz global coordinate frame. The following format is applied:

### 3.55.1 General

**WAVELO ELGRP WAVETYPE ...**

where the following definitions and units applies:

**ELGRP**: Element group name.

**WAVETYPE**: Wave type. Possible values are **REGULAR** and **IRREGULAR**

### 3.55.2 Regular wave loading, WAVETYPE=REGULAR

**WAVELO ELGRP REGULAR WAVENO WAVEHIST X0 Y0 ANG T H D PHI**

where the following definitions and units applies:

**WAVENO:** Wave load number.

**WAVEHIST:** Wave history number.

**X0:** Global x-coordinate of the wave elevation origin (unit: L).

**Y0:** Global y-coordinate of the wave elevation origin (unit: L).

**ANG:** Wave direction angle,  $\varphi_w$ , relative to the global coordinate system x-axis and counterclockwise, see Fig. 3.1. (unit: R)

**T:** Wave period (T).

**H:** Wave height, double amplitude (L).

**D:** Water depth (L).

**PHI:** Phase angle  $\phi$  (unit: R).

The wave elevation at any point x,y along the sea surface is calculated as:

$$\eta(x, y, t) = \frac{H}{2} \sin(\omega t - k(x - x_0) \cos(\varphi_w) - k(y - y_0) \sin(\varphi_w) + \phi) \quad (3.55)$$

where  $k = 2\pi/L_w$  is the wave number and  $\omega = 2\pi/T$  is the wave circle frequency. The above means that the waves propagate along the positive x-axis at direction angle  $\omega_w = 0$ .

EXAMPLE:

---

```
#  
#      seagrp   type      wavno   hist    x0          y0   phi     T    H    D   Phase  
WAVELO  sea1    REGULAR   100     250   1667.270   0   0.000  10   7.0  2200   0  
#
```

---

### 3.55.3 Irregular wave loading, WAVETYPE=IRREGULAR

**WAVELO ELGRP IRREGULAR WAVENO WAVEHIST X0 Y0 ANG TP HS D  
DT TDUR T0 DKIN SEED TYPE [PKDNESS]**

where the following definitions and units applies

**WAVENO:** Wave load number.

**WAVEHIST:** Wave history number.

**X0:** Global x-coordinate of the wave elevation origin (unit: L).

**Y0:** Global y-coordinate of the wave elevation origin (unit: L).

**ANG:** Wave direction angle,  $\varphi_w$ , relative to the global coordinate system x-axis and counterclockwise, see Fig. 3.1. (unit: R)

**TP:** Peak period,  $T_p$ , (unit: T).

**HS:** Significant wave height,  $H_s$ , (unit: L).

**D:** Water depth applied for generation of the wave kinematics, (unit: L).

**DT:** Time step used for generation of wave kinematics,  $dt$ , (unit: T)

**TDUR:** Duration of wave kinematics time series,  $T_{dur}$ , ( unit: T)

**T0:** Starting point in time for the wave kinematics time series,  $T_{start}$  , ( unit: T)

**DKIN:** Kinematic water depth, nodes located below this depth at **T0** will be assigned zero wave kinematics, ( unit: L).

**SEED:** Seed, *i.e.* the starting integer value of the generation of the random phase angles of the wave kinematics time series.

**TYPE:** Wave spectrum type. Implemented: **1** = two parameter Pierson Moskowitz spectrum **2** = three parameter Jonswap spectrum

**PKDNESS:** Peakedness parameter, only for Jonswap spectrum.

The wave elevation is expressed as:

$$\eta(t, x, y) = \sum_{k=1}^{N_\omega} A_k \sin(\omega_k t + \phi_k^p + \phi_k) \quad (3.56)$$

Where  $\phi_k$  is the random phase angle sampled from a uniform distribution over  $[-\pi, \pi]$ .  $\phi_k^p$  is the position dependent phase angle and  $A_k$  is the wave component amplitude given as

$$A_k = \sqrt{2S_\eta(\omega)\Delta\omega} \quad (3.57)$$

The two-parameter Pierson Moskowitz spectrum is defined as

$$\begin{aligned} S_\eta(\omega) &= A\omega^{-5}e^{-\frac{B}{\omega^4}}, 0 < \omega < \infty \\ A &= 124.2 \frac{H_s}{T_z^4} \\ B &= \frac{496}{T_z^4} \end{aligned} \quad (3.58)$$

where  $H_s$  is the significant wave height and  $T_z$  is the zero-up crossing frequency. The relation between the zero-up crossing frequency and the peak period  $T_p$  is  $T_p = 1.408T_z$

The three parameter Jonswap spectrum is given as

$$\begin{aligned}
 S_\eta(\omega) &= \alpha g^2 \omega^{-5} e^{-\beta \left(\frac{\omega_p}{\omega}\right)^4} \gamma e^{-\frac{(\omega - \omega_p)^2}{2\sigma^2 \omega_p^2}} \\
 \alpha &= 1.2905 \frac{H_s^2}{T_z^2} \\
 \beta &= 1.205 \text{for North Sea Conditions} \\
 &= 1 \text{where } T_p \geq 5\sqrt{H_s} \\
 \gamma &= e^{5.75 - 1.15 \frac{T_p}{\sqrt{H_s}}} \\
 &= 5.0 \text{where } T_p < 3.6\sqrt{H_s} \\
 \sigma &= 0.07 \text{for } \omega \leq \omega_p \\
 &= 0.09 \text{for } \omega \geq \omega_p \\
 \omega_p &= \frac{2\pi}{T_p} \\
 \frac{T_p}{T_z} &= 1.407(1 - 0.287 \ln \gamma)^{\frac{1}{4}}
 \end{aligned} \tag{3.59}$$

Adding the harmonic components together to obtain time series is performed using a Fourier Transformation algorithm. This approach implies a set of relations between the time increment,  $\Delta t$ , frequency increment,  $\Delta\omega$ , number of time steps,  $N_t$ , and the number of components,  $N_\omega$ .

$$\begin{aligned}
 \Delta\omega &= \frac{2\pi}{N_t \Delta t} \\
 N_\omega &= \frac{N_t}{2} + 1
 \end{aligned} \tag{3.60}$$

The duration of the time series is limited to

$$T_{dur} = N_t \Delta t \approx 2N_\omega \Delta t \tag{3.61}$$

The duration of the time series,  $T_{dur}$ , will be adjusted to fulfill the following FFT criteria

$$N_t = 2^\alpha \tag{3.62}$$

In addition the length,  $T_{dur}$ , will be, if necessary, adjusted to cover the duration of dynamic analysis.

EXAMPLE:

---

```

# 
#      seagr type      wav hist x0  y0  beta Tp Hs   D     dt     Tdur Tstart Kdepth Seed Spec
WAVELO sea1  IRREGULAR 100 250  0.0 0.0 0.0 10 4.0 2200 0.25 4096 1.0      125      1.0  1

```

---

## 4 SIMPost Report Generator

### 4.1 General

SIMPost includes the following files:

**prefix.spi** the SIMPost Input File.

**prefix.spl** the SIMPost Log File where warnings and error messages are written.

References are given in the input file to the relevant .raf file and the output .mpf files. All output is given on MATRIXPLOT (.mpf) format, which are ASCII files that can be further processed by e.g. Matlab or Python scripts.

Note that results are available only at time steps which are stored on the .raf file.

When running SIMPost from the command line, the basis is to write the name referring to SIMPost in the prompt. In addition, the following optional command line arguments can be applied:

- Adding -m and a float number representing the multiplication factor for scaling of the default local work memory size used by SIMPost.
- Adding -n and the input file name prefix makes SIMPost execute the post-processing without further requesting the user to specify the file name.
- Adding -d and a float number representing the multiplication factor for scaling of the default memory size for retrieving the RAF file results from SIMLA.

The optional arguments above may be specified in an arbitrary order. The option -d may be required when SIMLA also has been executed with the -m option as described in Section 2.5.

EXAMPLE: \_\_\_\_\_

---

```
command line: SIMPost -n prefixname -m 2.1 -d 4.4
```

---

### 4.2 Input data

All input is described on ASCII file format. The maximum line length is 136 characters and the maximum number of letters in one single text string is 32.

Comment text strings are defined by introducing # at the start of the line.

The following identifiers define the different data groups:

**NO PLOT** for NOdal history PLOTs

**IP PLOT** for Integration point history PLOTs

**EL PLOT** for EElement history PLOTs

**GN PLOT** for Global Nodal PLOTs

**GL PLOT** for Global Element PLOTs

**RS PLOT** for ReStart PLOTs

**NR PLOT** for Nodal Reaction PLOTs

**FATIGUE** for FATIGUE calculations

**EN PLOT** for ENvelope PLOTs

**VIVFAT** for Vortex Induced Vibration FATigue

In the following the input data will be explained.

### 4.3 NO PLOT - NOdal history PLOTs

**NO PLOT** is linked to the nodal quantities in terms of accumulated nodal displacement and rotation only.

The **NO PLOT** format is:

```
NO PLOT RAFPRE MPFPRE XLEG XRES YLEG YRES FNODID LNODID XSCL  
YSCL
```

**RAFPRE**: The SIMLA.raf file name prefix. If the name is defined by a combination of lower case and upper case letters, use e.g. 'input-filename'

**MPFPRE**: The output .mpf file name prefix. If the file name is defined by a combination of lower case and upper case letters, use e.g. 'mpf-filename'

**XLEG**: The legend name for the x-axis. If the legend name contains more than one word or if the name is defined by a combination of lower case and upper case letters, use e.g 'Displacement (m)'

**XRES**: x-axis result type. The following result types are available:

**LOADSTEP** : Load step

**TIME** : Time

**HISTN** : History number with reference to SIMLA input file, e.g HIST100

**YLEG**: The legend name for the y-axis. If the legend name contains more than one word or if the name is defined by a combination of lower case and upper case letters, use e.g. 'Displacement (m)'

**YRES**: y-axis result type. The result types available are according to the user selected results, see below:

**NODISP-X**: x-displacement (unit: L)

**NODISP-Y**: y-displacement (unit: L)

**NODISP-Z**: z-displacement (unit: L)

**NOROT-X**: x-displacement (unit: L)

**NOROT-Y**: y-displacement (unit: L)

**NOROT-Z**: z-displacement (unit: L)

**FNODEID**: First node ID number in visual model

**LNODEID**: Last node ID number in visual model

**XSCL**: Scaling factor to be used for x-axis (to convert to convenient unit)

**YSCL**: Scaling factor to be used for y-axis (to convert to convenient unit)

EXAMPLE: \_\_\_\_\_

```
#      Raf file name      plotfilename
NOPLOT "bellmouthdpwumb11" "belldpwumb11-sxx"
#
# x-legend    x-res    y-legend    y-res    fnodid lnodid xscl yscl
"Angle (deg)" HIST200 "Stress (MPa)" NOROT-X 16550   16611 10   1
```

#### 4.4 IPPLLOT - Integration Point history PLOTS

The **IPPLLOT** is linked to the nodal selection defined in Section 3.54

The **IPPLLOT** format is:

<b>IPPLLOT RAFPRE MPFPRE XLEG XRES YLEG YRES FELID LELID LSECID CSE-CID XSCL YSCL</b>
---

**RAFPRE**: The SIMLA.raf file name prefix. If the name is defined by a combination of lower case and upper case letters, use e.g. 'input-filename'.

**MPFPRE**: The output .mpf file name prefix. If the file name is defined by a combination of lower case and upper case letters, use e.g. 'mpf-filename'.

**XLEG**: The legend name for the x-axis. If the legend name contains more than one word or if the name is defined by a combination of lower case and upper case letters, use e.g 'Displacement (m)'.

**XRES:** x-axis result type. The following result types are available:

**LOADSTEP :** Load step

**TIME :** Time

**HISTN :** History number with reference to SIMLA input file, e.g HIST100

**YLEG:** The legend name for the y-axis. If the legend name contains more than one word or if the name is defined by a combination of lower case and upper case letters, use e.g. 'Displacement (m)'.

**YRES:** y-axis result type. The result types available are according to the user selected results, see Section 3.54.

**FELID:** First element ID number in visual model.

**LELID:** Last element ID number in visual model.

**LSECID:** Integration section number along element. Max 2 for PIPE31, PIPE34 and PIPE37. Max 3 for PIPE33, PIPE36, PIPE39, COMPIPE42 and CABLE111.

**CSECID:** Integration section number in cross-section. The total number is given in the control card of the SIMLA analysis.

**XSCL:** Scaling factor to be used for x-axis (to convert to convenient unit).

**YSCL:** Scaling factor to be used for y-axis (to convert to convenient unit).

EXAMPLE: \_\_\_\_\_

```
#      Raf file name plotfilename
IPPLOT "bellmouth11" "bell111-sxx"
#
#x-legend      x-res      y-legend      y-res      fnodid lnodid felid lelid lsecid csecid xscl yscl
"Angle(deg)" HIST200 "Stress(MPa)" SIGMA-XX 16550 16611   1      5      10      1      1.0  1.0
```

---

## 4.5 ELPLOT - EElement history PLOTS

The **ELPLOT** format is:

<b>ELPLOT RAFFPRE MPFPRE XLEG XRES YLEG YRES FELID LELID XSCL YSCL [ELEND ]</b>
---

**RAFFPRE:** The SIMLA.raf file name prefix. If the file name is defined by a combination of lower case and upper case letters, use e.g. 'input-filename'.

**MPFPRE:** The output .mpf file name prefix. If the file name is defined by a combination of lower case and upper case letters, use e.g. 'mpf-filename'.

**XLEG:** The legend name for the x-axis. If the legend name contains more than one word or if the name is defined by a combination of lower case and upper case letters, use e.g. 'Displacement (m)'.

**XRES:** x-axis result type. The following result types are available:

**LOADSTEP :** Load step

**TIME :** Time

**HISTN :** History number with reference to SIMLA input file, e.g. HIST100

**YLEG:** The legend name for the y-axis. If the legend name contains more than one word or if the name is defined by a combination of lower case and upper case letters, use e.g. 'Displacement (m)'.

**YRES:** y-axis result type. The following result types are available:

**ELFORCE-X :** Fx force (unit: F)

**ELFORCE-Y :** Fy force (unit: F)

**ELFORCE-Z :** Fz force (unit: F)

**ELMOM-X :** Mx moment about centroid x-axis (unit: FL)

**ELMOM-Y :** My moment about centroid y-axis (unit: FL)

**ELMOM-Z :** Mz moment about centroid z-axis (unit: FL)

**ELDISP-X :** Local element x displacement (element elongation for beams) (unit: L)

**ELDISP-Y :** Local element y displacement (zero for beam) (unit: L)

**ELDISP-Z :** Local element z displacement (zero for beam) (unit: L)

**ELROT-X :** Local x-rotation (rigid body rotation eliminated for beam) (unit: R)

**ELROT-Y :** Local y-rotation (rigid body rotation eliminated for beam) (unit: R)

**ELROT-Z :** Local z-rotation (rigid body rotation eliminated for beam)(unit: R)

**ELTOR-X :** torsion about centroid (unit: 1/L)

**ELCUR-Y :** y-curvature about centroid (unit: 1/L)

**ELCUR-Z :** z-curvature about centroid (unit: 1/L)

**CONDIS-X :** Contact element displacement in local x-direction (unit: L)

**CONDIS-Y :** Contact element displacement in local y-direction (unit: L)

**CONDIS-Z :** Contact element displacement in local z-direction (unit: L)

**CONFOR-X :** Contact element force in local x-direction (unit:  $FL^{-1}$  for all contact elements except CONT124, 130, 152, 153, 164, 170 having unit: F ).

**CONFOR-Y :** Contact element force in local y-direction (unit:  $FL^{-1}$  for all contact elements except CONT124, 130, 152, 153, 164, 170 having unit: F ).

**CONFOR-Z :** Contact element force in local z-direction, (unit:  $FL^{-1}$  for all contact elements except CONT124, 130, 152, 153, 164 having unit: F ).

**SUBMASS :** dry mass - buoyancy mass (unit:  $ML^{-1}$ )

**DRYMASS :** dry mass (unit:  $ML^{-1}$ )

**INISTRA-X** : element initial axial strain (unit: -)  
**TEMP** : element temperature (unit: °C)  
**INTPRES** : internal pressure (unit: FL<sup>-2</sup>)  
**EXTPRES** : external pressure (unit: FL<sup>-2</sup>)  
**INITOR-X** : initial torsion (unit: L<sup>-1</sup>)  
**INICUR-Y** : initial curvature about local y-axis (unit: L<sup>-1</sup>)  
**INICUR-Z** : initial curvature about local z-axis (unit: L<sup>-1</sup>)  
**ICONDIS-X** : Contact element initial displacement in local x-direction (unit: L)  
**ICONDIS-Y** : Contact element initial displacement in local y-direction (unit: L)  
**ICONDIS-Z** : Contact element initial displacement in local z-direction (unit: L)

**FELID**: First element ID number in numerical model.

**LELID**: Last element ID number in numerical model.

**XSCL**: Scaling factor to be used for x-axis (to convert to convenient unit).

**YSCL**: Scaling factor to be used for y-axis (to convert to convenient unit).

**ELEND**: Element end 1 or 2. Default value: 1

EXAMPLE:

---

```

#           Raf file name      plotfilename
ELPLOT  "slay-np"          "slay-np-cdz"
#
# x-legend     x-res       y-legend                  yres      felid   lelid   xscl yscl
"Load step"  LOADSTEP  "Contact z-disp (m)"    CONDIS-Z  2301   2316   1     1
#
#           Raf file name      plotfilename
ELPLOT  "slay-np"          "slay-np-cfz"
#
# xleg        xres       yleg                  yres      felid   lelid   xscl yscl
"Load step"  LOADSTEP  "Contact z-lload(kN/m)"  CONFOR-Z  2301   2316   1     1e3

```

---

## 4.6 GN PLOT - Global Nodal PLOTs

The **GN PLOT** format is:

<b>GNPLOT RAFPRE MPFPRE XLEG XRES YLEG YRES FNODEID LNODEID XSCL YSCL [LOADSTEP ]</b>
---

**RAFPRE**: The SIMLA.raf file name prefix. If the name is defined by a combination of lower case and upper case letters, use e.g. 'input-filename'.

**MPFPRE**: The output .mpf file name prefix. If the file name is defined by a combination of lower case and upper case letters, use e.g. 'mpf-filename'.

**XLEG:** The legend name for the x-axis. If the legend name contains more than one word or if the name is defined by a combination of lower case and upper case letters, use e.g. 'Displacement (m)'.

**XRES:** x-axis result type. The following result types are available:

**X-COR** : x-coordinate (unit: L)

**Y-COR** : y-coordinate (unit: L)

**Z-COR** : z-coordinate (unit: L)

**T-COR** : Polar theta coordinate (for nodes generated by this option) (unit: R)

**A-COR** : Polar lay angle (for nodes generated by this option) (unit: R)

**E-COR** : curvilinear element coordinate (unit: L)

**K-COR** : curvilinear xy-coordinate, i.e. kp-coordinate for pipeline problems requires that a seabed has been defined in SIMLA (unit: L)

**YLEG:** The legend name for the y-axis. If the legend name contains more than one word or if the name is defined by a combination of lower case and upper case letters, use e.g. 'Displacement (m)'.

**YRES:** y-axis result type. The following result types are available:

**X-COR** : nodal x-coordinate (unit: L)

**Y-COR** : nodal y-coordinate (unit: L)

**Z-COR** : nodal z-coordinate. If a seabed surface is defined (k-cor is used on x-axis) then the seabed z-coordinate will also be plotted (unit: L)

**T-COR** : Polar theta coordinate (for nodes generated by this option) (unit: R)

**A-COR** : Polar lay angle (for nodes generated by this option) (unit: R)

**NODISP-X** : nodal x-displacement (unit: L)

**NODISP-Y** : nodal y-displacement (unit: L)

**NODISP-Z** : nodal z-displacement (unit: L)

**NOROT-X** : nodal x-rotations (unit: 1/L)

**NOROT-Y** : nodal y-rotations (unit: 1/L)

**NOROT-Z** : nodal z-rotations (unit: 1/L)

**VNDISP-X** : visual model nodal x-displacement (unit: L)

**VNDISP-Y** : visual model nodal y-displacement (unit: L)

**VNDISP-Z** : visual model nodal z-displacement (unit: L)

**FNODEID:** First node ID number in numerical model.

**LNODEID:** Last node ID number in numerical model.

**XSCL:** Scaling factor to be used for x-axis (to convert to convenient unit).

**YSCL:** Scaling factor to be used for y-axis (to convert to convenient unit). If a negative value is used, **NODISPX-Z** is transformed to local x-y system for the element,

which is convenient when evaluating displacement from the route configuration for pipeline cases.

**LOADSTEP:** Reference load step number, only applicable for (nodisp-x), (nodisp-y) (nodisp-z) Allows the user to select which loadstep from which the nodal displacements are measured.

#### EXAMPLE:

---

```
#      transform to xy system and measure displacement from step 10
#
#      .raf prefix      .mpf prefix
GNPLOT "slay-np"      "slay-np-xz"
#
#      Legend x      x-res.  Legend y          y-res. fnodid lnodid xscl yscl Loadstep
"X-coordinate (m)" K-COR   "KP-coordinate (m)" Z-COR   1       441     1     -1     10
```

---

## 4.7 GLPLOT - Global EElement PLOTs

The **GLPLOT** format is:

**GLPLOT RAFPRE MPFPRE XLEG XRES YLEG YRES FELID LELID XSCL YSCL  
[ELEND INTPOI NLOADSTEP INC]**

**RAFPRE:** The SIMLA.raf file name prefix. If the file name is defined by a combination of lower case and upper case letters, use e.g. 'input-filename'.

**MPFPRE:** The output .mpf file name prefix. If the file name is defined by a combination of lower case and upper case letters, use e.g. 'mpf-filename'.

**XLEG:** The legend name for the x-axis. If the legend name contains more than one word or if the name is defined by a combination of lower case and upper case letters, use e.g. 'Displacement (m)'.

**XRES:** x-axis result type. The following result types are available:

**X-COR :** x-coordinate (unit: L)

**Y-COR :** y-coordinate (unit: L)

**Z-COR :** z-coordinate (unit: L)

**T-COR :** Polar theta coordinate (for elements connected to nodes generated by this option) (unit: R)

**A-COR :** Polar lay angle (for elements connected to nodes generated by this option) (unit: R)

**E-COR :** curvilinear element coordinate (unit: L)

**K-COR** : curvilinear xy-coordinate, i.e. kp-coordinate for pipeline problems requires that a seabed has been defined in SIMLA (unit: L)

**YLEG**: The legend name for the y-axis. If the legend name contains more than one word or if the name is defined by a combination of lower case and upper case letters, use e.g. 'Displacement (m)'.

**YRES**: y-axis result type. All integration point results can be applied, see Section 3.54. In addition, the following result types can be applied:

**ELFORCE-X** : Fx force (unit: F)

**ELFORCE-Y** : Fy force (unit: F)

**ELFORCE-Z** : Fz force (unit: F)

**ELMOM-X** : Mx moment about centroid x-axis (unit: FL)

**ELMOM-Y** : My moment about centroid y-axis (unit: FL)

**ELMOM-Z** : Mz moment about centroid z-axis (unit: FL)

**ELDISP-X** : Local element x displacement (element elongation for beams) (unit: L)

**ELDISP-Y** : Local element y displacement (zero for beam) (unit: L)

**ELDISP-Z** : Local element z displacement (zero for beam) (unit: L)

**ELROT-X** : Local x-rotation (rigid body rotation eliminated for beam) (unit: R)

**ELROT-Y** : Local y-rotation (rigid body rotation eliminated for beam) (unit: R)

**ELROT-Z** : Local z-rotation (rigid body rotation eliminated for beam) (unit: R)

**ELTOR-X** : Torsion about centroid (unit: 1/L)

**ELCUR-Y** : y-curvature about centroid (unit: 1/L)

**ELCUR-Z** : z-curvature about centroid (unit: 1/L)

**CONDIS-X** : Contact element displacement in local x-direction (unit: L)

**CONDIS-Y** : Contact element displacement in local y-direction (unit: L)

**CONDIS-Z** : Contact element displacement in local z-direction (unit: L)

**CONFOR-X** : Contact element force in local x-direction (unit:  $FL^{-1}$  for all contact elements except CONT124 and 130 having unit: F ).

**CONFOR-Y** : Contact element force in local y-direction (unit:  $FL^{-1}$  for all contact elements except CONT124 and 130 having unit: F ).

**CONFOR-Z** : Contact element force in local z-direction (unit:  $FL^{-1}$  for all contact elements except CONT124, 130 having unit: F ).

**SUBMASS** : dry mass - buoyancy mass (unit:  $ML^{-1}$ )

**DRYMASS** : dry mass (unit:  $ML^{-1}$ )

**INISTRAX** : element initial axial strain (unit: -)

**TEMP** : element temperature (unit: °C)

**INTPRES** : internal pressure (unit:  $\text{FL}^{-2}$ )  
**EXTPRES** : external pressure (unit:  $\text{FL}^{-2}$ )  
**INITOR-X** : initial torsion (unit:  $\text{L}^{-1}$ )  
**INICUR-Y** : initial curvature about local y-axis (unit:  $\text{L}^{-1}$ )  
**INICUR-Z** : initial curvature about local z-axis (unit:  $\text{L}^{-1}$ )  
**ICONDIS-X** : Contact element initial displacement in local x-direction (unit: L)  
**ICONDIS-Y** : Contact element initial displacement in local y-direction (unit: L)  
**ICONDIS-Z** : Contact element initial displacement in local z-direction (unit: L)

**FELID**: First element ID number in numerical model.

**LELID**: Last element ID number in numerical model.

**XSCL**: Scaling factor to be used for x-axis (to convert to convenient unit).

**YSCL**: Scaling factor to be used for y-axis (to convert to convenient unit).

**ELEND**: Element end 1 or 2. Default value: 1

**INTPOI**: Integration point number (around cross section, point 1 start at local  $y=\text{radius}$  and  $z = 0$  and then clockwise). If no integration point, this is the negative reference load step number. Default value: 1

**NLOADSTEP**: If a positive load step number is given, then the results will be measured relative to the state at that step. If no or a negative loadstep is given, then actual values are used. Default value: 0

**INC**: Element increment related for each nodal point. Default value: 1

EXAMPLE:

---

```
#       .raf prefix      .mpf prefix
GLPLOT "slay-np"      "slay-np-ax"
#
# Legend x   x-res. Legend y           y-res.    nodeid lnodeid xscl yscl Elend
"S-coord (m)" E-COR  "Axial force (kN)"  ELFOR-X  1      440      1      1      1
#
#       .raf prefix      .mpf prefix
GLPLOT "slay-np"      "slay-np-sxx"
#
# Legend x   x-res. Legend y           y-res.    nodeid lnodeid xscl yscl Elend Intpoi
"KP-coord (m)" K-COR  "Stress-xx (MPa)" Sigma-xx  1      440      1      1      1      5
```

---

## 4.8 FAPLOT- Fatigue calculation

The purpose of the **FAPLOT** option is to allow the user to perform fatigue damage calculation for all visual nodes in the structure and where the stress components xx, yy and xy are all stored. Note that this requires the following:

1. That the fatigue data relevant for structural element material is defined, see Section 3.25
2. That the result types SIGMA-XX and FATIGUE is selected, see Section 3.54

The **FAPLOT** format is:

**FAPLOT RAFPRE LOFPRE I3 FTIME LTIME OPTSTR UNTCONV**

**RAFPRE**: The .raf file name prefix. If the file name is defined by a combination of lower case and upper case letters, use e.g. 'input-filename'.

**LOFPRE**: The output .lof file name prefix. If the file name is defined by a combination of lower case and upper case letters, use e.g. 'fat-filename'.

**I3**: the number of load cycles.

**FTIME**: the first load step for calculating stress range.

**LTIME**: the last load step point for calculating stress range.

**OPTSTR**: Option for stress range calculation.

**# 1** : Stress range is taken to be the difference between the stress ranges obtained at load steps **FTIME** and **LTIME**.

**1** : Stress range is taken to be the largest stress range between load steps **FTIME** and **LTIME**.

**UNTCONV**: unit conversion factor to fit the fatigue data.

## 4.9 NR PLOT- Nodal Reaction PLOTS

The purpose of the **NR PLOT** option is to allow the user to sum the element forces acting on a certain node and follow this node as function of history. The total force is transformed to global system. Note that if reaction forces is to be obtained, the effect of external loads need to be added manually. This is because only element info is stored on data base.

The **NR PLOT** format is:

**NR PLOT RAFPRE MPDPRE XLEG XRES YLEG YRES FNODEID LNODEID XSCL YSCL [IEND]**

**RAFPRE**: The SIMLA.raf file name prefix. If the file name is defined by a combination of lower case and upper case letters, use e.g. 'input-filename'.

**MPDPRE**: The output .mpf file name prefix. If the file name is defined by a combination of lower case and upper case letters, use e.g. 'mpf-filename'.

**XLEG:** The legend name for the x-axis. If the legend name contains more than one word or if the name is defined by a combination of lower case and upper case letters, use e.g. 'Displacement (m)'.

**XRES:** x-axis result type. The following result types are available:

**LOADSTEP :** Load step

**TIME :** Time

**HISTN :** History number with reference to SIMLA input file, e.g. HIST100

**YLEG:** The legend name for the y-axis. If the legend name contains more than one word or if the name is defined by a combination of lower case and upper case letters, use e.g. 'Displacement (m)'.

**YRES:** y-axis result type. The following result types are available:

**QX-REACT :** Fx restoring force (unit: F)

**QY-REACT :** Fy restoring force (unit: F)

**QZ-REACT :** Fz restoring force (unit: F)

**MX-REACT :** Mx restoring moment about centroid x-axis (unit: FL)

**MY-REACT :** My restoring moment about centroid y-axis (unit: FL)

**MZ-REACT :** Mz restoring moment about centroid z-axis (unit: FL)

**QX-DAMP :** Fx damping force (unit: F)

**QY-DAMP :** Fy damping force (unit: F)

**QZ-DAMP :** Fz damping force (unit: F)

**MX-DAMP :** Mx damping moment about centroid x-axis (unit: FL)

**MY-DAMP :** My damping moment about centroid y-axis (unit: FL)

**MZ-DAMP :** Mz damping moment about centroid z-axis (unit: FL)

**QX-INER :** Fx inertia force (unit: F)

**QY-INER :** Fy inertia force (unit: F)

**QZ-INER :** Fz inertia force (unit: F)

**MX-INER :** Mx inertia moment about centroid x-axis (unit: FL)

**MY-INER :** My inertia moment about centroid y-axis (unit: FL)

**MZ-INER :** Mz inertia moment about centroid z-axis (unit: FL)

**QX-ALL :** Fx restoring+damping+inertia force (unit: F)

**QY-ALL :** Fy restoring+damping+inertia force (unit: F)

**QZ-ALL :** Fz restoring+damping+inertia force (unit: F)

**MX-ALL :** Mx restoring+damping+inertia moment about centroid x-axis (unit: FL)

**MY-ALL :** My restoring+damping+inertia moment about centroid y-axis (unit: FL)

**MZ-ALL** : Mz restoring+damping+inertia moment about centroid z-axis (unit: FL)

**FNODEID**: First node ID number in numerical model.

**LNODEID**: Last node ID number in numerical model.

**XSCL**: Scaling factor to be used for x-axis (to convert to convenient unit).

**YSCL**: Scaling factor to be used for y-axis (to convert to convenient unit).

**IEND**: The element end that shall contribute. Default value = 0 means that all element ends are considered.

#### 4.10 RS PLOT - ReStart PLOT

The ReStart PLOT format is:

**RS PLOT RAFPRE MPFPRE LOADSTEP FNODEID LNODEID NODESCL VALSCL**

**RAFPRE**: The .raf file name prefix. If the file name is defined by a combination of lower case and upper case letters, use e.g. 'input-filename'.

**MPFPRE**: The output file name If the file name is defined by a combination of lower case and upper case letters, use e.g. 'mpf-filename'.

**LOADSTEP**: Load step number.

**FNODID**: First node ID number in numerical model.

**LNODEID**: Last node ID number in numerical model.

**NODESCL**: Scaling factor to be used for node number ( $i = i - NODESCL$  ).

**VALSCL**: Scaling factor to be used for x-,y and z-values ( $xi = xi * VALSCL$  ) .

The output file format is:

Node number,  $Xi$  ,  $Yi$  ,  $Zi$  , T11, T12, T13, T21, T22, T23, T31, T32, T33

#### 4.11 EN PLOT- ENvelope PLOTs

The ENvelope PLOT format is:

**EN PLOT RAFPRE MPFPRE XRES XSCL YSCL**

**RAFPRE**: The SIMLA.raf file name prefix. If the file name is defined by a combination of lower case and upper case letters, use e.g. 'input-filename'.

**MPFPRE**: The output .mpf file name prefix. If the file name is defined by a combination of lower case and upper case letters, use e.g. 'mpf-filename'.

**XRES**: The x-axis type, that may have the following value:

**E-COR** : Curvilinear s-coordinate

**ELEM** : Element or node ID number

**XSCL**: Scaling factor to be used for x-axis (to convert to convenient unit).

**YSCL**: Scaling factor to be used for y-axis (to convert to convenient unit).

Then all defined envelopes will be plotted stored on .mpf files.

EXAMPLE:

---

```
#      raf file      plot file      xres  xscl  yscl
ENPLOT "slay-np"  "envelopes-np" ELEM   1      1
```

---

## 4.12 VIVFAT - Vortex Induced Vibration FATigue

VIVFAT generates a set of cross flow and in line eigen modes and the corresponding eigen frequencies. The added mass which is used is still water added mass found from DNV1986.

The **VIVFAT** format is:

<b>VIVFAT RAFPRE MPFPRE XLEG XRES KPSTRT KPEND LOADSTP NMODES ROUGH</b>
---

**RAFPRE**: The SIMLA.raf file name prefix. If the file name is defined by a combination of lower case and upper case letters, use e.g. 'input-filename'.

**MPFPRE**: The output .mpf file name prefix. If the file name is defined by a combination of lower case and upper case letters, use e.g. 'mpf-filename'.

**XLEG**: The legend name for the x-axis. If the legend name contains more than one word or if the name is defined by a combination of lower case and upper case letters, use e.g. 'Displacement (m)'.

**XRES**: x-axis result type. The following result types are available:

**X-COR** : x-coordinate (unit: L)

**Y-COR** : y-coordinate (unit: L)

**Z-COR** : z-coordinate (unit: L)

**E-COR** : curvilinear element coordinate (unit: L)

**K-COR** : curvilinear xy-coordinate, i.e. kp-coordinate for pipeline problems requires that a seabed has been defined in SIMLA (unit: L)

**KPSTRT**: KP start value for the free span.

**KPEND**: KP end value for the free span.

**LOADSTP**: Load step.

**NMODES**: The total number of cross flow and in-line eigen modes.

**ROUGH:** Roughness of pipe.

*Note:* Only single pipes should be applied for this options. Piggy back pipes or PiP systems are not handled.

EXAMPLE: \_\_\_\_\_

```
#       .raf prefix     .mpf prefix
VIVFAT      "snaking"      "snaking-viv"
#
# Legend x           x-res.   KPstrt   Kpend    LOADST NMODES ROUGH
"KP-coordinate [m]"   K-COR      4500.0   4700.0    25      5      MARIN
#
```

---

## 5 DYNPost Report Generator

### 5.1 General

The **DYNRES** results are stored on .dyn files having the same prefix as the SIMLA input files. Each time series plot is stored in consecutive order and may be plotted from the XPOST program. The DYNPOST program is applied for generation if ASCII files for each time series. As opposed to SIMPOST, the results are available for all time steps.

DYNPOST requires an input file with extension .sdi and will report **MXPLOT** on a log file ending with extension .sdo and **DYNPLOT** to .mpf files specified by the user. All output is given on MATRIXPLOT (.mpf) format, which are ASCII files that can be further processed by e.g. Matlab or Python scripts.

When running DYNPOST from the command line, the basis is to write the name referring to DYNPOST in the prompt. In addition, the following optional command line arguments can be applied:

- Adding -m and a float number representing the multiplication factor for scaling of the default local work memory size used by DYNPOST.
- Adding -n and the input file name prefix makes DYNPOST execute the post-processing without further requesting the user to specify the file name.
- Adding -d and a float number representing the multiplication factor for scaling of the default memory size for retrieving **DYNRES** results from SIMLA.

The optional arguments above may be specified in an arbitrary order. The option -d may be required when SIMLA also has been executed with the -d option as described in Section 2.5.

EXAMPLE: \_\_\_\_\_

```
command line: DynPost -n prefixname -m 3.4 -d 7.2
```

---

### 5.2 Input data to DYNPost

All input is described on ASCII file format. The maximum line length is 136 characters and the maximum number of letters in one single text string is 32.

Comment text strings are defined by introducing # at the start of the line.

The following identifiers define the different data groups:

- **MXPLOT** for MaXima or MiNima postprocessing.
- **DYNPLOT** for dynamic plot postprocessing.

In the following the input data will be explained.

### 5.2.1 MXPLOT - MaXima/MiNima postprocessing

The format is:

**MXPLOT I1 R2 I3 I4 R5 A6 A7 - AN**

**I1** : Extreme value type:

- 1** : Find maxima
- 1** : Finds minima
- 2** : Finds difference from local maxima to following local minima. By local the extreme value between two crossings of the mean value in the current result file is meant.
- 2** : Finds difference from local minima to following local maxima. By local the extreme value between two crossings of the mean value in the current result file is meant.

**R2** : The fraction relative to the absolute maxima/minima found that will be evaluated

**I3** : The **DYNRES** number in consecutive order as defined in the simla input file

**I4** : number of maxima to be counted

**R5** : start time of evaluation (to avoid transient)

**A6** : output file name prefix

**A7-AN** : prefix of all dyn files to be processed

EXAMPLE:

---

```
#      imax fraction idynres nmaxim tstart outputfile .dyn files
MXPLOT    1  0.3      1       100     30.0   "test"      "screening_270_9"
                                         "screening_181_9"
                                         "screening_225_9"
```

---

Typical result file:

EXAMPLE:

---

Nodal	Velocity	Node	1001	Dof	1	Maxima Statistics
time		value				seastatefile

```
0.3490E+02 0.1477E+00 screening_181_9.dyn
0.8490E+02 0.1227E+00 screening_181_9.dyn
0.7740E+02 0.9049E-01 screening_181_9.dyn
0.6420E+02 0.7781E-01 screening_181_9.dyn
0.9290E+02 0.7075E-01 screening_181_9.dyn
0.5390E+02 0.2325E+00 screening_225_9.dyn
0.9670E+02 0.1657E+00 screening_225_9.dyn
0.6760E+02 0.8834E-01 screening_225_9.dyn
0.7380E+02 0.8607E-01 screening_225_9.dyn
```

---

### 5.2.2 DYNPLOT - Dynamic plot postprocessing

The format is:

```
DYNPLOT A1 A2 I3 [I3B] R4
```

**A1** : the .dyn file name

**A2** : the .mpf plot storage file name

**I3** : the plot number

**I3B** : optional parameter. If specified, the plot number range i3-i3b will be stored to the same .mpf file

**R4** : y-axis scaling factor

EXAMPLE: \_\_\_\_\_

```
#      raf file    plot file        plotno   y-scale
DYNPLOT "slay-nt"  "slay-nt-my"      1       1e-3
```

---

EXAMPLE: \_\_\_\_\_

```
#      raf file    plot file        plotnos  plotnoe y-scale
DYNPLOT "slay-nt"  "slay-nt-my"      1         3       1e-3
```

---

## 6 Searching for Errors

Below are a few tips that may be useful when you get an error return, and the reason is not clear to you.

- You have got a error message when a input card is read, but you can not find any errors in the card:

Check the spelling of the next card identifier. SIMLA has a list of all the card keywords. If the spelling is incorrect SIMLA does not recognize it, and assume that all belongs to the previous card.

- You have just made a long list of nodes and/or element connectivities with the **NOCOOR** or **ELCON** card, and SIMLA goes in error:

Each card has a maximum number of parameters that can be read. (Maximum is usually 1000) Use several cards to define the nodes or element connectivities.

- You have designed your model and perform some sort of autostart, specifying the location of the model by a touch down point in the **CONTROL** card. However, the model does not end up where you expected:

SIMLA operates in a global system, and the points defined in the route file (given in **COSURF** card) is taken to be relative to this. However, you can define a additional local system, by specifying this in the **COSURF** card. It can be convenient to define a coordinate system closer to the route. When positioning your model by autostart, KP is the only concept that matters. This is the curvilinear length of the route, as given in the route file. KP value of the first point is defined in **COSURF**. Note: In **AUTOSTART** with **ICATEN**=0, the pipe will be positioned with the last end located at the given touch down KP

## 7 Third party licenses

The following list shows third party licenses including their conditions used by SIMLA:

- FINTERP: Modern Fortran Multidimensional Linear Interpolation  
<https://github.com/jacobwilliams/finterp>

Copyright (c) 2016-2022, Jacob Williams All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

\* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

\* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

\* The names of its contributors may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

## 8 Examples

### 8.1 General remarks

In the following, FE analyses examples will be presented, investigating different aspects of steel pipeline structural behaviour. These analyses are based on application of some of the finite elements presented in theory manual. The examples include:

- Dynamic simulation of pipeline installation by S-lay
- Simulation of cable kinking during J-lay
- J-tube pull-in
- Pipeline walking
- Lateral buckling versus alternative soil models
- Pipeline trawl-gear interaction loads

### 8.2 Dynamic simulation of pipeline installation by S-lay

#### 8.2.1 Modelling

The model is schematically shown in Fig. 8.1. Applied units are MN and m. Note that in the model, the tensioner node 441 is positioned at  $x = 71.56$  m and  $z = 25$  m relative to the COG node 3001 which is 15 m below the water plane (in the figure 441 is then to the right of node 3001). The water depth is 2000 m.

The pipe model is 3536 m long using 440 PIPE elements ID 1-440, 340 PIPE31 linear elements and 100 PIPE33 elastoplastic elements. Initially, the pipe is a straight line starting at  $x=9.999$  m and ending at  $x=3546.809$ , node IDs 1-441, which is at the tensioner, where the stinger curve start, see Section 3.39. The stinger is modelled by CONT164 elements where the master node for all elements is represented by the vessel COG node ID number 3001. Element eccentricity is introduced in these elements in order to position the roller box relative to node ID 3001. Note that since the pipe centreline is applied as the basis for the geometrical parameters, the radius need to be adjusted by  $R_{stinger} - R_{pipe} - R_{roller}$  when calculating roller positions to obtain consistency. This is because SIMLA takes the pipe and roller radii into account when calculating contact. There are 16 roller boxes, element IDs 2301-2316, which are positioned with 10-8 m spacing. The stinger radius is 120 m and the first roller starts at angle 10 deg. There

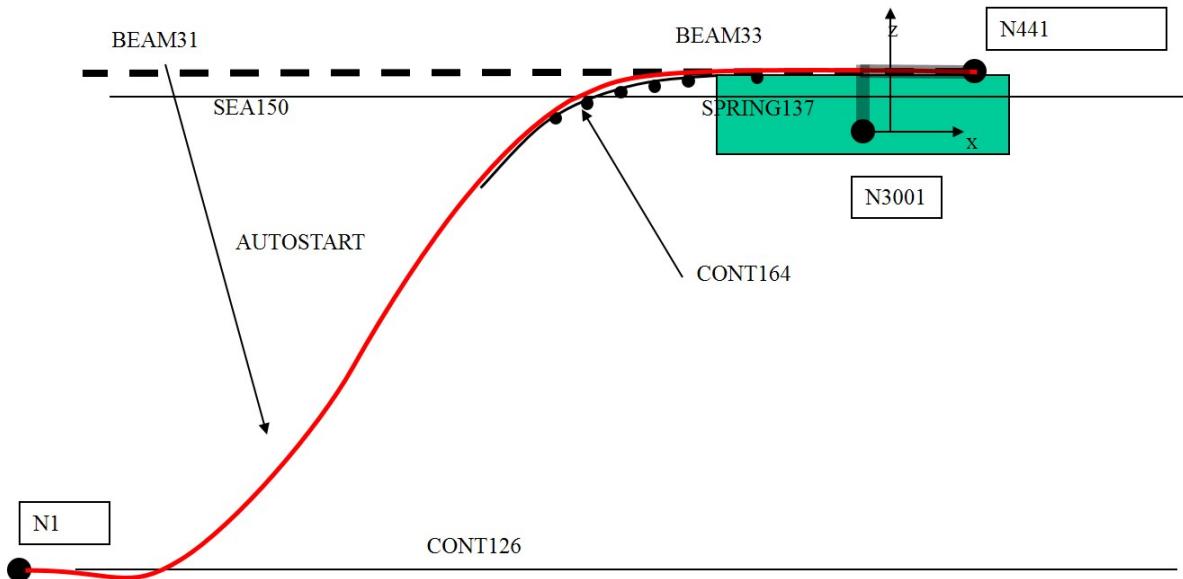


Figure 8.1: S-lay model

are two vertical rollers at the last and first roller boxes to avoid transverse displacement, so the total number of contact elements is 20, IDs 2401-2404. In order to connect the pipe end node ID 441 to the vessel and to allow tensioner modelling, one SPRING137 element is introduced, end 1 at node ID 3001 which is at the vessel COG and end 2 at pipe node ID 441. The theoretical distance from COG to the tensioner centreline is  $ex = 71.56$  m and  $ez = 25$  m. An eccentricity of  $x = 71.56$  m and  $z = 24.99$  m is introduced in the local element x-direction and at end 1 of spring element ID 3000 to ensure that the element end is positioned at the same point as node 441. The 24.99 m instead of 25 m is in order to avoid zero element length of the spring element SPRING137. Note that the element orientation of this element is obtained by using the **EULERANGLE** option for node 3001, see Section 3.39. There are also 200 CONT126 seabed contact elements which are numbered from IDs 1001-1200 which are connected to pipe node IDs 1-201 and 100 SEA150 elements, IDs 2101-2200, in order to introduce buoyancy and hydrodynamic loading. One SEA150 element is sufficient for the numeric model. For visual presentation of the wave, however, 100 elements have been selected.

The CONT126 element requires that the seabed geometry is imported from a text file using the **CODURFPR** card, see Section 3.8. Furthermore the contact elements need to be activated by using the **CONTINT** card, see Section 3.5. This also governs for Sea/Structure interaction, i.e. all elements that may be submerged must be told to do so by defining an interface between the respective structure element groups and the sea element group.

The static analysis is done in one load step using the **AUTOSTART** feature that enables initiation of both J-lay and S-lay, see Section 3.6.

The static and dynamic analysis is defined in two sequences using the **TIMECO** card, steps 0-1 with step length 1s for static analysis and then steps 1.1s-120s with 0.1s step length for dynamic analysis. For the static analysis gravity loading is applied in one step to the full value. Note that in this case the material properties for the SPRING137 vessel element have been set to zero as the displacement of node 441 is kinematically fully described by using the **CONSTR PDISP SPECIAL** option for node 441 using the eccentricities between node ID 3001 and node ID 441, see Section 3.4.1 and the **CONSTR PDISP RAO** for node 3001 which is the vessel. If the tensioner deadband effect was to be modelled, then the analysis would be carried out in two steps; the first step with

**CONSTR PDISP SPECIAL** 441 1 3001 0 0.0 0 71.56 0 25.000

to keep node 441 fixed; in the next step this constraint equation is omitted and the dead band characteristic introduced in the SPRING137 material curve together with applying the full value of the tension in element ID 440 as a concentrated load in node 441 (If not the pipe will fall off the stinger).

In order to allow visual representation of the model including the sea surface, the **VISRES** option have been applied. Visualization of sea surface elevation is provided by the **PDISP WAVE** option of the **CONSTR** card, see Section 3.4 whereas vessel movement is provided by the **PDISP RAO** option.

Note that the wave x0 is set to 1648.873 m. This value is taken from the .sof output file based on the coordinates of node ID 3001, the vessel node, after the s-lay has been initiated. This to ensure that the wave elevation is zero at t=0 and that wave elevation is initiated at COG from t > 0. Note that from version 2.06, this is automatically taken care of by Simla by specifying the COG node during **AUTOSTART** then using the COG node coordinate after initiation to set x0 and y0 for the wave generation. Transfer functions for surge, heave and pitch have been defined by the **RAOPRP**. This gives maximum combined surge and heave displacement amplitudes at the tensioner of 0.16 m and 0.325 m respectively for 7 m height wave with period 10 s applied here.

### 8.2.2 Example of results

In results in terms of dynamic plots are presented, see **DYNRES** card in Section 3.12. The plots are obtained from the Results menu in XPOST. By pointing with the mouse on the plot and entering CTRL-C, then moving to the word document and enter CTRL-V the plot is automatically pasted into the document.

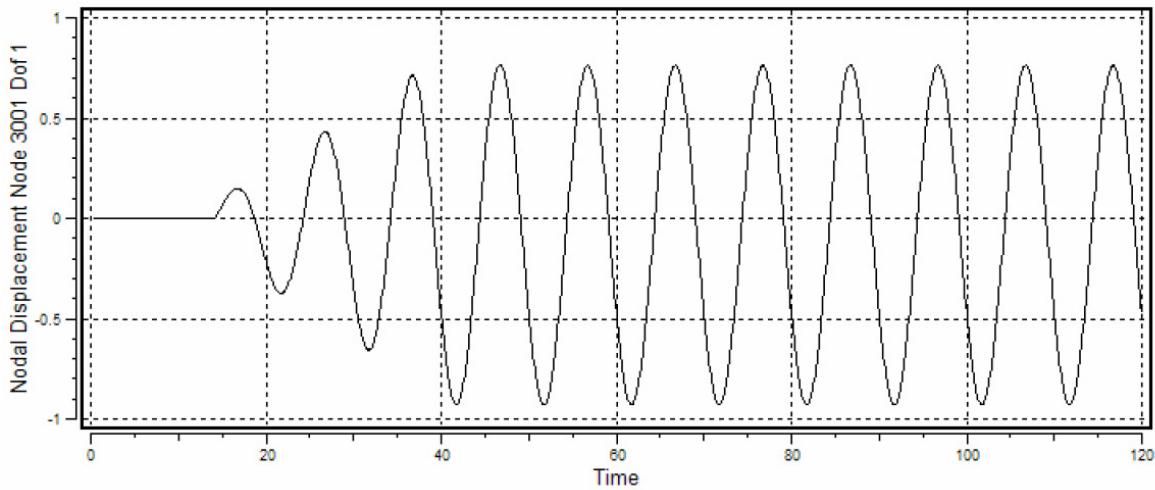


Figure 8.2: Vessel dynamic displacement history in surge node ID 3001.

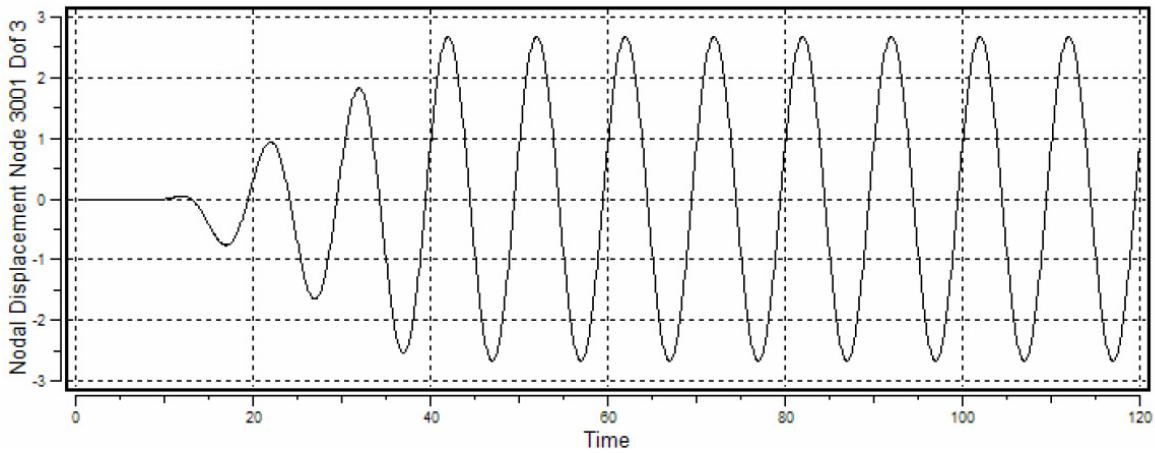


Figure 8.3: Vessel dynamic displacement history in heave node ID 3001.

### 8.3 The cable kinking problem

#### 8.3.1 Objectives

Installation of flexible pipes and cables is normally carried out in the J-lay mode. The weather window within which the installation operation can be safely conducted is limited by the vessel motions giving rise to dynamic tension and curvature at the touch down point (TDP). This may introduce torsion instabilities resulting in kink formation, see Fig. 8.10. Kink formation will in most cases be associated with plastic deformations of the cross-section due to the friction between the layers, i.e. the friction moment effect. Then any attempt of straightening the kink may result in severe curvatures that may destroy the cross-section components. The torsion instability will further be influenced by a range of parameters such as the cross-section torsion balance, the routing, vessel

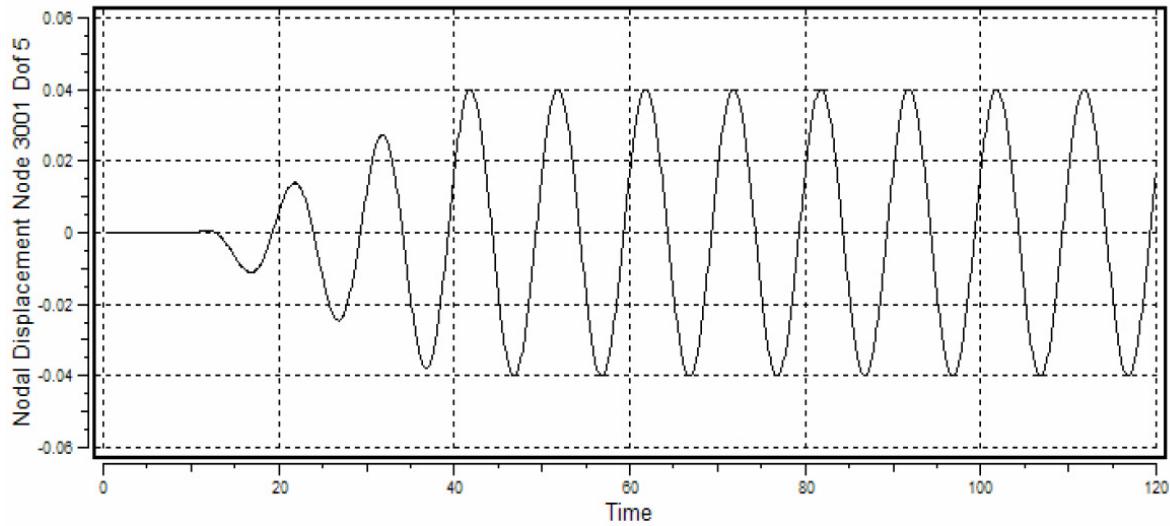


Figure 8.4: Vessel dynamic displacement history in pitch node ID 3001.

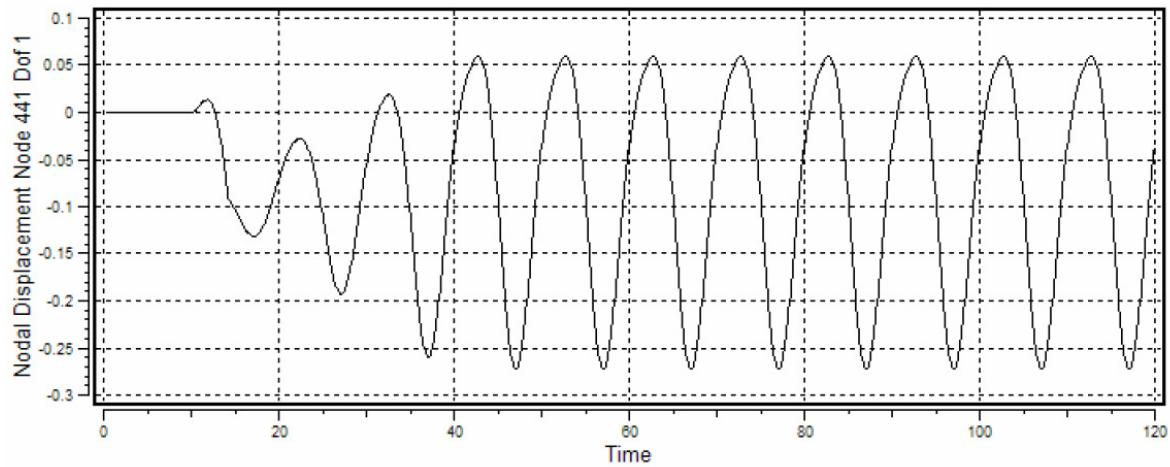


Figure 8.5: Dynamic displacement in surge at pipe tensioner node ID 441.

motions, seabed friction and built-in torque from the manufacturing and installation procedures.

An important question for the industry in later years is whether or not effective compression can be allowed at TDP during installation. If this can be allowed, the current practice of not allowing compression can be relaxed which will have a direct economic benefit with respect to the installation costs. In order to shed some light with respect to this question, a case study is performed with reference to the work by Neto and Martins, see ([Neto and Martins, 2013](#)). In their study the primary focus was to investigate how much torsion that can be allowed before instability occurs under the influence of the catenary configuration, the seabed constraint and the seabed friction forces and with reference to the Greenhill equation, see ([Greenhill, 1883](#)). Elastic material properties

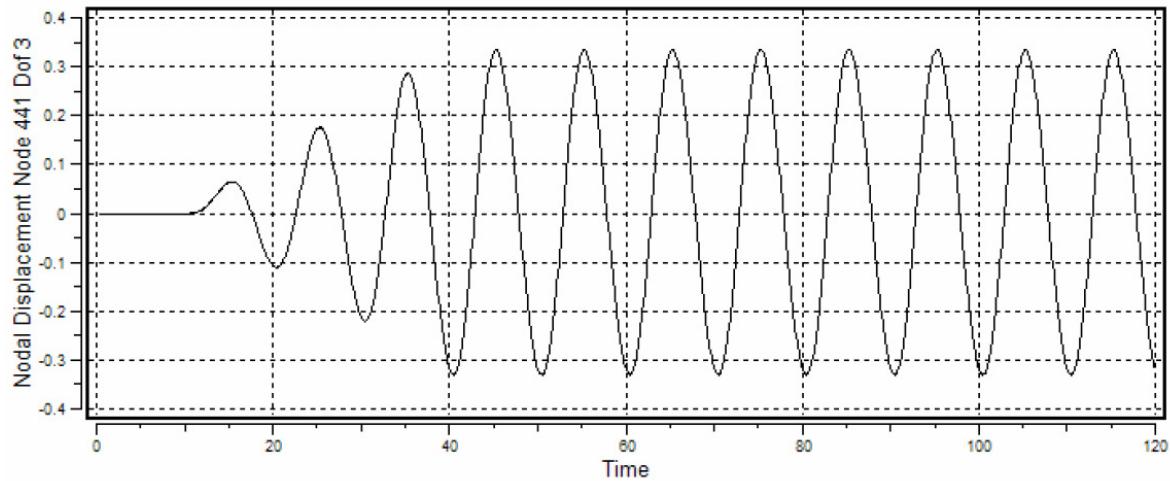


Figure 8.6: Dynamic displacement in heave at pipe tensioner node ID 441.

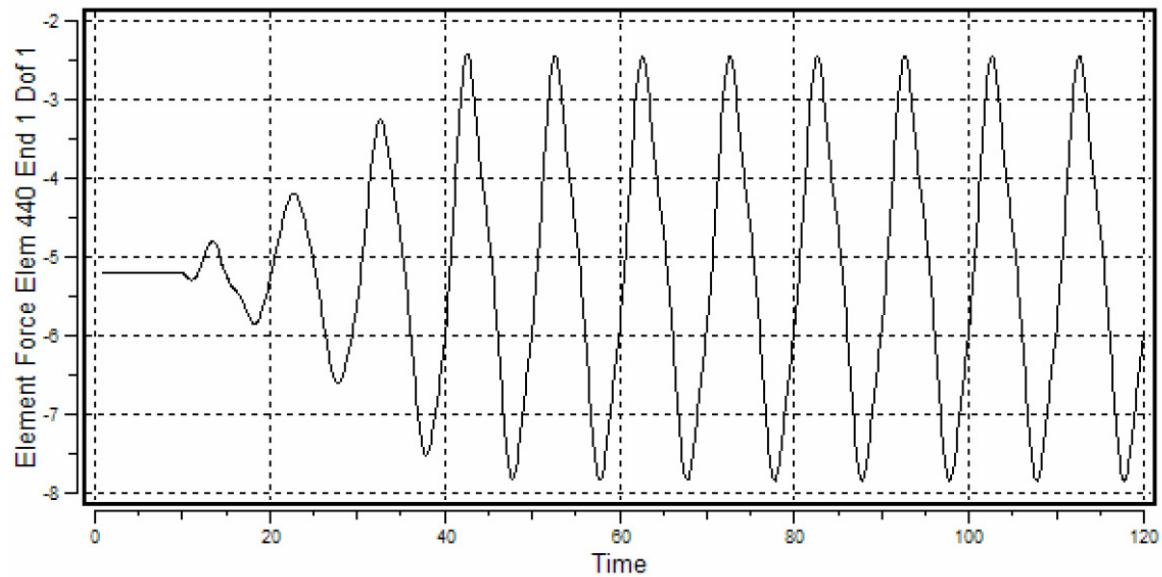


Figure 8.7: Dynamic axial force in element ID 440.

were assumed.

$$M_{cr} = \sqrt{4EI T} \quad (8.1)$$

with bending stiffness  $EI$  and effective tension  $T$ .

In this study the effect of the cross-section friction moment and the dynamic load conditions is additionally investigated and a design procedure proposed, ([Koloshkin, 2016](#)).

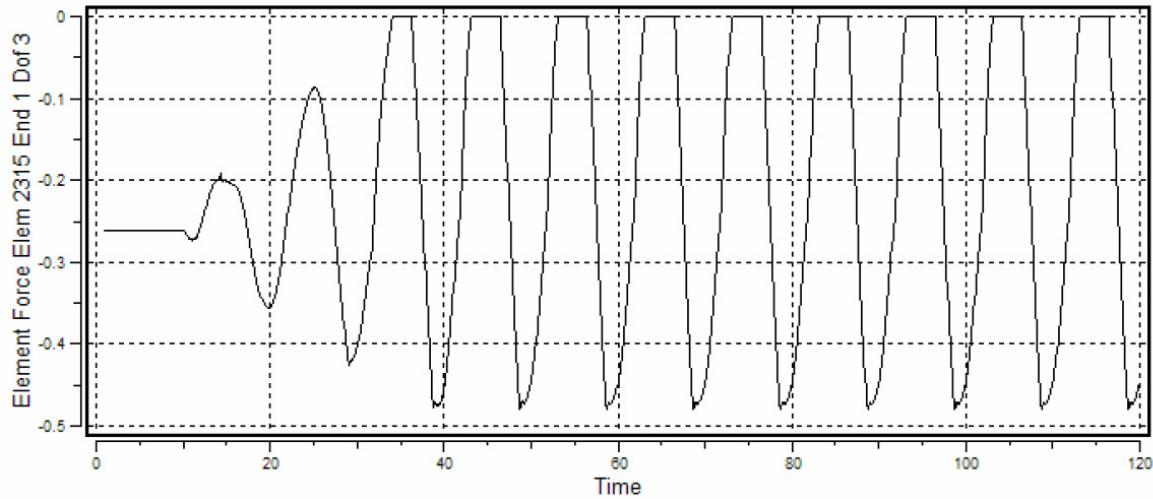


Figure 8.8: Contact force plot at stinger tip.

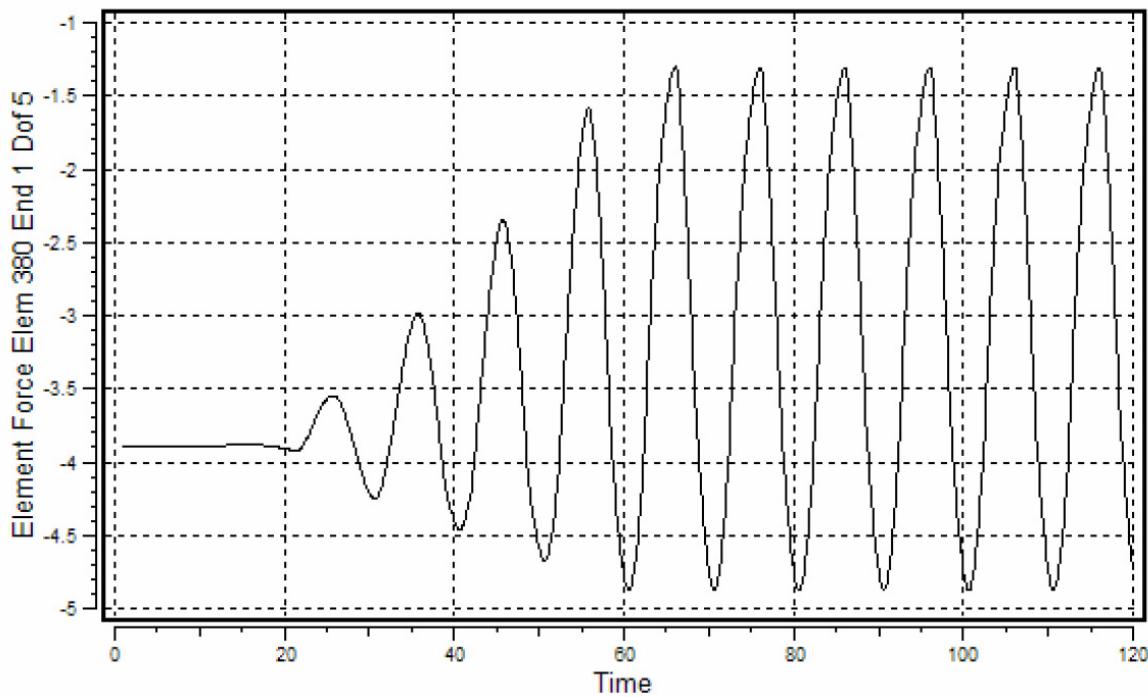


Figure 8.9: Dynamic bending moment plot at stinger tip.

### 8.3.2 Modelling

All the models reported herein have been established in the computer code SIMLA applying the following element types:

- PIPE31 - an elastic pipe element based on Euler-Bernoulli (plane surfaces remain plane) beam theory assuming a constant axial force and torque (st.Venant torsion) and linear curvature per element,

- CONT126 - a seabed contact element that includes a penalty spring representation of the soil vertical stiffness and a Coulomb friction model for the transverse directions.
- COMPIPE42 - a resultant based non-linear pipe element that allows describing non-linear user defined curves with (or without) hysteresis for the axial, torsion and bending characteristics.

The large displacement kinematics is described by a co-rotational scheme, see ([Mathisen, 1990](#)). Thus the coupling between torsion and bending as relevant to the torsion buckling problem relies on the co-rotational update of each elements coordinate system.

### 8.3.3 Validation of finite elements

First the finite element procedures capability to describe the torsion buckling feature was investigated using the Greenhill equation as a reference, see Eq. (8.1).

The FE model was based on studying a 100 m long straight beam according to [Fig. 8.10](#). A prescribed torsion rotation was applied in one end whereas the other end was kept torsionally fixed. The input data with respect to stiffness values are according to [Table 8.1](#)



Figure 8.10: Straight beam model

A mesh convergence study was firstly carried out with respect to the critical torsion moment concluding with an element length of 0.1 m. The comparison between the Greenhill formula (red line) and the results obtained by BFLEX2010 is presented in [Fig. 8.11](#) demonstrating excellent agreement with the Greenhill equation with respect to the critical torque.

### 8.3.4 Case study - elastic catenary model

When it comes to the structural stability of a flexible structure hanging in a catenary configuration subjected to simultaneous torque and tension, TDP represents the area of primary concern with respect to kink formation where torsion energy is basically transformed into bending energy. Firstly an elastic beam element model was established

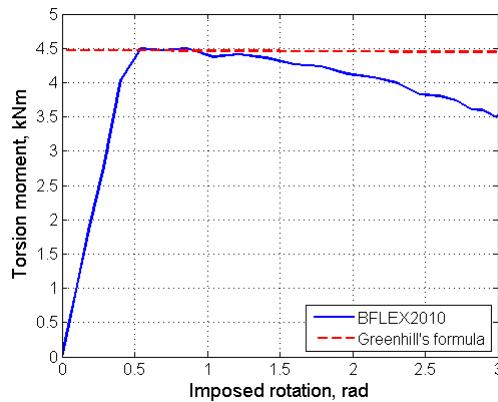


Figure 8.11: Torsion buckling of the straight beam model with 10 cm mesh size

Table 8.1: STUDY INPUT DATA

Parameter	Value
Top angle	89.6 (Deg.)
Cable diameter	0.12 (m)
Submerged weight	140.3 ( $Nm^{-1}$ )
Water depth	1000 (m)
Axial stiffness	409 (MN)
Bending stiffness	5.03 ( $kNm^2$ )
Torsion stiffness	1.0 ( $MNm^2$ )

for a catenary to validate its performance with respect to the Greenhill's analytical expression and with reference to the work by Neto and Martins (Neto and Martins, 2013). The model included a single catenary from a flat seabed to the sea surface using input data according to Table 8.1.

Using the catenary equation, the above gave a catenary length of 1007 m and an associated horizontal bottom tension of 986.4 N.

Three cases have been studied based on (Neto and Martins, 2013) to evaluate the torsion buckling capacity of the catenary and the role of the seabed friction, see Fig. 8.12:

- *Case A* - The model did not include the seabed section.
- *Case B* - The model included the catenary section of 1007 m plus a horizontal section of 600m resting on the seabed without friction.
- *Case C* - The model was the same as for the case B except that friction was activated by a Coulomb model with friction coefficients  $\mu_x = 0.4$  and  $\mu_y = 1.0$  in the axial and lateral directions, respectively.

In order to determine the critical torsion moment, prescribed torsion rotation  $\theta_x$  was

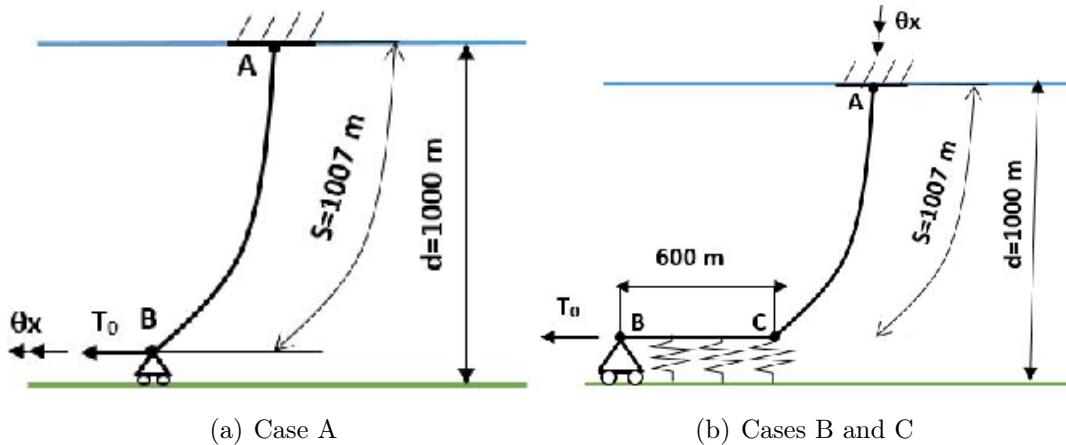


Figure 8.12: Study cases

applied either at TDP or at the free surface and the torsion moment was monitored as the constrained reaction. From Fig. 8.12 it is seen that for Case A,  $\theta_x$  was imposed in point B, while point A was kept fixed. For Cases B and C  $\theta_x$  was applied at the sea surface point A whereas Point B was kept rotationally fixed. An element length of 0.1 m was applied at the TDP section, then gradually increasing to maximum 1 m away from the activated section.

For the three cases studied, the analysis procedure included two phases. Firstly, a static analysis was performed to achieve the catenary configuration. Then, the torsion stability study was performed in the dynamic domain by prescribed rotation until torsion buckling was obtained.

Fig. 8.13 gives the results from Case A in terms of the torsion reaction moment as a function of prescribed rotation. The structure experience some deformations in the TDP region due to very low tension. When the peak value of the torsion moment is passed, torsion buckling occurs and the catenary forms into a kink as shown in Fig. 8.14.

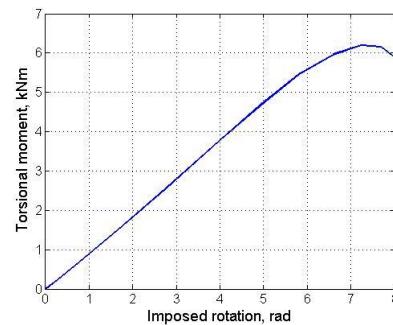


Figure 8.13: Torsion behavior for Case A

For Cases B and C the plots of torsion moment against prescribed rotation follow similar

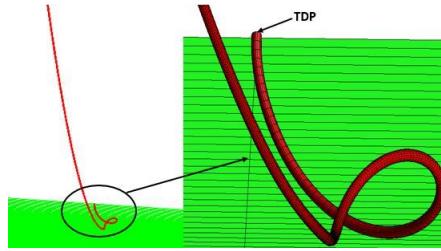


Figure 8.14: Deformation pattern for Case A

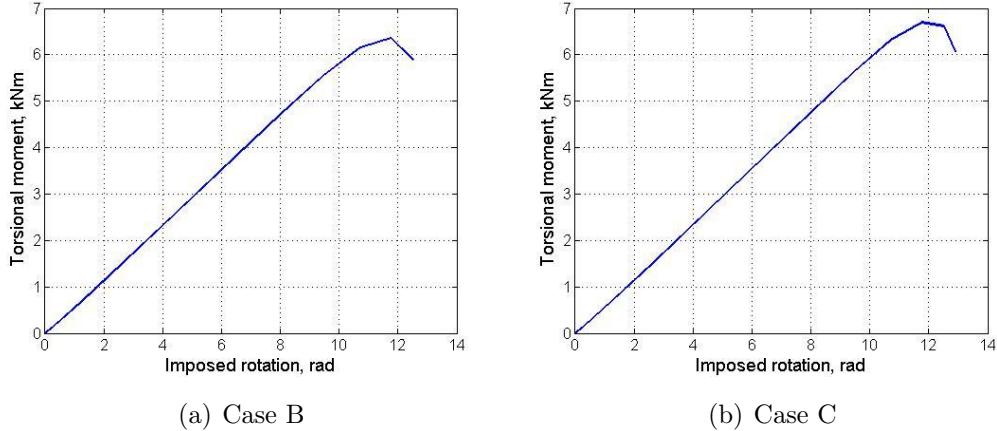


Figure 8.15: Torsion behavior for the case studies B and C

trends as for Case A, see Fig. 8.15).

However, the deformational behaviour of both Cases B and C are different from that of Case A because the position of the TDP varies during rotation. From Fig. 8.16 it can be noticed that for the frictionless Case B, the TDP is significantly shifted in transverse direction from the center line of the seabed section. This does not happen for the case study C , where the frictional forces restrain the TDP movement.

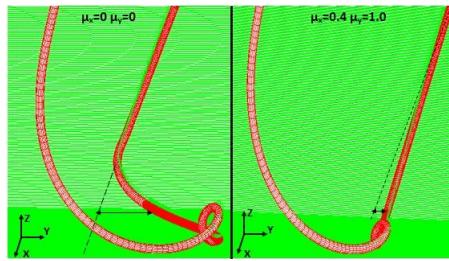


Figure 8.16: Deformation patterns for the case study B (left) and the case study C (right)

During the process, out-of-plane deformations of the catenary are initiated in the TDP region due to the low tension. Then, the torsion energy transfers into the bending energy. The length of the catenary section subjected to bending deformations (triggered by torsion) is hereafter termed the "activated length". The activated length includes

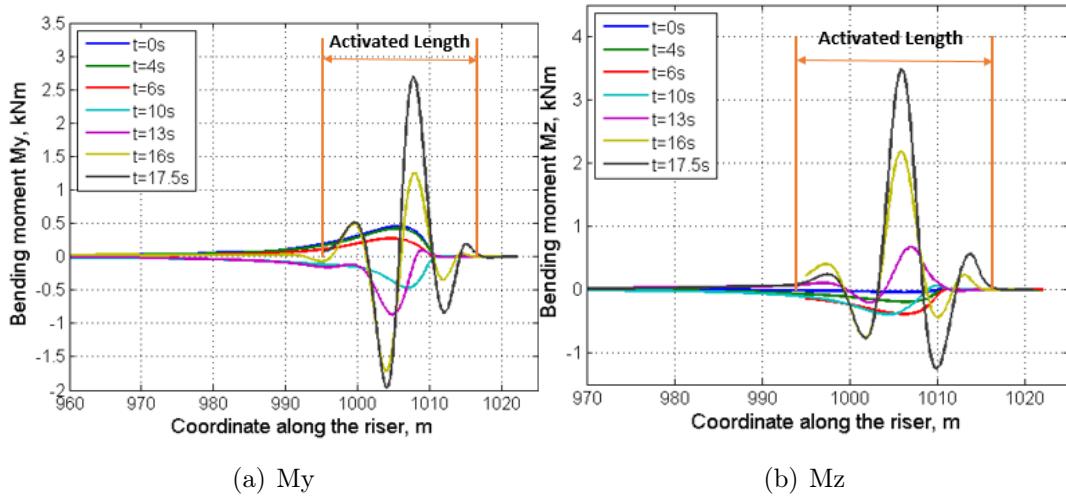


Figure 8.17: Bending behavior for Case C in the activated region

the section where the kink is formed and can be determined by studying the moments about the lateral transverse axes  $M_y$  and  $M_z$  along the catenary length.

To demonstrate how the activated length can be identified, Case C is chosen. Fig. 8.17 shows how the reaction bending moments  $M_y$ ,  $M_z$  are developed over time. The activated length of the catenary can be found as the section where  $M_z$  and  $M_y$  deviate from zero. For the case C with the TDP coordinate of 1007m, the activated length is located within the interval 990m-1015m. After buckling  $M_y$ ,  $M_z$  (black color curves) are substantially larger than for the time moment when buckling happens , which means that the structure have experienced large deformations increasing the size of the formed kink.

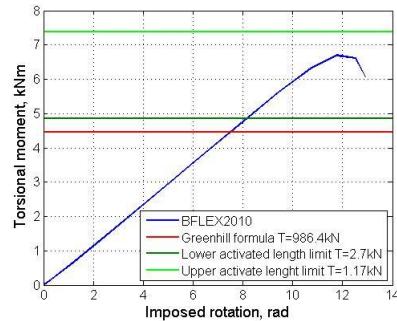


Figure 8.18: Influence of tension variation on critical torsion moment

Substituting the horizontal TDP tension  $T_0=986.4\text{ N}$  into the Greenhill's formula of Eq. (8.1), the critical torsion moment is found to be  $M_x=4.46\text{kNm}$ . However, this result is valid for a straight beam with constant tension. For a catenary configuration, the tension varies along the length. Additionally , the tension distribution in the activated section tends to increase significantly during the process of buckling onset. Therefore,

the critical torsion moment value is expected to be larger than the one obtained by the Greenhill's equation. Substituting the observed tension values at the beginning of the activated length of  $T=1.17$  kN and at the end of the activated length of  $T=2.7$  kN in the Greenhill's expression, gives the range of critical torsion moment within which the numerical value is expected to occur. This can be observed in Fig. 8.18. The result of the critical torsion moment obtained by BFLEX2010 is seen to be within the range.

### 8.3.5 Case Study - nonlinear moment-curvature model

The model was based on Case C described in the previous section, however, by applying COMPIPE42 instead of PIPE31 to describe a non-linear moment-curvature characteristic. Assuming that this is a cable structure and considering the low tension at TDP, the friction moment value will be governed by manufacturing effects such as shrinking of the outersheath, plastic deformations in the helix elements and also hydrostatic local deformation effects. Experience from testing has revealed that the friction moment can be significant even at zero tension, however, there is no unified procedure based on first principles or testing to predict the value for such cases. A case study is therefore carried out by varying the moment-curvature curves according to Fig. 8.19 where the range of the friction moment  $M_f$  is selected between 0.15 and 1.25 kNm.

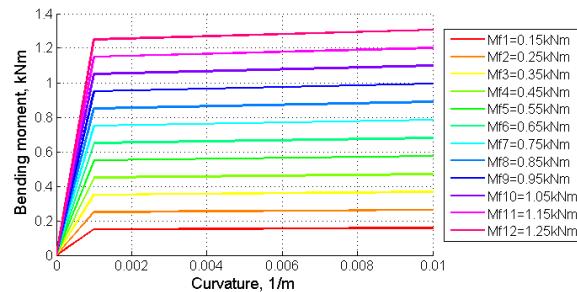


Figure 8.19: Material models for parametric study

Simulations were carried out by applying prescribed torsion rotation until buckling occurred according to the procedure outlined for Case C above. Due to the non-linear moment-curvature characteristics, the torque does not remain constant along the length, see Fig. 8.20. This is a direct result of the plastic curvature introduced and the value to be reported for design purposes is not obvious. In the following, the maximum value is applied as a reference.

The critical torsion moment is then computed for each value of  $M_f$  and plotted in Fig. 8.21 against the fraction  $\frac{M_f}{My_{max}}$ , where  $My_{max}$  is the maximum bending moment before prescribed torsion rotation is applied ( $M_z = 0$ ) for the linear elastic Case C.

It is seen from Fig. 8.21 that the friction moment significantly affects the torsion buckling capacity. The bigger friction moment  $M_f$ , the larger critical torsion moment.

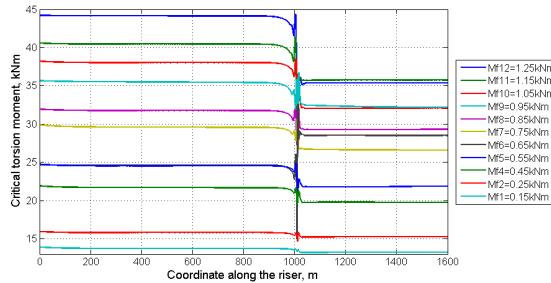


Figure 8.20: Critical torsion moment variations along the length of the non-linear model

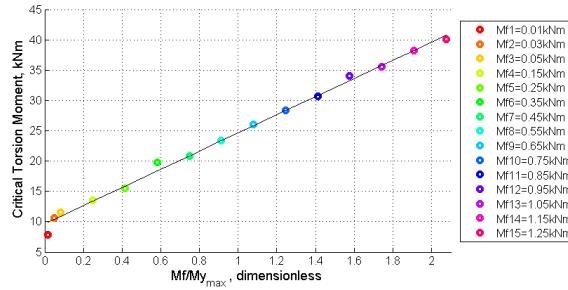


Figure 8.21: Torsion buckling results for COMPIPE42 model

It is also seen that for low values of  $M_f$  the critical torsion moment approaches the value computed for the linear elastic Case C (when  $M_f=0$ ).

### 8.3.6 Dynamic criterion for evaluation of kink formation

During installation, vessel motions will introduce a variation in the axial force and the curvature at TDP. In addition there will be a built in torsion moment that results from vessel heading, installation procedure and the cross-section torsion balance. Noting that kink formation is a process where torsion energy is transformed into bending energy and the fact that the maximum curvature is well established as a design criterion in the industry, the question then is whether it is possible to establish a curvature based design criterion that includes the effect of kinking for decision making.

The procedure proposed here is to first establish the maximum allowable curvature resultant based on the analysis procedure applied for Case C and defined as:

$$\kappa_t = \sqrt{\kappa_y^2 + \kappa_z^2} \quad (8.2)$$

The maximum resultant curvature that is found along the catenary can be then plotted versus the torsion moment utilization, see Fig. 8.22

The critical curvature value (when buckling occurs)  $\kappa_{crit}$  is then used as an indicator for prediction of kink formation. For the Case C model, if  $\kappa_t$  exceeds  $\kappa_t = 0.439 \frac{1}{m}$ ,

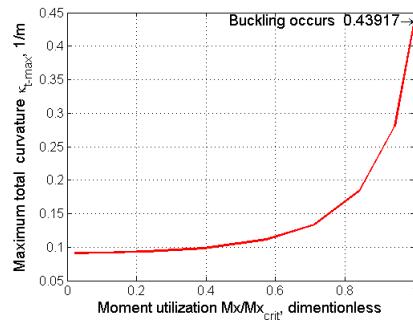


Figure 8.22: Development of the maximum value of the total curvature for elastic model

a kink will be formed in the TDP. For design purposes a safety factor would have to be included that needs to be established based on dynamic analysis to ensure that the maximum curvature does not grow due to plastic deformations as addressed in the following.

In order to investigate the application of the curvature as a measure for safety against kink formation, dynamic analyses have been carried out for the Case C model applying three different utilizations of the critical torque, 25 %, 50 % and 80 %. The analysis is performed in three steps, first the catenary configuration is established, then prescribe torque is applied and at last vertical heave motions are introduced over 3 regular periods. The heave motion was defined as:

$$a = a_0 \sin\left(\frac{2\pi t}{T_h}\right) \quad (8.3)$$

where,  $a_0$  is the heave amplitude and  $T_h$  is the heave period. The values for  $a_0$  were varied in the range 0.5-5m whereas  $T_h$  was kept fixed to 10sec.

Fig. 8.23-Fig. 8.25 below shows the maximum curvature resultant versus time for the three cases of torque utilization and for variable heave amplitudes. The red line represents the critical curvature value obtained by the prescribed rotation procedure  $\kappa_t$ . For all cases where the maximum curvature exceeded this line a kink was formed indicating that the proposed criteria is valid.

A similar exercise was carried out for the non-linear moment curvature model. However, the study included different values of initial torque which were taken to be 27 %, 40 %, 46 % and 54 % of the critical torque value. In addition only two load periods were investigated. The results in terms of maximum total curvature versus time is presented in Fig. 8.26 - Fig. 8.27, also including the red line to indicate the critical curvature value.

It is seen that for some cases the curvature amplitude grows which clearly indicates a need for more cycles in the analysis to conclude on whether or not the dynamic response

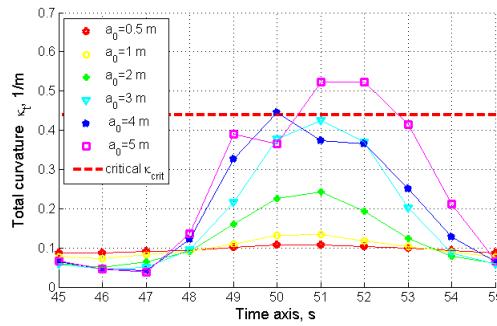


Figure 8.23: Time history - maximum total curvature -  $\frac{M_x}{M_x^{crit}} = 0.25$

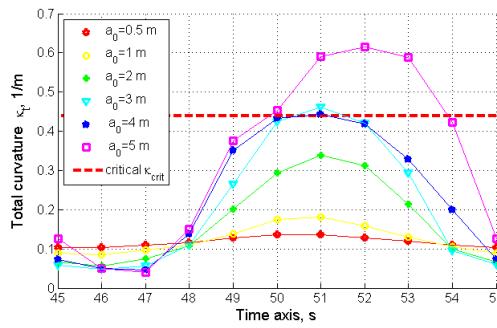


Figure 8.24: Time history - maximum total curvature -  $\frac{M_x}{M_x^{crit}} = 0.5$

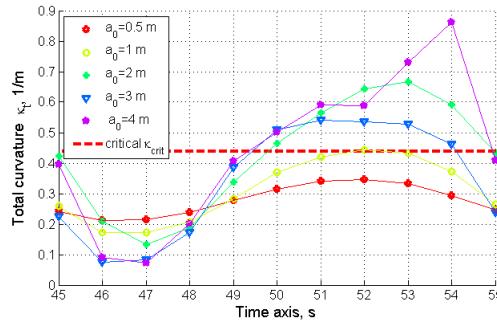


Figure 8.25: Time history - maximum total curvature -  $\frac{M_x}{M_x^{crit}} = 0.8$

stabilizes before the critical level is reached. This is due to plastic deformations resulting from the moment-curvature hysteresis that builds up gradually.

The following observations can be made from the above parametric study:

1. The torque utilization  $\frac{M_x}{M_x^{crit}}$  and heave motion amplitude  $a_0$  both influences  $\kappa_{t-max}$ .
2. The role of torque utilization in the kink formation process is much more crucial for the non-linear model as compared to the linear model. For instance, having the torsion utilization of around 50% and higher, it appears that a kink will be inevitably be formed for any value of heave amplitude as seen from Fig. 8.27.

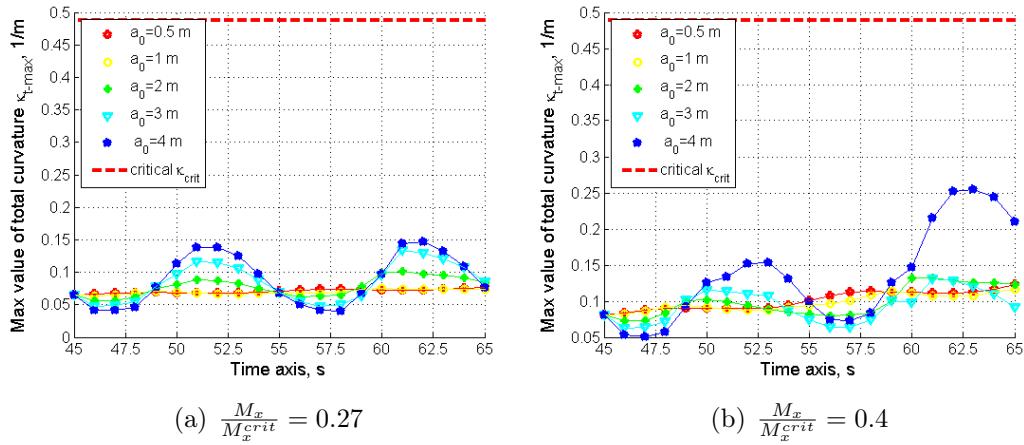


Figure 8.26: Maximum total curvature versus time

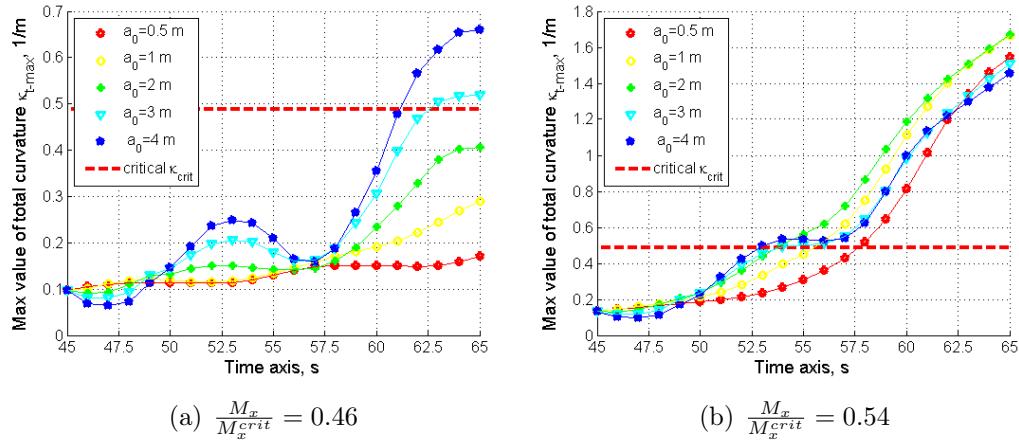


Figure 8.27: Maximum total curvature versus time

3. With respect to kink formation it appears that it is more conservative to apply the elastic model for lower torque utilizations as compared to the non-linear model. This is due to the fact that for the elastic model, a large heave amplitude motion can cause kink formation even at small torque utilizations. However, for larger values of  $\frac{M_x}{M_x^{\text{crit}}}$  (starting approximately from value of 0.5) it appears to be more reasonable with respect to safety perspective to apply a non-linear moment-curvature model.

Current design practice is to not allow compression at TDP which is a conservative approach with respect to avoiding buckling failure modes.

However, with respect to the torsion buckling failure mode this current industry practice can be argued. As can be noticed from Fig. 8.26, for the case of torque utilization of  $\frac{M_x}{M_x^{\text{crit}}} = 0.27$  can be operated without torsion failure with sufficient safety margin for all applied heave amplitudes. By looking at the axial force distribution for the heave

amplitude of 4m as presented in Fig. 8.28 a compressive force in the range of 9kN appears without indication of kink formation.

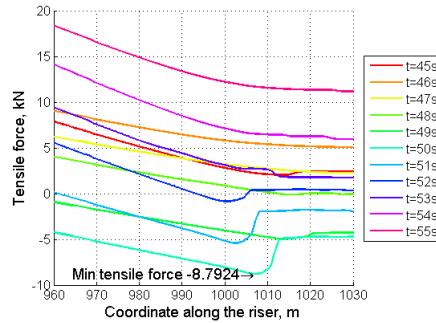


Figure 8.28: Axial force distribution for Heave amplitude  $a_0 = 4$  m,  $\frac{M_x}{M_x^{crit}} = 0.27$

This means that it is possible to have axial compression at TDP without kink formation. However this requires:

1. Accurate prediction of the inherent friction moment
2. Accurate prediction of the inherent torque to ensure sufficient low utilization of the critical torque
3. Establishment of the critical curvature associated to kink formation
4. Perform dynamic analysis using a non-linear moment curvature model of a sufficient number of cycles to prove that kink formations is not developed due to accumulated plastic deformations.

The above can be applied to propose design criteria and recommended analysis procedures for evaluating kink formation of offshore cables during installation.

The proposed procedure the includes the following steps:

1. Evaluate the installation route and focus on sections where the relative yaw rotation between the vessel and TDP is large.
2. Evaluate the torsion balance of the cross-section and calculate the  $\beta$  parameter.
3. On the basis of the above, calculate the inherent torque for the different installation scenarios at the critical route sections.
4. Calculate lower and upper bound values of the cross-section friction moment
5. Predict the critical curvature associated with kink formation for these cases.
6. Perform dynamic analysis with the built-in torque level and based on using a non-linear moment curvature model and use the dynamic maximum curvature at

TDP as a measure of kink formation

7. Use a sufficient number of cycles to prove that kink formations is not developed due to accumulated plastic deformations. If a stable value of the maximum curvature is obtained without kink formation and the standard maximum curvature design criteria is not exceeded, the sea state is acceptable.
8. If compression occurs during the dynamic analyses, the design check also need to include local helix buckling and birdcaging.

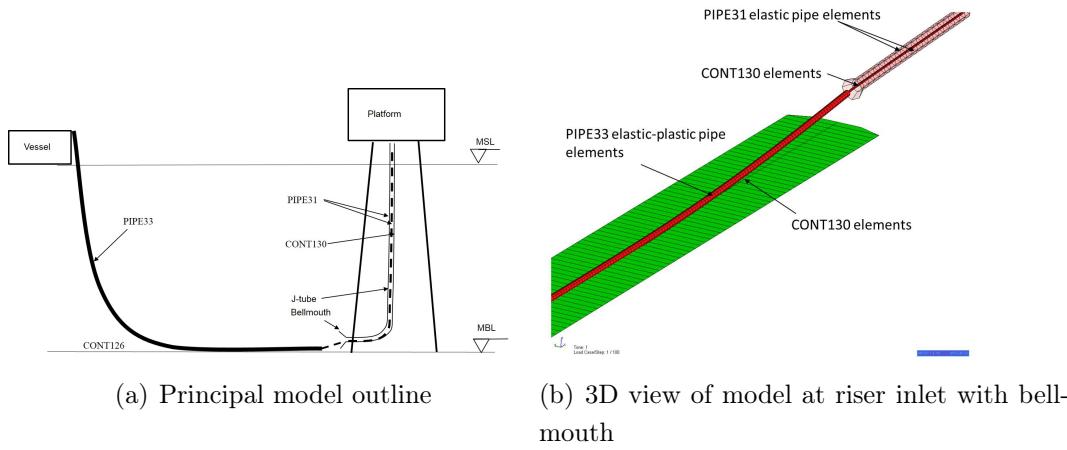


Figure 8.29: J-tube pull-in model

## 8.4 J-tube pull-in

### 8.4.1 Objectives

In order to connect flowlines to the platform infrastructure there are different principles in use, see (Bai and Bai, 2005). For fixed platforms it is possible to connect a rigid steel riser to the platform leg that allows pull-in of smaller diameter flowlines through the riser. Such risers are termed J-TUBES, It starts with a near horizontal entry section at some elevation above the seabed. The entry section also includes a BELLMOUTH which is a conical pipe section that eases the entry of the flowline into the riser. Then there is a slightly sloping straight section which is followed by a bend to enable a smooth transition into the vertical riser section, see Fig. 8.29 (a). The bend radius is so small that plastic deformations of the flowline will result in most cases (unless very small diameter flowlines). A wire is pre-installed inside the J-tube that allows the platform winch pull cable to be connected to a pull-head that is welded to the end of the flowline. Then the pull-in process is performed in steps by feeding the flowline from the installation vessel and simultaneous activation of the pull-in winch.

The task is to evaluate how large pull force that is needed to pull the flowline through the J-tube. The analysis was performed by SIMLA

### 8.4.2 Input data

A pipe in pipe flowline is to be connected to the platform topside by means of a J-tube pull-in procedure. The input parameters are shown in Table 8.2.

### 8.4.3 Modelling

A principal outline of the model is shown in Fig. 8.29 (a) and a 3D view of the stinger and touch down sections of the model are shown in Fig. 8.29 (b).

Table 8.2: INPUT PARAMETERS - SIMULATION OF J-TUBE PULL

Parameter	Value	Unit
Outer flowline outer diameter	0.3239	m
Outer flowline wall thickness	14.3	mm
Inner flowline outer diameter	0.186	m
Inner flowline wall thickness	14.3	mm
Total flowline dry weight	1.68	kN/m
Total flowline submerged weight	0.58	kN/m
Flowline yield stress	637	MPa
Stress at 10% strain	657	MPa
Water depth	250	m
J-tube inner diameter	0.632	m
J-tube entry high above seabed	2.7	m
J-tube total length	299	m
J-tube length horizontal section	22.0	m
J-tube bend radius	30.0	m
J-tube slope entry section	2.5	deg.
Pipe in pipe contact stiffness	1	MN/m/m
Pipe in piped friction factor axial	0.5	
Seabed vertical stiffness	39	kN/m/m
Seabed friction factor axial	0.63	

The model consisted of 400 PIPE33 elements for the outer flowline. The inner flowline was represented by 400 linear elastic PIPE31 elements connected to the same node system. The pull cable and J-tube was represented by 300 PIPE31 elements. Pipe in pipe and seabed contact was handled by CONT130 and CONT126 elements using a constant surface (penalty) stiffness in the normal directions and a Coulomb friction model for the tangential directions. The catenary section was not included. Instead the horizontal tension at TDP was applied as a constant tension load at the left end of the model.

The analysis was performed as a combination of static and dynamic steps. The model was initiated statically in horizontal mode positioned at the seabed. Prescribed displacements of the pull-cable and J-tube model parts were thereafter applied to introduce the J-tube geometry (with the pull cable inside). Then the pull-in procedure was initiated in dynamic mode, by activating the CONT130 elements and by vertical prescribed displacement of the upper pull-cable node.

#### 8.4.4 Results

Fig. 8.30 shows the pull force versus pull distance and Fig. 8.31 shows the stress distribution in the outer pipe at different stages of the pull-in. It is seen that there is a local maximum in the pull-force as the pull-head first meets the outer J-tube wall. Then a new maximum is reached as the plastically bent pipe experiences reversed bending resulting in tensile stresses at the inner surface. This is a consequence of the residual radius resulting from the plastic bending induced in the J-tube bend. Then the maximum value is reached due to the combination of plastic bending and weight at the end of the operation.

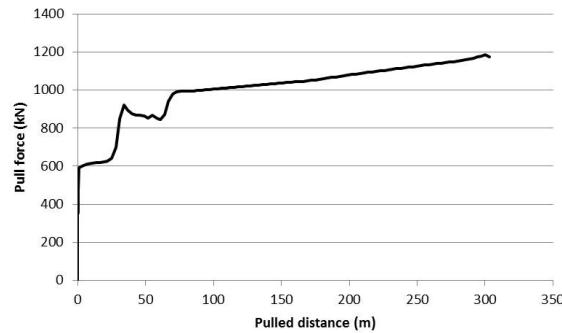


Figure 8.30: Pull-in force versus length of pull

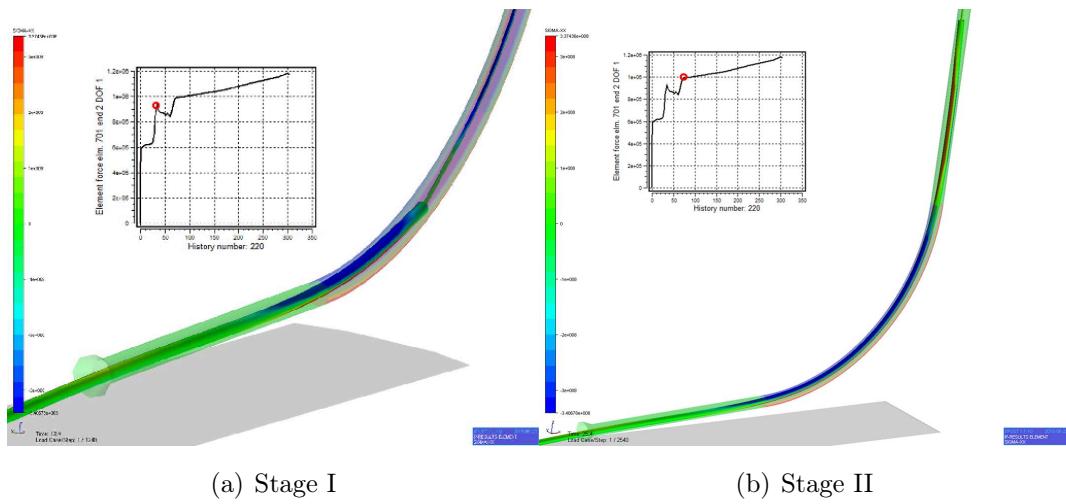


Figure 8.31: Pull force versus different stages of pull-in process

Table 8.3: INPUT PARAMETERS - SIMULATION OF FLOWLINE WALKING

Parameter	Value	Unit
Flowline outer diameter	0.16	m
Flowline wall thickness	15.0	mm
Flowline dry weight	0.53	kN/m
Flowline submerged weight	0.34	kN/m
Seabed vertical stiffness	1000	kN/m/m
Seabed friction factor axial	0.5	
Seabed friction mobilization displacement	5	mm

## 8.5 Pipeline walking

### 8.5.1 Objectives

Here the task is to investigate the pipeline walking phenomena that results from the combination of transient temperature variation and short flowlines which are not long enough to be fully anchored, see e.g. (M. Carr and Bruton, 2006) and (Chaudhury, 2010). This may cause the pipe to move towards the cool-end. In addition, the presence of dynamic tension variation from steel catenary risers or a seabed sloping downwards in the direction of the cool end may accelerate the process. A flowline with length 4 km resting on a flat seabed was studied using the computer code SIMLA.

### 8.5.2 Input data

The 4 km long flowline is exposed to  $100^{\circ}\text{C}$  temperature difference at the hot end. The non-dimensional temperature profile at different instants in time are shown in Fig. 8.32. Other input data are summarized in Table 8.3.

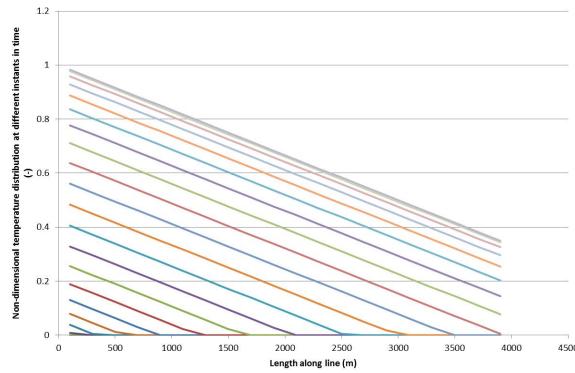
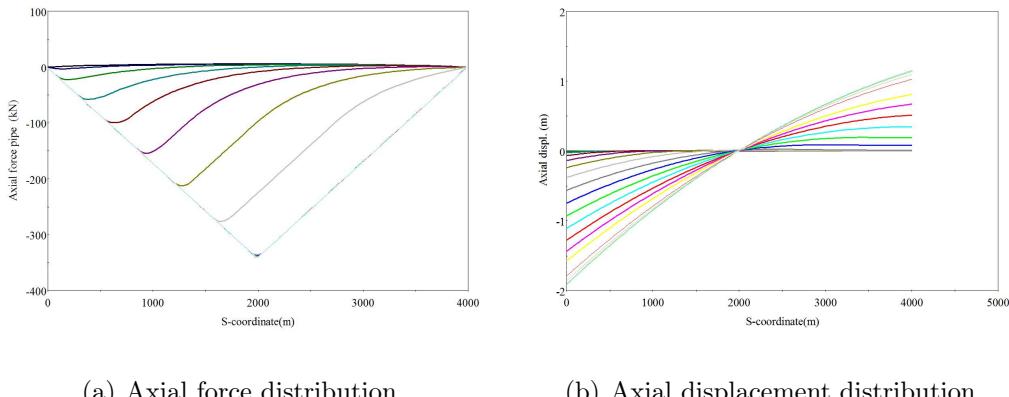


Figure 8.32: Non-dimensional transient temperature profiles



(a) Axial force distribution

(b) Axial displacement distribution

Figure 8.33: Axial force and associated displacements at different instants in time during start-up

### 8.5.3 Modelling

The pipeline was modelled by 200 PIPE31 elements and 201 CONT126 elements. A standard Coulomb model was applied with a friction coefficient 0.5 at mobilization displacement 5mm. The temperature profiles were applied throughout 4 cycles.

### 8.5.4 Results

The axial force and associated axial displacement distributions obtained by subsequent application of the temperature profiles are shown in Fig. 8.33. The triangular shape of all axial force profiles demonstrate that the flowline is not fully anchored. Taking a closer look at the displacement distribution by zooming the displacement profiles as in Fig. 8.34 (a), it is seen that the entire flowline will be exposed to small but positive relative displacements in the direction towards the cool end. Due to the hysteresis effect contained in the Coulomb model this will lead to permanent relative displacements between the pipe and the soil as seen in Fig. 8.34 (b). This again results in permanent movement towards the cool end for each cycle as seen in Fig. 8.35.

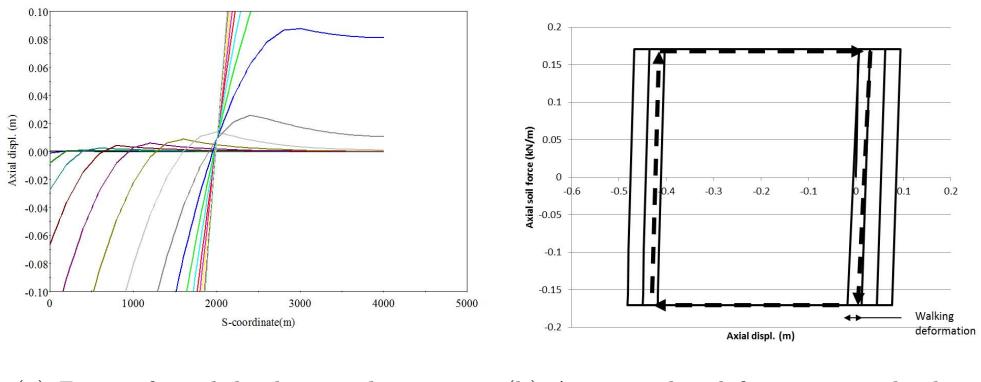


Figure 8.34: Explanation of walking physical process

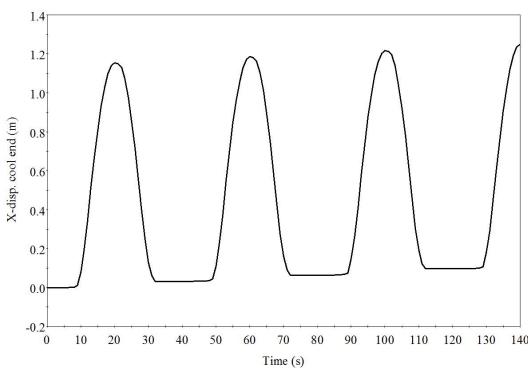


Figure 8.35: Displacement at cool end as function of time

## 8.6 Lateral buckling versus alternative soil models

### 8.6.1 Review of lateral soil models

The application of soil models for describing the lateral transverse pipe-soil interaction behaviour is a critical issue when dealing with high temperature pipelines resting on the seabed.

Traditionally such analyses are performed by using standard Coulomb models where the soil resistance is treated by a friction factor that attempts to capture the inherent soil mechanics.

Several alternative soil models related to lateral motions of pipeline resting on seabed have been developed for the following purposes:

- The evaluation of hydrodynamic stability of pipelines resting on the seabed and exposed to wave and current forces
- The evaluation of lateral buckling behaviour specially for pipelines resting on

## cohesive soils

For the hydrodynamic stability case the movement of the pipe is governed by the stochastic nature of the wave kinematics at the seabed during a storm event. This normally takes place as a number of small oscillations that causes the pipe penetration into the soil to increase. Then for the big wave event, large hydrodynamic lift and drag forces cause the pipe moving out of the trench into a new position and from where the process is restarted. The design criteria is governed by the maximum allowable pipe strain and/or displacement, see ([DNV, 2010](#)).

The nature of lateral motion due to global buckling is somewhat different, taking place in the form of subsequent large amplitude cycles following the variation in operating temperature (independent of hydrodynamic lift forces), ultimately leading to the build-up of soil berms at the side of the pipe that will increase the resistance.

With regard to hydrodynamic stability, the lateral soil models contain two main ingredients: Coulomb friction and passive soil force components. The model for passive soil force was developed by Verley and Sotberg ([Sotberg and Verley, 1992](#)) for sand, and by Verley and Lund ([Verley and Lund, 1995a](#)) for the clay model. The mathematical model of the passive soil resistance is created using simple dimensional analysis methods fitted to large scale laboratory test programs such as PIPESTAB ([Brennoddern et al., 1986](#)) and AGA ([Brennoddern et al., 1989](#)). Both models have been incorporated in ([DNV, 2010](#)) as recommended soil models for pipeline design. These models were also included in the PONDUS program developed by MARINTEK ([Sotberg et al., 1994](#)).

In general, the passive resistance of the soil has four stages of development, as described in ([DNV, 2010](#)):

1. An elastic region where the lateral displacement is less than typically 2% of the diameter. The upper limit of the passive resistance in this stage is denoted as FR1 .
2. A region where significant lateral displacement may be experienced, up to half the pipe diameter for sand and clay soils in which the pipe-soil interaction causes an increase in the penetration and thus in the pipe-soil resistance. The upper limit of soil passive resistance is called breakout resistance, FR2.
3. After breakout, the resistance and penetration decrease.
4. When displacement exceeds typically one diameter, the passive resistance and penetration may be assumed constant. The soil resistance at this stage is denoted as FR3 and is lower than FR1 and FR2.

In the elastic region, the initial passive soil resistance is linear elastic and no energy

is dissipated. During this stage, the initial pipe penetration does not change from the initial value. When exposed to cyclic motions in the plastic regime, the soil penetration will increase due to the accumulated plastic work (energy). The maximum lateral motion, where breakout occurs is defined as  $0.5D$  in (DNV, 2010) whereas in the PONDUS model this value is set to  $0.75D$ .

If the pipe continues to move in the same direction after breakout, some horizontal resistance in addition to friction will be present due to the soil mound being pushed ahead of the pipe (Sotberg et al., 1994). The accumulated work used to calculate the penetration is set to zero. The soil penetration is reduced, accordingly. DNV recommends this value to be half of the soil penetration at lateral motion 1D whereas in PONDUS a slightly different approach is used.

As shown by the works being part of the SAFEBUCKJIP (D.Brunton et al., 2006), the soil resistance resulting from cyclic lateral buckling motions in cohesive soils cannot be captured by the above models. This is a result of soil berm build up. Some research has been conducted to formulate the soil berm resistance, see (D.Brunton et al., 2006) and (Cardoso and Silveira, 2010), for such cases. Both proposed that the soil berm resistance can be formulated as friction load. (D.Brunton et al., 2006) proposed that additional berm resistance, can be added on top of residual soil resistance during the berm build-up. The number of cycles required to achieve this maximum berm resistance is five cycles (typically).

### 8.6.2 Objectives

The task in the following example is to investigate the consequences in terms of the resulting bending moment at the buckling apex of using the DNV, the traditional Coulomb and the SAFEBUCK models for a pipe resting on a clay soil. The analysis was carried out in SIMLA.

### 8.6.3 Input data and modelling

A 400 m long pipeline resting on a flat seabed was considered. The model included 200 elastic PIPE31 and 201 CONT126 elements, see Fig. 8.36. The cyclic motion due to temperature expansion was treated in a simplified way by prescribing an axial displacement in the ends ranging between 0.5 m to 1 m and throughout 5 cycles.

Other input data are according to Table 8.4.

### 8.6.4 Results

The soil force histories at the apex by applying the different soil model are shown in Fig. 8.37 (a) whereas the associated bending moment distributions obtained after 5 cycles

Table 8.4: INPUT PARAMETERS - SIMULATION OF GLOBAL BUCKLING

Parameter	Value	Unit
Flowline outer diameter	0.35	m
Flowline wall thickness	20.0	mm
Flowline dry weight	2.12	kN/m
Flowline submerged weight	1.09	kN/m
Undrained shear strength	5	kPa
Lateral friction factor	0.25	
Lateral friction mobilization displacement	50	mm

LATERAL BUCKLING ANALYSIS - DNV LATERAL MODEL TEST CASE

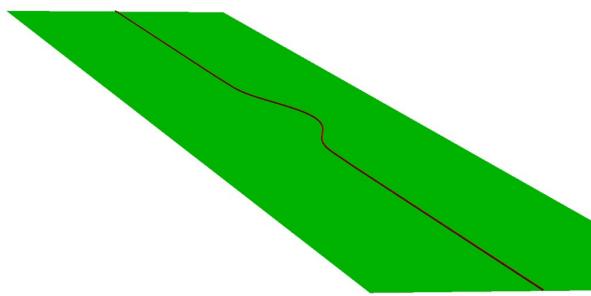
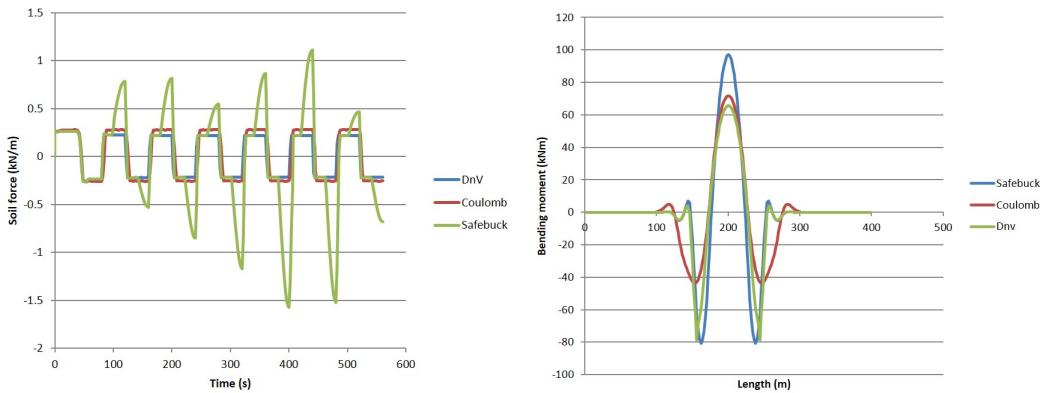


Figure 8.36: Model for pipeline lateral buckling

are shown in Fig. 8.37 (b). The increase in bending moment at the apex by applying the Safebuck model is evident. However, significant differences are also seen with respect to the moment distributions obtained by the DNV and SAFEBUCK models and the Coulomb model. This leads to the conclusion that soil models applied in such analyses need to be carefully selected.



(a) Cyclic soil resistance force at buckle apex (b) Bending moment distribution after last cycle

Figure 8.37: Comparison of lateral snaking results obtained for different soil models

## 8.7 Pipeline trawl-gear interaction loads

### 8.7.1 General remarks

Interference load due to trawl gear interaction in many cases represents the governing design criteria with respect to the maximum allowable free-span length in North Sea pipelines.

A typical trawl gear set-up is seen in Fig. 8.38 (a). The two trawl nets are kept open by the hydrodynamic lift force of two flat steel plates (trawl boards) at the sides which is connected to the vessel on either side by warp-lines. In this case two trawl nets are used and a clump-weight is attached in the middle to keep the trawl nets at the seabed. The clump-weight is pulled by a third warp-line. Sweep-lines are used to connect the trawl nets to the boards and clump-weight. A side profile of the clump-weight hitting the pipeline is seen in Fig. 8.38 (b). It includes a roller that is connected to a steel frame structure and including a bracket that may hit the pipeline.

The heaviest trawl gear operated today are used in the Arctic fisheries with steel masses up to 10000 kg (DNV, 2014) and hydrodynamic masses of similar magnitude. Combined with a towing velocity of 2–3 m/s over-trawling events may result in a severe utilization of the pipeline capacity. During the last decades, optimization of the fishery has resulted in geometry changes of the trawl gear and increased steel masses. This continuous development necessitates regular updating and calibration of the design interference loads.

Assessment of pipelines subjected to trawl gear loads is commonly divided into three parts according to the load characteristics and response analysis method. The first part focuses on energy absorption and denting of the cross-section due to the initial impact load. The succeeding part is termed the pull-over phase in which the global

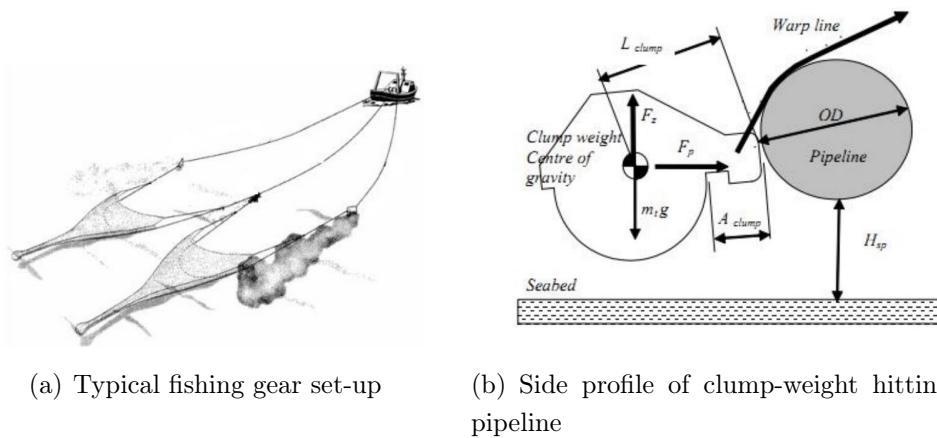


Figure 8.38: Typical trawl gear set-up (DNV, 2014)

pipe response due to the long-duration interaction forces is the main concern. This response type is dynamic and requires use of non-linear FE methods due to large lateral displacements, seabed contact, axial force changes and possible elasto-plastic material response. In current design practices simple physical models fitted to experimental results constitute the basis for the recommended pull-over loading. The third part is the rarely occurring hooking event where the trawl gear is assumed to get stuck between the seabed and the pipeline. Hooking design load effects may be obtained by static non-linear FE analysis of the pipeline subjected to a prescribed vertical lifting height. With regard to current design recommendations (DNV, 2014), the pull-over loading seems to have the largest potential for improvements.

Model testing has traditionally been the preferred method for determination of trawl loads. Research addressing the pull-over phase was initially based on laboratory tests and full-scale tests. Extensive testing was carried out in a Norwegian joint industry project (JIP) in the 1970s to study the interaction between pipelines and trawl gear, see (Kjeldsen and Moshagen, 1980). Regarding pull-over of trawl boards these tests revealed that spanning pipelines were subjected to larger forces than pipelines resting on the seabed, skew passings resulted in reduced loads as compared to perpendicular crossings and that hooking could be regarded as an unlikely occurring event. Nygaard (Nygaard, 1988) conducted model tests of protective subsea structures subjected to interference with trawl boards and beam trawls, in which the behaviour was found to depend on gear geometry, weight of the towed equipment and that small variations of the initial conditions could greatly change the interaction performance. An extensive model test program was conducted in 1990 (Nygaard, 1990), which forms the basis for the trawl board pull-over loads recommended by DNV today (DNV, 2014). Verley (Verley, 1994) conducted comprehensive parameter studies based on the recommended pull-over loads, and concluded that the practice of not burying pipelines with diameter larger than 0.4 m was non-conservative for pipes with large compressive axial forces.

Further improvements of the recommended design procedures for pull-over interaction has been the focus in many research efforts the last decades. see e.g. (Fyrilev et al., 1997). Removal of unnecessary conservatism by use of structural reliability methods on pull-over events was recently addressed by Amdal, Røneid and Etterdal (Amdal et al., 2011).

Numerical methods for response prediction of pipelines subjected to prescribed pull-over loads were introduced in work carried out by Bergan and Mollestad (Bergan and Mollestad, 1982) and Guijt and Horenberg (Guijt and Horenbergl, 1987) in the 1980s. A 2-dimensional simulation model was proposed by Horenberg and Guijt to predict a peak pull-over load and duration within 10% margin of model test measurements for a fixed pipeline and a beam trawl. A far more general approach was recently used by Søreide (Søreide and Igland, 2008) who demonstrated agreement between model tests and FE simulations of a clump-weight interacting with a pipeline resting on the seabed. In their work the recommended design load was found too conservative in case of soft seabeds. Based on a similar approach (Maalø et al., 2012) validated a clump-weight FE pull-over model against model tests of a fixed pipe section at low span heights. They demonstrated that the maximum value of the recommended design load could be halved if span flexibility was accounted for.

The behaviour of the trawl board and the associated hydrodynamic loads during interaction are far more complex than for clump-weights and beam trawls. The first step towards FE simulation of the rather complex interaction between pipelines and trawl boards was initiated by Teigen et.al. (Teigen et al., 2009). They investigated seabed proximity effects of the hydrodynamic mass and used flume tank experiments for determination of the forward-speed induced loads. Their work was later merged into FE simulations, but issues with the contact model between board and pipeline were reported (Longva et al., 2011). The performance of the contact model was recently improved and consistent behaviour with experimental tests was demonstrated, see (Longva and Sævik, 2013) and (Longva et al., 2013).

A computational strategy was outlined in terms of a FE formulation for frictional contact between pipelines and rigid three-dimensional bodies. This included:

- Development of a contact model for board-pipe interaction developed and validate its performance for the set of parameters.
- Validation of the numerical pull-over model for trawl boards against existing model tests considering variation of span height, towing line stiffness, board geometry, board mass, pipe support condition and towing velocity.
- Identify sensitive parameters which are not examined in previous work and study their influence on the pullover process.

The approach in the above was that if agreement is demonstrated for trawl boards it should be feasible also to achieve consistent behaviour for other and future trawl gear types as well. However, the conclusion from this exercise was that even if consistent interaction behaviour was demonstrated with respect to the model tests investigated, the same model cannot be extrapolated to other trawl types without performing hydrodynamic testing in each case to provide a hydrodynamic load model of sufficient accuracy. This is caused by the strong interplay between the hydrodynamic Coriolis-centripetal loads and the pressure-induced rotational damping which poses challenges with regard to the simulation input data, even for experienced users with good knowledge about trawl board pull-over interaction. Another difficulty is introduced by the board-pipe friction coefficient sensitivity for low span heights. However, if necessary input is provided by e.g. flume testing, the strength of the model procedure developed is in terms of enabling sensitivity studies to be performed, thus reducing the uncertainties in model tests.

### 8.7.2 Objectives

As noted above, the task of simulating trawl gear pipe interaction is easier for the case of pipeline clump-weight interaction and the purpose of this example is to demonstrate how this can be modelled in SIMLA and which results can be expected.

### 8.7.3 Input data

The input data were based on the same data as those used by (Maalø et al., 2012) except that here a 1000 m long pipeline with a free-span is investigated to capture the full global pipeline response. Other input data are according to Table 8.5

### 8.7.4 Modelling

A 3D outline of the model is shown in Fig. 8.39

The clump-weight roller and frame structure were modelled by linear elastic PIPE31 elements. The roller was attached to the frame by constraints allowing the roller to rotate during towing. The roller/warp-line bracket was modelled with one pipe element, where the length is equal to the vertical straight edge of the warp-line bracket of 0.2 m. The clump-weight was restrained from translation in lateral direction along the pipe, and from rotation about the vertical direction. Introducing these restrictions ensured that the clump-weight would collide with the pipeline at the pipe midspan.

The warp line was modelled in one lower and one upper part. The lower part that may come in contact with the pipeline consisted of 132 linear pipe elements. The element mesh size was reduced towards the warp-line bracket, to maintain a consistent contact

Table 8.5: INPUT PARAMETERS - SIMULATION OF CLUMP-WEIGHT

Parameter	Value	Unit
Velocity	1.95	m/s
Horizontal sweep-line angle	20	deg
Vertical warp-line angle	23	deg
Trawl net drag coefficient	20.8	$m^2$
Pipeline dry weight	1.81	kN/m
Pipeline submerged weight	0.85	kN/m
Pipeline outer diameter	0.35	m
Pipeline wall thickness	23	mm
Clump-weight mass	3624	kg
Clump-weight roller diameter/length	0.76/1.55	m
Applied added mass coefficient	1.0	
Applied drag coefficient	1.0	
Lateral soil friction factor	0.67	
Soil vertical stiffness	28	MN/m/m
Contact stiffness	8	MN/m/m
Warp-line axial stiffness	26	MN
Warp-line length	892	m
Sweep-line axial stiffness	40	MN
Sweep-line length	40	m
Free span length	50	m
Free span height	0.25	m

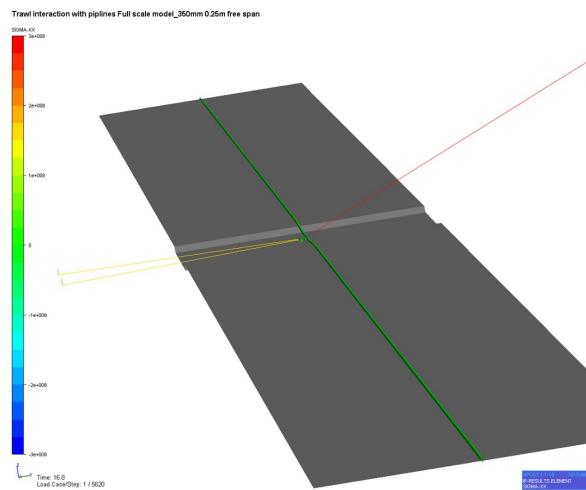
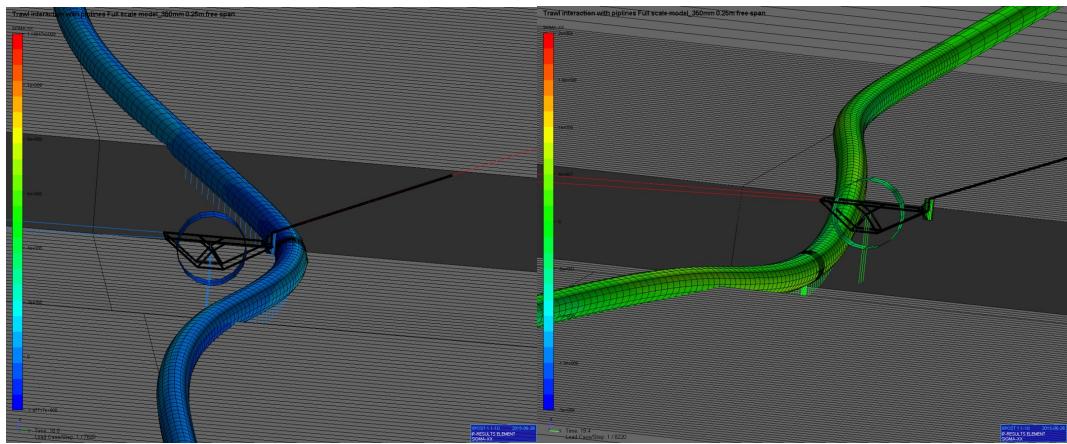


Figure 8.39: Model used to simulate clump-weight/pipeline interaction

interface. The lower part was 2.05 m long with 130 elements and the bending stiffness was set to a small value to allow warp-line bending during interaction.



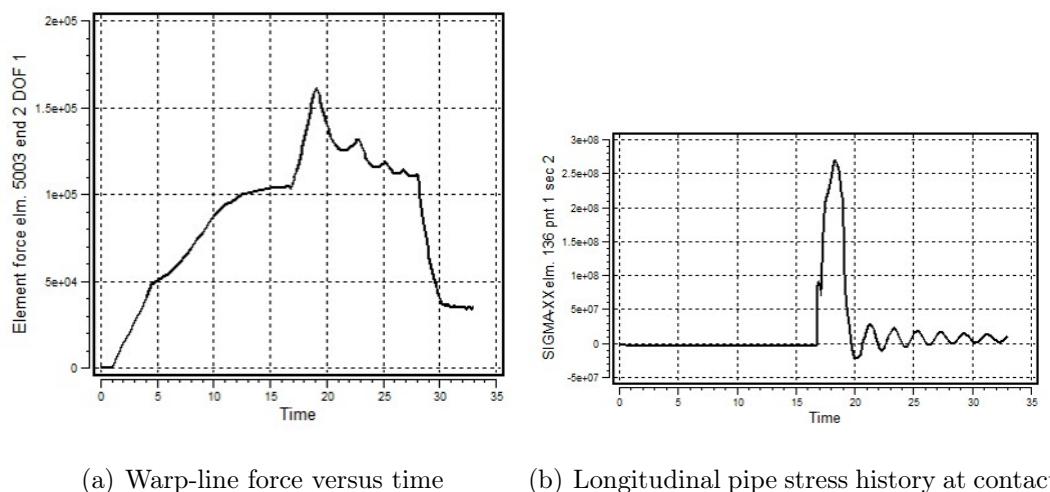
(a) Configuration during pull-over phase    (b) Configuration after release of clump-weight

Figure 8.40: Pipeline configuration during different pull-over phases

The sweep-lines were modelled by cable elements that have no stiffness in compression whereas contact between the bracket, warp-line and pipe was modelled by CONT164 roller elements. The pipeline section was 1000 m long, and included a free span section of 50 m at the middle of the pipeline. The element size was 14 m at the pipeline ends but gradually reduced when approaching the mid point of the pipeline, where the element size was 0.2 m. A total of 270 linear pipe elements were used. The pipeline ends were fixed and no internal pressure or temperature were applied.

### 8.7.5 Results

The results in terms of pipeline/clump-weight interaction during different instants in time is shown in Fig. 8.40 and plots of warp-line force and pipeline midpoint stress is shown in Fig. 8.41. It is seen that severe deformations occur, however, the total response is significantly less than one would obtain if the loads defined in (DNV, 2014) were applied as also noted by (Søreide and Igland, 2008) and (Maalø et al., 2012).



(a) Warp-line force versus time      (b) Longitudinal pipe stress history at contact point

Figure 8.41: Results in terms of warp-line force and pipe stresses

## Bibliography

- Amdal, L., Røneid., S., and Etterdal, B. (2011). Optimised design of pipelines exposed to trawl pull-over. In *Proc. of ISOPE'2011*, pages 130–134.
- Bai, Y. and Bai, Q. (2005). *Subsea Pipelines and Risers*. Elsevier, Oxford, UK.
- Bergan, P. and Mollestad, E. (1982). Impact response behavior of offshore pipelines. *J. Energy Resour. Technol.*, 4:325–329.
- Brennodden, H., Lieng, J., and Verley, R. (1989). An energy-based pipe-soil interaction model. In *Proc. of OTC'1989*.
- Brennodden, H., Sveggen, D., and Murf, J. (1986). Full-scale pipe-soil interaction tests. In *Proc. of OTC'1986*.
- Cardoso, C. and Silveira, R. (2010). Pipe soil interaction behaviour for pipelines under large displacement on clay soils - a model for lateral residual friction factor. In *Proc of OTC'2010*.
- Chaudhury, G. (2010). Managing unidirectional movements (walk) of hpht submarine flowlines during startup heating and shutdown cooling. In *Proc. of ISOPE'2010*.
- D.Brunton, D.White, Cheuk, C., Bolton, M., and Carr, M. (2006). Pipe/soil interaction behaviour during lateral buckling. *SPE journal of projects, Facilities and Construction*, 1(3):1–9.
- DNV (2010). *Recommended Practice DNV-RP-F109: On-bottom stability design of offshore pipelines*. DNV, Høvik, Norway.
- DNV (2014). *Recommended Practice DNV-RP-F111: Interference between Trawl Gear and Pipelines*. DNV, Høvik, Norway.
- DNVGL (2017). *Recommended Practice DNV-RP-F109, On-bottom stability design of submarine pipelines, May 2017*. Høvik, Norway.
- Fyrileiv, O., Spiten, J., and Mellem, T. (1997). Dnv'96, acceptance criteria for interaction between trawl gear and pipelines. In *Proc. of OMAE'1997*, volume 5, pages 91–98.
- Giertsen, E. (1998). *Matrixplot Users Guide*. MARINTEK, Trondheim, Norway. Memo.

- Giertsen, E., Taby, J., and Økland, O. (2017). *SIMLA Quick Start User Guide*. SINTEF Ocean, Trondheim, Norway.
- Greenhill, A. (1883). On the strength of shafting when exposed both to torsion and end thrust. In *Proc. instn. mech. engrs.*
- Guijt, J. and Horenbergl, J. (1987). Recent investigations concerning the effect of bottom trawl gear crossings on submarine pipeline integrity. In *Proc of OTC'1987*.
- Kjeldsen, S. and Moshagen, H. (1980). Fishing gear loads and effects on submarine pipelines. In *Proc. of OTC'1980*, pages 383–92.
- Koloshkin, E. (2016). Torsion buckling of dynamic flexible risers. Master thesis, NTNU - Department of Marine Technology.
- Longva, V. and Sævik, S. (2013). A penalty-based contact element for pipe and 3d rigid body interaction. *Engineering Structures*, 56:1580–1592.
- Longva, V., Sævik, S., Levold, E., and Istad, H. (2013). Dynamic simulation of subsea trawlboard pull-over interaction. *Marine Structures*, 34:156–184.
- Longva, V., vik, S. S., Levold, E., Istad, H., and Teigen, P. (2011). Dynamic simulation of free-spanning pipeline trawl board pull-over. In *Proc. of OMAE'2011*, pages 561–568.
- M. Carr, F. S. and Bruton, D. (2006). Pipeline walking - understanding the field layout challenges, and analytical solutions, developed for the safebuck jip. In *Proc. of OTC'2006*.
- Maalø, K., Alsos, H., and Sævik, S. (2012). Detailed analysis of clump-weight interference with subsea pipelines. In *Proc. of OMAE'2012*.
- Mathisen, K. M. (1990). *Large Displacement Analysis of Flexible and Rigid Systems Consdering Displacement-Dependent Loads and Non-Linear Constraints*. PhD thesis, Norwegian Institute of Technology, Trondheim, Norway.
- Neto, A. and Martins, C. (2013). Structural stability of flexible lines in catenary configuration. *mst*, 34:16–40.
- Nygaard, I. (1988). Improved design for protective subsea structures. In *Proc. of Subsea 88 international conference*.
- Nygaard, I. (1990). Trawl pipeline span interaction model tests. Final report Technical Report 511181.01-07, MARINTEK, Trondheim Norway.
- Sævik, S. (2007). *SIMLA - Test Manual*. MARINTEK, Trondheim, Norway.

- Sævik, S. (2017). *SIMLA Theory Manual*. SINTEF Ocean, Trondheim, Norway.
- Sævik, S. and Levold, E. (1995). High temperature snaking behaviour of pipelines. In *The Proceedings of The Fifth (1995) International Offshore and Polar Engineering Conference, The Hague, Netherlands, June 11-16, 1995*, volume II, pages 63–71.
- Søreide, T. and Igland, R. (2008). Advanced pipeline trawl gear impact design. In *Proc. of OMAE'2008*, pages 271–277.
- Sotberg, T., Holthe, K., Lund, K., and Igland, R. (1994). Pondus technical manual. Technical report, MARINTEK.
- Sotberg, T. and Verley, R. (1992). A soil resistance model for pipelines on sandy soils. In *Proc. of OMAE'1992*, pages 123–131.
- Teigen, P., Ilistad, H., Levold, E., and Hansen, K. (2009). Hydrodynamical aspects of pipeline overtrawling. In *Proc. of ISOPE'2009*, pages 435–442.
- Verley, R. (1994). Pipeline on a flat seabed subjected to trawling or other limited duration point loads. In *Proc. of ISOPE'1994*, volume 2, pages 128–134.
- Verley, R. and Lund, K. (1995a). A soil resistanse model for pipelines placed on clay soils. In *Proc. of OMAE'1995*.
- Verley, R. L. P. and Lund, K. M. (1995b). A soil resistance model for pipelines placed on clay soils.
- Verley, R. L. P. and Sotberg, T. (1994). A soil resistance model for pipelines placed on sandy soils. *J. of Offshore Mechanics and Arctic Engineering*, 116:145–153.

# Index

## A

ASCII ..... 34  
AUTOSTART ..... 57

## B

BEAM ..... 103, 105  
BFLEX2010POST  
– FAPLOT ..... 244  
BODY ..... 101, 124  
BONCON ..... 34, 39  
BOX ..... 70  
BREAKOUT\_Y ..... 191

## C

CABLE ..... 97, 125  
CLAY\_Y ..... 188  
CLOAD ..... 34, 40  
COMPIPE ..... 97, 120  
CONCRETE ..... 202  
CONEQ ..... 41, 46  
CONSTR ..... 34, 41  
CONT ..... 98  
CONTACT ..... 103, 178  
CONTINT ..... 34, 50  
CONTROL ..... 34, 52  
CORR ..... 69  
COSUPR ..... 34, 62  
COSURFPR ..... 34  
COSURFR ..... 62  
CROSSGEOM ..... 34, 68  
– BOX ..... 70  
– CORR ..... 69  
– CYLR ..... 69  
– GENERAL ..... 72  
– RECT ..... 70  
– TUBE ..... 69

CURLOAD ..... 34, 74  
CYLR ..... 69

## D

DRAIN\_Z ..... 200  
DROPS\_GRID ..... 34, 80  
DROPS\_HCOEF ..... 34, 83  
DROPS\_LOAD ..... 34, 86  
DYNCONT ..... 34, 74  
DYNRES\_ ..... 34, 75  
DYNRES\_E ..... 76  
DYNRES\_I ..... 78  
DYNRES\_N ..... 75

## E

EIGEN ..... 53  
eigen ..... 53  
ELASTOPLASTIC ..... 171  
ELCON ..... 34, 94  
ELDAMP ..... 34, 102  
ELECC ..... 34, 104  
– BEAM ..... 105  
– RADIUS ..... 108  
– STINGER ..... 105  
ELHIST ..... 35, 111  
ELLOAD ..... 35, 113  
ELMASS ..... 35, 114  
– BEAM ..... 103  
– CONTACT ..... 103  
– RALEIGH ..... 104  
ELORIENT ..... 35, 115  
ELPLOT ..... 238  
ELPROP ..... 35, 118  
ENPLOT ..... 247  
ENVRES ..... 134  
ENVRES\_E ..... 35, 134  
ENVRES\_I ..... 35, 135

ENVRES_N .....	35, 134
EPCURVE.....	172
 <b>F</b>	
FAPLOT .....	244
FATPROP .....	35, 136
FEED.....	35, 137
FEEDCONEQ .....	41, 49
FILEINIT.....	54
FLOWLOAD .....	35, 140
FROMFILE.....	90
 <b>G</b>	
GENERAL .....	72
general.....	91
GENSPRING .....	129, 202
GEOM.....	35, 142
GLOBAL .....	42, 47
GLPLOT .....	242
GNPLOT.....	240
 <b>H</b>	
HEAD .....	35
HEADING.....	148
HISTCONEQ .....	41, 48
HLA .....	35, 148
HLAPLOT.....	35, 149
HLAVIS.....	35, 152
HSHEAR.....	100
HYCURVE .....	174
HYDROPRO.....	35, 153
HYPERELASTIC .....	200
 <b>I</b>	
INIJLAY.....	55
INISLAY.....	56
INISTR.....	35, 164
IPPLOT .....	237
IRREGULAR.....	87

IRREGULARDNVRPF109CURR ...	89
IRREGULARSTANDARDCURR ...	88
ISOCONTACT .....	182
ISOHYCURVE .....	175
ISOKXYCONTACT .....	183
 <b>J</b>	
JOINTPR_APPLY .....	35, 165
JOINTPR_DEFINE .....	35, 166
 <b>L</b>	
LINEAR.....	170
LINSPRING.....	201
LOCAL .....	42, 46
LOCSL .....	47
 <b>M</b>	
MATERIAL.....	35, 168
– BREAKOUT_Y.....	191
– CLAY_Y.....	188
– CONCRETE.....	202
– CONTACT .....	178
– DRAIN_Z.....	200
– ELASTOPLASTIC .....	171
– EPCURVE.....	172
– GENSPRING .....	202
– HYCURVE .....	174
– HYPERELASTIC .....	200
– ISOCONTACT .....	182
– ISOHYCURVE .....	175
– ISOKXYCONTACT .....	183
– LINEAR .....	170
– LINSPRING .....	201
– R_CONTACT .....	183
– RESULTANT .....	176
– SAND_Y.....	190
– SEA .....	175
– SPRING .....	201
– UNDRAIN_Z.....	198

MOVE_GROUP .....	35	RS PLOT .....	247		
move_group .....	203				
<b>N</b>					
NOCOOR .....	35, 204	SAND_Y .....	190		
NODPROP .....	35, 207	SDDSTART .....	60		
NOORIENT .....	35, 208	SEA .....	100, 175		
NO PLOT .....	236	SEALO .....	35, 218		
NR PLOT .....	245	SECONDORDER .....	90		
<b>O</b>					
ORREGULAR .....	232	SIMLA .....	29, 35, 219		
<b>P</b>					
PDISP .....	41	SIMPOST .....	235		
PELOAD .....	35, 209	- ELPLOT .....	238		
PILOAD .....	35, 210	- ENPLOT .....	247		
PIPE .....	97, 119	- FAPLOT .....	244		
prefix .....	30	- General .....	235		
- dyn .....	30	- GLPLOT .....	242		
- ldat .....	30	- GN PLOT .....	240		
- raf .....	30	- Input data .....	235		
- sif .....	30	- IP PLOT .....	237		
- slf .....	30	- NO PLOT .....	236		
- sof .....	30	- NR PLOT .....	245		
		- RS PLOT .....	247		
		- VIVFAT .....	248		
		SPECIAL .....	45		
R_CONTACT .....	183	SPRING .....	100, 201		
RADIUS .....	108	STINGER .....	105		
RALEIGH .....	104	STRESSFREE .....	53		
RAO .....	43				
RAOPROP .....	35, 211	<b>T</b>			
READTRF .....	35, 214	TABLE .....	35, 222		
RECT .....	70	THIST .....	35, 223		
REEL .....	35, 216	TIMECO .....	35, 226		
REGULAR .....	232	TIMEINIT .....	54		
RESTART .....	53	TIMEINIT SDD .....	56		
RESULTANT .....	176	TLOAD .....	35, 228		
ROLLER .....	127	TRAWLBOARD .....	154		
		TUBE .....	69		
<b>U</b>					
UNDRAIN_Z .....	198				

UNITS ..... 35, 228  
USERCONTACT ..... 126

## V

VISRES ..... 35, 229  
VIVFAT ..... 248

## W

WAVE ..... 43  
WAVELOAD ..... 35, 230  
– FROMFILE ..... 90  
– GENERAL ..... 91  
– IRREGULAR ..... 87, 232  
– IRREGULARDNVRPF109CURR ..... 89  
– IRREGULARSTANDARDCURR ..... 88  
– REGULAR ..... 232  
– SECONDORDER ..... 90