NTNU – Trondheim
Norwegian University of
Science and Technology

Analysis of Ice-Induced Vibrations and Comparison with Full-Scale Experimental Data

Andreas Nordby Meese

Marine Technology
Submission date: July 2013
Supervisor: Jørgen Amdahl, IMT
Co-supervisor: Morten Bjerkås, Reinertsen AS

Norwegian University of Science and Technology
Department of Marine Technology

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

MASTER THESIS 2013

for

Andreas Nordby Meese

Analysis of ice-induced vibrations and comparison with full-scale experimental data

Analyse av is-induserte vibrasjoner og sammenligning med fullskala, eksperimentelle data

Offshore structures subjected to the action of drifting ice floes may experience several kinds of interactions with the ice feature. This ranges from quasi-static response from ductile crushing of the ice to vibration lock-in effects. The forces experienced by an offshore structure during lock-in are magnified if steady-state vibrations arise. The vibrations disturb working conditions on the structure. They may also provide the critical loading condition in the design. Failures due to fatigue have been reported.

The scope for the present project/MSc work is to generate or further develop existing models of a cylindrical lighthouse structure subjected to ice loads. The candidate shall establish force time series which generates the vibration lock-in effect. The numerical analyses shall be compared with actual test results.

Supervisors: Prof. Jørgen Amdahl (NTNU), Dr. Ing. Morten Bjerkås (Reinertsen AS)

The thesis should be delivered at the Department of Marine Technology within the 20th of July.

Jørgen Amdahl
Main supervisor

Abstract

In the present work, a model which estimates the fatigue life for a cylindrical structure interacting with ice is developed. The model is based on the lighthouse of Norströmsgrund, and estimates the fatigue damage for one season. Fatigue is the long-term, cumulative effect of cyclic loads. Cumulative fatigue damage is found by using the Miner-sum together with an SN-curve. Fatigue of the cross-section located 7.5 m above the seabed is investigated. Failure of this particular part is in the model assumed to cause global failure of the lighthouse. The cross-section consists of concrete with reinforcing steel bars.

Different cases are used when estimating the fatigue life. First, an extreme case is considered, assuming continuous cycles the entire season. The next case uses onset criteria for cyclic load conditions. Lastly, the field-observed number of cycles is used. The model produces a total number of cycles ranging from 541008 to 40950144. Field-observed number of cycles is for the considered season 5090. Thus, the model overestimates the number of cycles. Fatigue life is found to be 7 years for the extreme case. The methods using proposed onset criteria give a fatigue life of 31, 54, 61, 70 and 239 years, depending on the method. For the observed number of cycles the time to failure is estimated to be 4911 years. A fatigue life of 7 years is highly unlikely, while 31 to 70 years are considered to be underestimations. This is based on the fact that lighthouse has been operational since the 1970s. Models used in design should always overestimate damages.

The model checks if cyclic load conditions are probable using environmental parameters. Data from the winter of 2002/2003, measured at the Luleå Airport and the lighthouse of Norströmsgrund, is utilized. Cyclic loads are associated with steady-state vibrations and structural resonance. The frequency of the ice crushing against the structure is for this condition seen to be close to the lowest natural frequencies of the structure. This phenomenon is termed frequency locked-in crushing, abbreviated FLC. Three different approaches are applied by the model to check for possible FLC conditions. The first uses standardization codes, while the second utilizes dimensionless groups based on ice thickness, ice velocity, structural diameter and structural frequency. Lastly, case-specific methods for Norströmsgrund lighthouse are implemented in the model.

Ice properties, such as the thickness, uni-axial compressive strength, drift speed and drift direction are estimated using standardization codes and established methods.

In addition, a single-degree of freedom model was established. Input values were taken from previous established FEM models of the lighthouse. The model used the Newmark-beta method, and was compared to the in-situ measured lighthouse response. It was seen that the full-scale response can be fairly represented by the use of simple models.

Acknowledgements

I would like to thank Dr. Ing. Morten Bjerkås at Reinertsen for providing me with guidance, feedback on results, and teaching me to write and present results in a scientific manner.

Professor Jørgen Amdahl at the Norwegian University of Technology and Science (NTNU) is to be thanked for valuable guidance and comments.

Dr. Ing. Hagbart Alsos at Reinertsen is also to be thanked for guidance and comments.

Recognition goes to Corporate Graduate Halvor B. Lindstad for proof reading this thesis.

Lastly, I would like to thank my parents, for helping me recover after suffering from epilepsy, making me able to finish my thesis.

Contents

List of Figures

List of Tables

Notation

Abbreviations

API American Petroleum Institute

AWI Alfred-Wegener-Institut

CAE Computer-aided engineering

CC Continuous crushing

CSA Canadian Standards Association

DC Ductile crushing

DNV Det Norske Veritas

EC2 Eurocode 2

EM Electromagnetic

FDD Freezing degree days

FEM Finite element method

FFT Fast Fourier transform

FLC Frequency locked-in crushing

GoB Gulf of Bothnia

IC Intermittent crushing

ISO International Organization for Standardization

LOLEIF Low Level Ice Forces Project

LTU Luleå University of Technology

MWL Mean water level

NTNU Norwegian University of Technology and Science

SDOF Single-degree of freedom

SMHI Swedish Meteorological and Hydrological Institute

SS Steady-state

STRICE Measurements on Structures in Ice Project

VIV Vortex-induced vibrations

w.r.t. With respect to

Greek Symbols

β A constant value between 0 and 1 in the Newmark-beta method

$\Delta\sigma$ Stress range

ΔF Double force amplitude

δ Miner-sum. Value between 0 and 1

δ_{wl} Waterline displacement

$\dot{\sigma}_c$ Ice stress rate

γ A constant value between 0 and 1 in the Newmark-beta method

γ_s Material factor for reinforcing steel, equal to 1.15

$\phi(z)$ Mode shape function

ϕ_{nC} The normalized modal amplitude at the ice action point

σ_0 Reference value for compressive strength

σ_c Ice compressive strength

σ_{max} Maximum stress in a cycle

σ_{min} Minimum stress in a cycle

σ_m Mean stress in a cycle

θ A coefficient equal to $40 \cdot 10^6$ kg per metre-second

θ_{ice} Ice drift direction

θ_t Duration of one FLC event

θ_{wind} Wind drift direction

ξ_n Total damping of the eigenmode as a fraction of critical

Roman symbols

$\ddot{u}$ Structural acceleration

$\dot{u}$ Structural displacement velocity

$\varnothing$ Reinforcing bar diameter

A_0 Reference contact area

A_{cs} Total cross-sectional area

a_h Minimum distance between reinforcing bars in same layer

A_n Nominal contact area

A_s Cross-sectional steel area

C_{iw} Ratio between ice and wind velocity

C_R Ice strength parameter

D Structural diameter at the waterline

f_1 Lowest translational frequency

F_C Static load component

F_D Dynamic load component

F_G Global ice force

f_n The natural frequency

f_n Translational frequency for mode number n

h_1 Reference thickness of 1.0 m

h_{ice} Ice thickness

I_z Second moment of area

k_{wl} Waterline stiffness

L_{crack} Ice crack length, i.e. distance between the structure and the point where the ice fails globally

L_{design} Design load

$m(z)$ Mass distribution along z-axis

M_d Design moment

M_n The true modal mass

N_i Number of cycles to failure for a given stress range

n_i Number of cycles for a given stress range

p_G Global ice pressure over nominal contact area

q Scaling factor in the force sawtooth-function

R Mean pipe radius

r_i Inner radius of a pipe

r_o Outer radius of a pipe

T Period

T_{air} Ambient air temperature

$t_{fatigue}$ Fatigue life in years

T_m Ice melting temperature

u Structural displacement

V_{10} Wind velocity 10 m above surface

$v_{2,h}$ Upper boundary for FLC as a function of ice thickness and ice velocity

$v_{2,T}$ Upper boundary for FLC as a function of temperature and ice velocity

$v_{i,tr,1}$ Lower velocity limit for intermittent crushing

$v_{i,tr,2}$ Upper velocity limit for intermittent crushing

V_{ice} Ice drift velocity

$\mathbf{C}$ Structural damping matrix

$\mathbf{K}$ Structural stiffness matrix

$\mathbf{M}$ Structural mass matrix

1 Introduction

As the industry of extraction and production of hydrocarbons moves further north, new engineering guidelines for structural design are required to operate in more severe conditions. In ice-infested waters, the ice-structure interaction is an imminent problem. Sea ice, icebergs and wind often induce large loads on a structure, and structural damage is a problem. Structures have to withstand highly dynamic actions, and some structures have been reported to fail during the first winter after installation.

There are mainly two kinds of structural failure modes; failure by rupture and failure by fatigue. Rupture can take place due to overloading or a high impact force, and can be an unstable fracture or a ductile buckling. An iceberg is a typical cause of impact damage. Fatigue is caused by cyclic stresses acting on the structure over longer periods of time. This is a cumulative effect of many load cycles, which can cause failure of individual components, or in worst case global failure. An example of a platform which has failed to maintain operational due to ice action, is a Russian platform, "Lunskoye-A". The platform experienced difficult conditions the first winter after installation. An offshore structure which has collapsed due to overloading, was the lighthouse of Nygrån. After 10 years of service, the lighthouse failed in late April 1969. The lighthouse was found to have failed by rupture 1 m below the MWL (Erntsons and Kjellgren, 1969), as shown in Fig. 1.2. Loads acting on offshore structures may be divided into two categories; operational loads and environmental loads. Operational loads are due to the structural weight, loading/offloading of equipment and buoyancy to mention a few.

Figure 1.1: Lunskoye-A. Photo courtesy of Gazprom.

Air temperature, wind velocity and ice thickness are important parameters when it comes to forecasting of ice conditions on a specific site. Given those parameters, it would be of great value if one could estimate structural damage, i.e. due to wind and ice interaction on an offshore structure located in arctic regions. Given the estimated damage, it would be easier to design structures to be installed at the site.

This thesis will investigate ice forces, fatigue damage and response of the Norströmsgrund lighthouse. This lighthouse is located in the Gulf of Bothnia (GoB), and has been a valuable source for collection of data concerning ice-structure interaction.

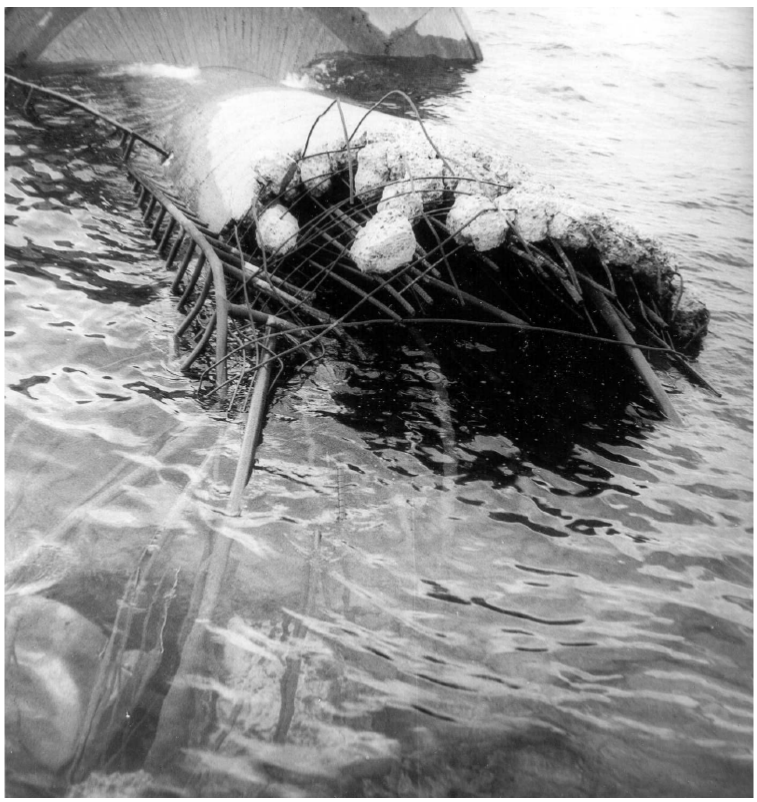


Figure 1.2: Rupture surface of Nygrån lighthouse. Photo courtesy of Lehmann (2010).

2 Background

Vertical structures interacting with level ice sheets typically cut through ice by crushing. A situation of ice crushing against a structure is shown in Fig. 2.1. This generates loads which build up as the ice pressure is increased. This is followed by an abrupt fall in contact load when the ice fails to sustain the pressure on the contact surface. The level of pressure the ice can sustain varies with (among several parameters) ice thickness, strain rate, ice temperature, compressive and flexural strength.

Figure 2.1: Ice crushing against the Norströmsgrund lighthouse. Photo courtesy of Kari Kolari.

2.1 Ice-structure interaction

The ice-interaction with the structure causes lateral deflections. These deflections can either be static, dynamic, or a combination of the two. This depends on several parameters, and it is common to divide the ice loading into three different regimes. Corresponding to three different load regimes, there are three different response regimes. This is shown in Fig. 2.2. Note that creep, which occurs for very low ice velocities, is in most cases not of concern, and is therefore not included in Fig. 2.2. The ice velocities $v_{i,tr,1}$ and $v_{i,tr,2}$ shown in Fig. 2.2 are called transition velocities, i.e. the boundaries between different response regimes. One of the first to introduce the distinct definitions of response due to ice-structure interaction was Sodhi (1991). This is reported and supported by the use of laboratory investigations. The different regimes are listed in Tab. 2.1. The first regime is characterized as quasi-static. The structure is displaced with nearly zero acceleration, before it is released at maximum displacement, i.e. when the ice sheet fails.

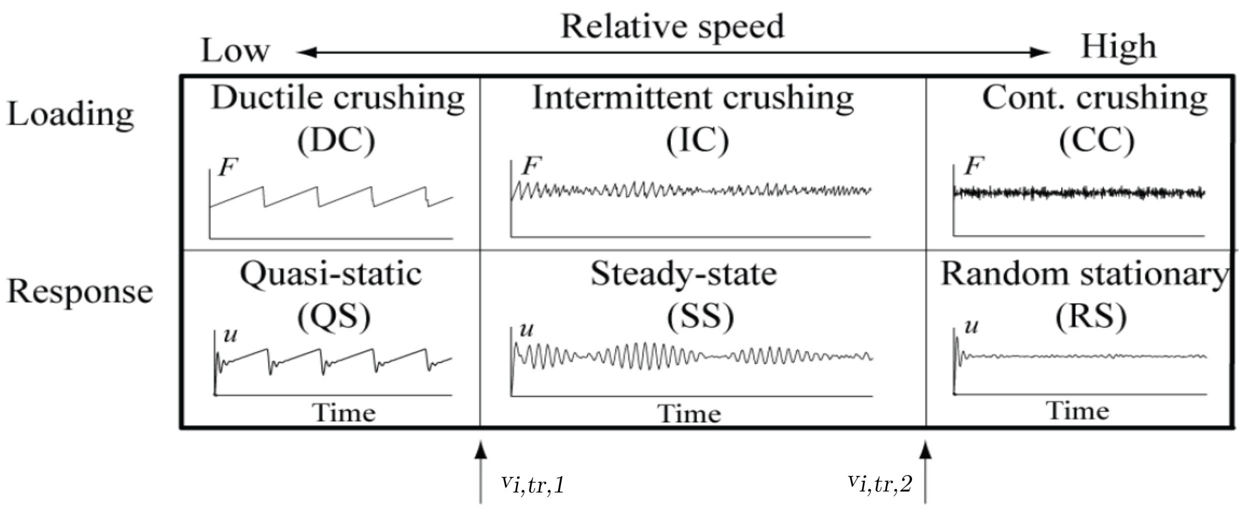


Figure 2.2: Different response regimes given relative velocity between structure and ice sheet. Figure taken from Bjerkås et al. (2013).

The forces are not magnified by the dynamics of the structure for this kind of response. The ice is often seen to fail in a ductile manner. Hence, it is called ductile crushing (DC).

For increasing ice sheet velocity, the quasi-static response may transition into steady-state vibrations. The ice sheet is here seen to fail simultaneously around the contact zone between ice and structure, and the ice failure mode is in between ductile and brittle. This kind of ice failure is often referred to as intermittent crushing (IC), as mentioned in Tab. 2.1. Steady-state vibrations and associated forces are magnified by the dynamics of the structure, i.e. the vibrations are of a resonant nature. The response fluctuates with almost constant amplitude, and the time signals of the response are often smooth and nearly sinusoidal (Kärnä, 1994). This mode of response is in some literature referred to as frequency locked-in crushing (FLC). When the frequency of the ice loading is close to the structural frequency, the frequencies tend to lock on to each other, giving stable resonance conditions. This lock-in phenomenon is analogous to the lock-in associated with vortex-induced vibrations (VIV) for a fluid flowing around a cylinder. Further information on vortex induced vibrations can be found in articles by e.g. Sarpkaya and Isaacson (1981).

At higher velocities, the steady-state response is lost, and the ice sheet fails non-simultaneous and apparently random around the contact zone. The structure is now displaced a certain distance, while the ice crushes continuously against it. The structure is oscillating frequent with small amplitudes around the displaced position. This response is often referred to as random stationary vibrations for continuous ice crushing (CC). The ice seems to fail in a brittle manner.

In addition to the different response regime, the load magnitude is naturally of concern. The ice load is closely related to the ice compressive strength for an ice sheet acting on a vertical structure. Some of the parameters determining the ice load are the ice thickness, ice velocity, ice compressive strength, ice flexural strength, ice temperature under which conditions the ice is formed etc. A typical model of the ice-structure interaction is shown in Fig. 2.3.

Table 2.1: Crushing regimes and structural response.

Crushing regime	**Structural response**
Ductile crushing including creep	Saw-tooth like response
Intermittent crushing	Steady-state response
Continuous crushing	Random stationary response

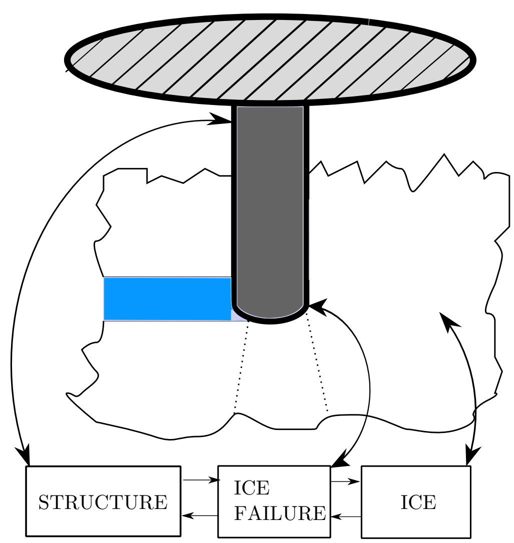


Figure 2.3: Ice-structure interaction. Figure based on Määttänen (2001).

2.2 Established methodology

The classification society and papers published on international conferences offer guidelines for engineering in Arctic conditions. Currently there is no widely accepted design code for ice pressure forces against structures, but there are many considerations given. Some of the standards and organizations providing guidelines regarding this topic are listed below:

- International Organization for Standardization (ISO).
- Det Norske Veritas (DNV).
- American Petroleum Institute (API).
- Canadian Standards Association (CSA).

The International standard, Petroleum and natural gas industries – Arctic offshore structures(ISO19906, 2010) is used in this thesis. Concerning structural fatigue and reinforced concrete, the NS-EN 1992-1-1:2004+NA:2008, referred to as EC2, is used.

2.3 Previous work on Norströmsgrund lighthouse

The Norströmsgrund lighthouse (Fig. 2.5) has a fairly cylindrical shape, and is resting on an underwater caisson, thus being a gravity-based structure. The lighthouse itself is reinforced with concrete and the underwater caisson is filled with rocks and sand. The seabed on location consists of dense moraine masses. Geometry of the lighthouse is shown in Fig. 2.6. Data have been collected from the Norströmsgrund lighthouse in the 1970s and 1980s (Engelbrektson, 1977, 1983). Newer measurements have been collected during the LOLEIF and STRICE campaigns in the time period 1999-2003. In addition to wind speed and direction, ambient temperature, these two campaigns had the possibility to measure ice pressure forces, structural tilt, acceleration and ice sheet thickness.

Ice thickness at Norströmsgrund lighthouse was measured by an EM laser scanner device. This particular measurement device was developed by the Alfred-Wegener-Institute for Marine and Polar Research (AWI) in Bremerhaven, Germany. Displacement and acceleration of the lighthouse were measured by two tilt-meters and two accelerometers. The top panels in Fig. 2.6 were recording ice pressure forces.

Figure 2.4: Location of Norströmsgrund lighthouse in the GoB.

Figure 2.5: Norströmsgrund lighthouse. Photo courtesy of LTU.

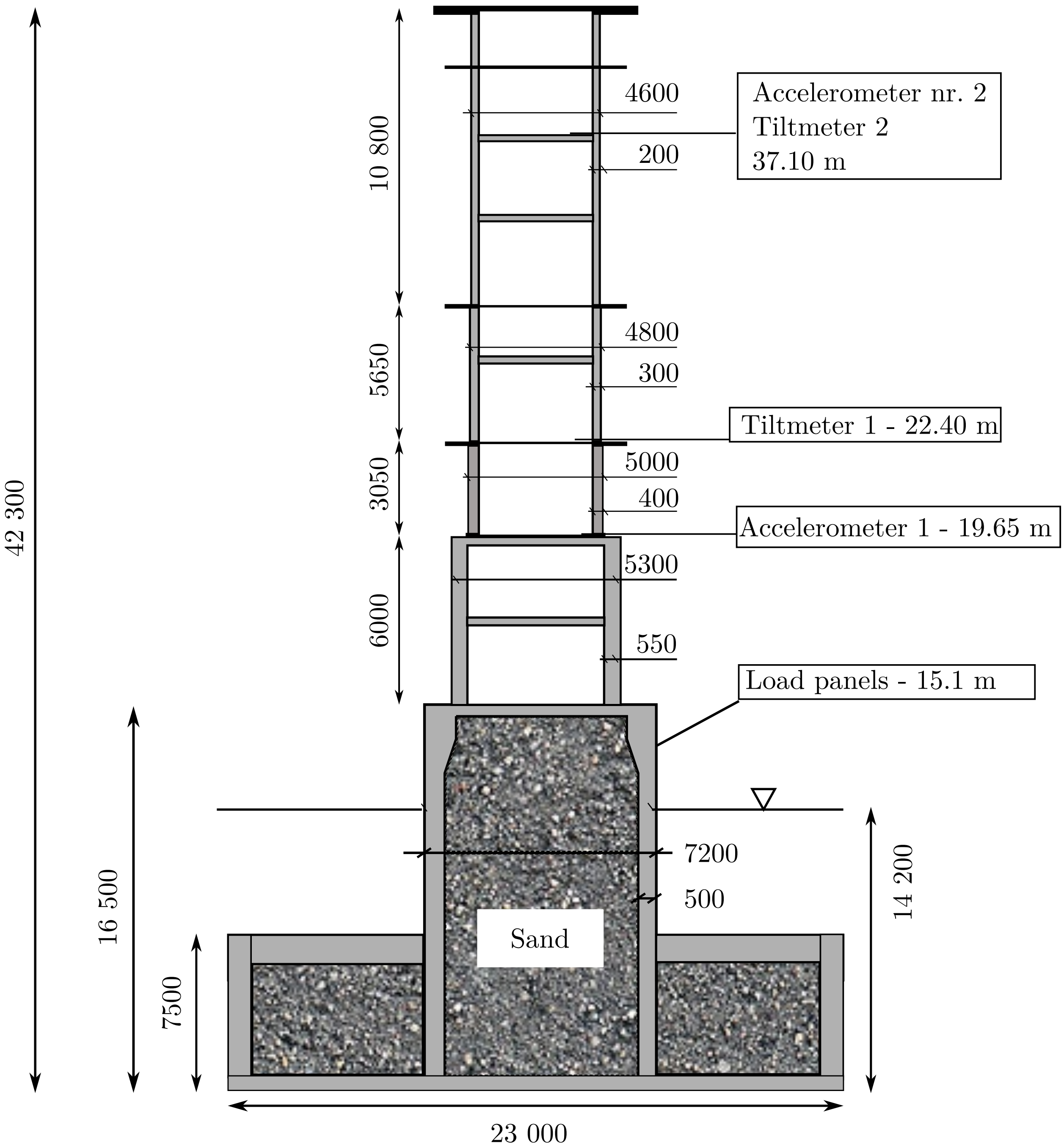


Figure 2.6: Geometry and instrumentation of Norströmsgrund lighthouse, dimensions in mm. unless otherwise specified. Figure after Bjerkås et al. (2013).

3 Method

A model which can estimate the structural fatigue of the lighthouse is to be established. The model utilizes the measured environmental data to estimate the forces acting on the structure. By using different criteria (section 3.2), the model calculates if the environmental conditions are giving SS vibrations or not. When a total accumulated number of cycles is found, the structural fatigue damage is estimated. This is done by summation of the damage dealt by the different events with SS vibrations (section 3.4). The following groups of input parameters are then needed:

- Ice properties based on measured environmental data.
- Structural properties of the lighthouse.
- Onset criteria for frequency locked-in crushing.
- Duration of a typical event with frequency locked-in crushing.
- Estimation of the total number of cycles.
- Stress range magnitude for the different sets of cycles.

3.1 Properties of ice

Calculation of ice actions on a structure is usually determined by (and closely related to) the forces required to fail an ice sheet in contact with a structure. Ice is a special material since it is often encountered in a temperature range which is very close to its melting temperature. Therefore, ice can creep with very little stress applied, or fracture under a very high strain rate. The grain structure of the ice is determined by which conditions it is formed under. Common ice structures are frazil ice, columnar ice, discontinuous ice and granular ice. Note that all the characteristics given in the sub-sections in this chapter are for first-year ice. Most of these qualities are taken from Timco and Weeks (2010), which also give properties for multi-year ice. Environmental parameters such as the wind velocity and wind direction are also needed to find the drift of the ice sheet. The main parameters related to the ice in this thesis are:

- Ice thickness calculated based on ambient air temperature.
- Ice drift velocity calculated based on wind velocity.
- Ice drift direction calculated based on wind direction.
- Ice compressive strength.

3.1.1 Ice thickness

As this is a study on ice forces, the ice thickness is naturally one of the most important parameters. It can either be determined by measured field data or calculations based on the ambient air temperature. Ice thickness is estimated by using Freezing Degree Days (FDD), which is a measure of how cold it has been for how long. When FFD is known, an empirical equation presented by Zubov (1943) can be used to estimate the ice thickness at time t. FDD is calculated as following:

$$FDD = \int_0^t (T_m - T_{air})\,\mathrm{d}t \tag{3.1}$$

where:

- T_m is the melting temperature ($°C$).
- T_{air} is the ambient air temperature in ($°C$).

The empirical relation between the ice thickness and FDD is given as:

$$h_{ice}^2 - 50 \cdot h_{ice} = FDD \tag{3.2}$$

where:

- h_{ice} is the ice thickness (cm).
- FDD is as defined above in Eq. 3.1.

3.1.2 Ice drift direction and velocity

Concerning the moving ice sheet, two parameters are used in the model; the drift velocity and direction. This section describes how these parameters are found. There are different classifications of moving ice. Three distinct cases are defined by Zubov (1943):

- Case 1 : The wind drift of a closed ice sheet.
- Case 2 : The wind drift of an isolated floe.
- Case 3 : The wind drift of scattered ice.

Factors that can influence the drift direction of the sea ice are for example the geostrophic conditions, the contour of the coast line, the bottom relief, the system of steady currents etc. Case 1 and 2 correspond well with the global ice situation in the north of the GoB, although scattered ice is also seen. When the ice is drifting due to the wind-induced shear stress on the ice cover, the ice cover will in turn cause a current in the water masses beneath the ice. This current follows the same laws of vertical distribution of velocity and direction as the current caused by the direct action of wind. In this thesis, the drift of the ice cover itself is of interest, not the current directions of the water far below it.

The Norwegian explorer Nansen (1902) made the observation during his expedition in 1893 to 1896 that sea currents and ice drift directions were not corresponding to the wind drift directions. Nansen let his vessel "Fram" get trapped by sea ice, and then he processed a total of 76 drift segments, finding drift angles and factors between wind and ice velocity. Zubov (1943) gives averages of these segments:

- Ice drift: 0.0555 m/s (2.98 miles/day).
- Ice drift due to 1 m/s wind speed: 0.0158 m/s (0.85 miles/day).
- Drift angle clockwise to wind direction: 28 °.

Based on this, Zubov approximated the wind factor, C_{iw}, to be 0.02. The wind factor is the ratio between the ice velocity and the wind velocity:

$$C_{iw} = \frac{V_{ice}}{V_{10}} \, (-) \tag{3.3}$$

where:

- V_{ice} is the ice velocity (m/s).
- V_{10} is the wind velocity 10 m above the surface (m/s).

Data collected by the drift of the vessel "Sedov" from September 1938 to January 1940, summarized in Zubov (1943, Tab. 100), shows that the average wind factor ranges between 0.013 and 0.020. The drift angle averages between wind and ice were seen to be from 13 ° to 42 °.

The Swedish oceanographer Ekman (1902), used the Nansens observations when he developed his well-known Ekman Spiral, see Fig. 3.1. Using the assumptions of active Coriolis forces, wind-induced shear stress on the surface, constant fluid particle velocity and viscous fluid, he found that the total mass transport was 90 ° clockwise to the wind surface velocity (on the Northern hemisphere).

Another result from the Ekman Spiral is that the fluid direction close to the surface is 45 ° clockwise to the wind direction on the Northern Hemisphere. However, if the water depth is less than 100 m, which is the case for Norströmsgrund lighthouse, the influence of the sea bottom becomes significant. The coastline will also affect the drift direction together with surface drift induced by wind generated waves.

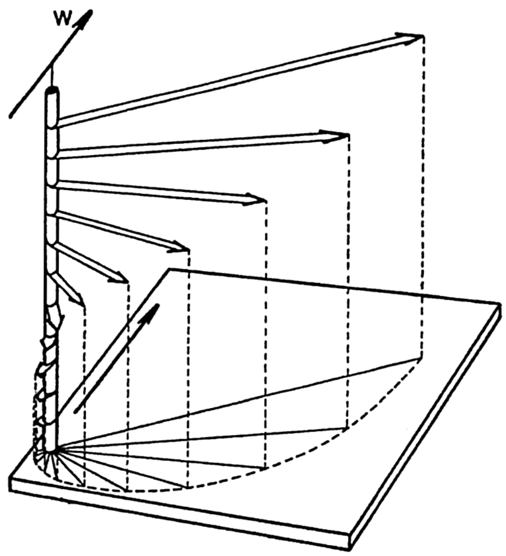


Figure 3.1: Illustration of the Ekman Spiral.

Including these effects, the observed angle is about 30 ° clockwise to the wind direction, see Fig. 3.2. This is close to the observations made by Nansen. Note that for Fig. 3.2 the wave current is wave-induced current (or mass transport) in second-order Stokes waves. Summarized, the ice drift direction is set to:

$$\theta_{ice} = \theta_{wind} + 30^{\circ} \tag{3.4}$$

where θ_{ice} is the ice drift direction and θ_{wind} is the wind drift direction, both relative to the north.

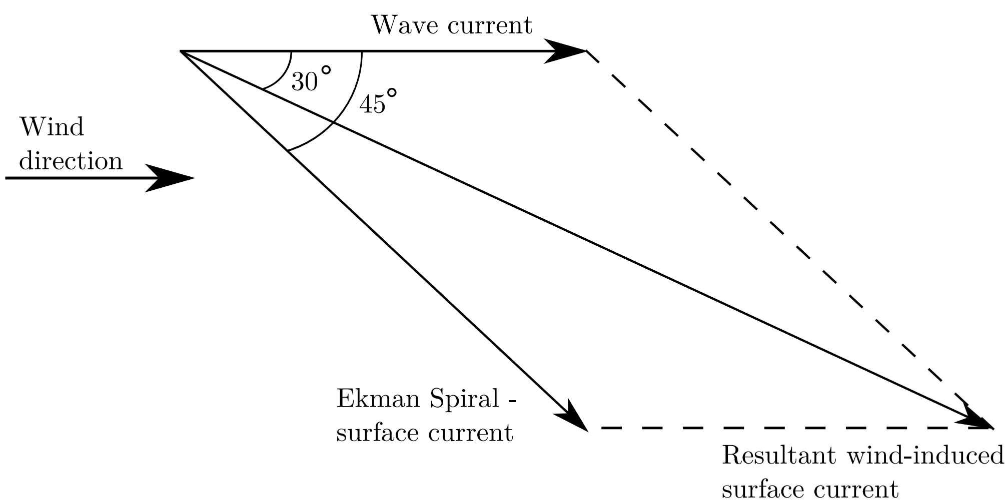


Figure 3.2: Possible explanation for surface current not being 45 degrees to the right of the wind. Figure after Williams et al. (1968).

3.1.3 Ice compressive strength

The compressive strength is an important factor when it comes to calculating the ice force. Note that this is when the ice fails by crushing. To find the peak force, one approach is to assume that the ice fails at a pressure level close to its uni-axial compressive strength. Two common methods two find the compressive strength are based on:

- Equations given by standardization codes.
- Use of stress versus strain rate curves.

The global ice pressure defined by ISO19906 (2010), is derived based on data from full-scale measurements from Cook Inlet, the Beaufort Sea, Baltic Sea and the Bohai Sea. Eq. 3.5 gives an upper bound for the global ice pressure. Both first-year and multi-year ice data were used when deriving this equation.

$$p_G = C_R \cdot (\frac{h_{ice}}{h_1})^n \cdot (\frac{D}{h_{ice}})^m \ (MPa) \tag{3.5}$$

where:

- m is equal to -0.16 (-).
- n is equal to -0.50 (-) + $h_{ice}/5$ for $h_{ice} < 1.0$ m and -0.30 (-) for $h_{ice} > 1.0$ m.
- h_1 is a reference thickness of 1.0 m.
- C_R is the strength parameter, set to 1.8 MPa for subarctic regions.

The coefficients n and m are both empirical and based on how ice thickness and structural width affect the global ice pressure. Eq. 3.5 is valid for aspect ratios D/h_{ice} greater than 2.0. This pressure is used together with nominal contact area to find the global ice force (Eq. 3.8). This method to find ice global force is chosen to be used further, as the method is more widely accepted than using stress/strain rate curves.

One example of a stress/strain rate curve will be presented below. Note that this method will not be used in the thesis, it is only included as additional theory on the subject of ice mechanics. A 4^{th} degree polynomial to describe the crushing strength given the stress rate is given by Määttänen (2001):

$$\sigma_c = (2.00 + 7.80\dot{\sigma_c} - 18.57\dot{\sigma_c}^2 + 13.00\dot{\sigma_c}^3 - 2.91\dot{\sigma_c}^4) \cdot \frac{A_0}{A} \ (MPa) \tag{3.6}$$

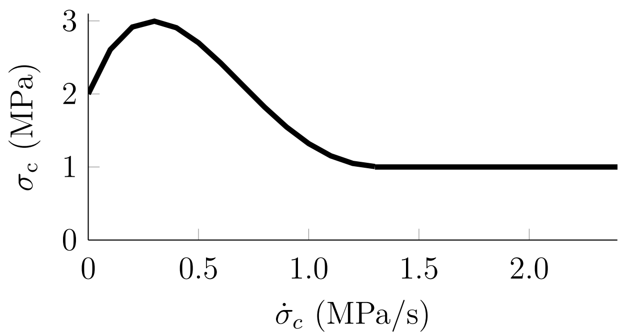


Figure 3.3: Stress/strain rate curve by Määttänen (2001)

.

where:

- σ_c is the compressive strength (MPa).
- $\dot{\sigma_c}$ is the stress rate (MPa/s).
- A_0 is a reference contact area, usually 1.0 m^2.
- A_n is the nominal contact area between ice and structure (m).

This curve is based on measurement data and the stress rate defined as:

$$\dot{\sigma_c} = (V_{ice} - \dot{u})\frac{8 \cdot \sigma_0}{\pi \cdot D}\ (MPa/s) \tag{3.7}$$

where:

- V_{ice} is the ice velocity (m/s).
- $\dot{u}$ is the structural displacement velocity (m/s).
- D is the structural diameter at the waterline (m).
- σ_0 is a reference value for the crushing strength (MPa).

Eq. 3.7 is based on an equation by Blenkarn (1970). Eq. 3.6 is valid for a stress rate up to approximately 1.3 MPa/s. Above 1.3 MPa/s the compressive strength value is set to 1.0 MPa. The curve is shown in Fig. 3.3 below.

3.1.4 Ice force

The force acting on the structure consists of two contributions; one static, F_C, and one dynamic part, F_D. To determine the fatigue life of the structure, the stress range, $\Delta\sigma$, is required in addition to number of cycles. The stress range is depending on the load applied.

This will be further described in section 3.4.2. The general equation for finding the static force is:

$$F_G = p_G \cdot A_N \tag{3.8}$$

where:

- F_G is the global force (MN).
- p_G is the ice pressure averaged over the nominal contact area (MPa) (Eq. 3.5).
- A_N is the nominal contact area (m^2).

In this thesis, events with frequency lock-in are examined. Therefore the dynamic load component is been assumed to be the same as described in ISO19906 (2010, section A.8.2.6.1.5). During an event with frequency lock-in, the load history will be represented by a sawtooth function. Here the frequency of the forcing function is $f = 1/T$. The period of the sawtooth function, T, is shown in Fig. 3.4. This frequency is assumed to be the same frequency as the lowest natural frequency. ISO19906 states that f must be below 10 Hz. Force peak value, F_{max}, and double amplitude, $\Delta F = F_{max} - F_{min}$, are assumed to be constant. The force peak value is assumed to be the same as the maximum global ice pressure, F_G, defined in Eq. 3.8. The double amplitude, which depends on the natural frequency and the ice velocity, is expressed as a fraction of the peak force:

$$\Delta F = q \cdot F_{max} \tag{3.9}$$

The coefficient q is in the range of 0.1 to 0.5. This factor should be scaled so that the response velocity at waterline is 1.4 times the highest ice velocity for which FLC can occur, $v_{i,tr,2}$. $v_{i,tr,2}$ is obtained as:

$$v_{i,tr,2} = \gamma_v \cdot f_n \tag{3.10}$$

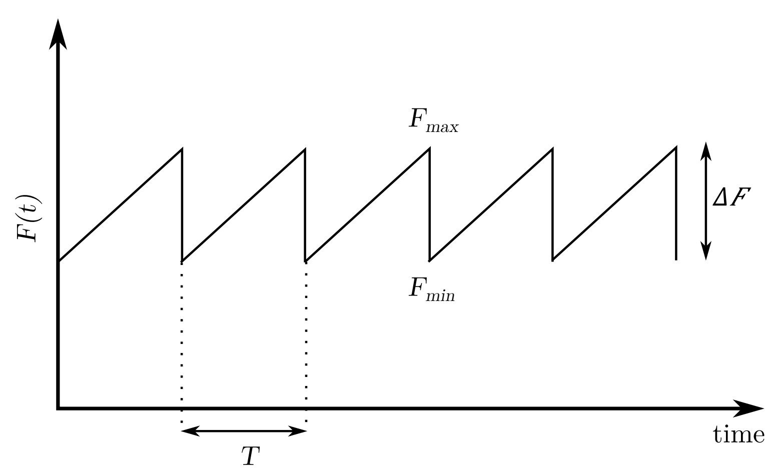


Figure 3.4: Assumed sawtooth function for FLC conditions.

Where $\gamma_v = 0.060$ m. This is based on field data for slender structures with natural frequencies up to 5 Hz. The forcing function used in this thesis is expressed as:

$$F = F_{max} + \Delta F(\frac{sawtooth(2 \cdot \pi \cdot f_n \cdot t) - 1}{2}) \tag{3.11}$$

The *sawtooth*-function is an inbuilt function in MATLAB.

3.2 Onset criteria for frequency locked-in crushing

To find an estimate for the number of cycles due to frequency locked-in crushing, criteria for FLC conditions are needed. Based on observations, general parameters determining if we have FLC are:

- Ice failure mode (section 3.2.1).
- Ice drift direction and velocity (section 3.2.2).
- Ice temperature profile (section 3.2.3).
- Static waterline displacement (section 3.2.4).

After defining these general criteria, different methods are used to calculate the onset criteria for FLC:

- Dynamic instability criterion by ISO19906 (2010) (section 3.2.5).
- Dimensionless groups using structural diameter, ice thickness and ice velocity by Palmer et al. (2009) (section 3.2.6).
- Temperature and thickness criteria Bjerkås and Skiple (2005) (section 3.2.7).

3.2.1 Criterion for ice failure mode

Kärnä and Jochmann (2003) report that the first-year ice at Norströmsgrund fails by splitting, bending, buckling or by crushing. Small and uniform sheets usually fail by splitting. Larger floes which are laterally confined are seen to fail mostly by bending, crushing or a mixed mode of the two. Crushing is the observed failure mode during steady-state vibrations. Thus, it is of interest to determine if the failure mode is crushing.

The aspect ratio between the waterline diameter, D, and the ice thickness, h_{ice}, is an useful indicator on this. Kärnä and Jochmann report that when the ice thickness becomes larger than approximately 0.2 m, the occurrence probability for crushing is larger. For Norströmsgrund lighthouse the ratio between waterline diameter and ice thickness will then determine the failure mode. The ratio limit, which is based on Kärnä and Jochmann (2003, Fig. 8), is:

$$\frac{D}{h_{ice}} = \frac{7.5\,m}{0.2\,m} \approx 40 \tag{3.12}$$

This value will be the boundary between bending and crushing failure. For ratios larger than 40, the failure will be by bending, while for ratios lower than 40 failure will be by crushing in the model presented in this thesis. Frequency locked-in crushing is, as the name suggests, most likely to occur when the failure is by crushing. Contrary, the probability for having FLC is low when the failure is by bending. Thus, the failure mode onset criterion for FLC is:

$$\frac{D}{h_{ice}} < 40 \tag{3.13}$$

3.2.2 Criterion for ice drift direction

The ice drift must be compatible for having ice drift against the structure. No ice drift against the structure will naturally give no structural vibrations. Having defined the ice drift direction relative to the wind direction (Eq. 3.4), boundaries for the ice drift must be set. That is, for which directions will the ice drift, and not be stopped due to physical limitations. One physical limitation for ice drift, can for example be a coastline. From Fig. 3.5 it is seen that the ice must drift toward the south, if the global ice sheet shall be able to move. Note that Fig. 3.5 is a model and that the drift limits are not necessarily the same as for the full-scale model. The drift limits for this model are assumed to be:

$$300\,^{\circ} < \theta_{ice} < 45\,^{\circ} \tag{3.14}$$

If the ice drift is above $45\,^{\circ}$ and mostly toward west, the western coastline of the gulf will prevent the ice from drifting. If the drift is below $300\,^{\circ}$ and toward east or north-east, the ice sheet will be stopped by the coastline on the eastern side of the gulf. This is supported by satellite images of the GoB (Bjerkås et al., 2013, Fig. 6). Fig. 3.6 shows the track of drifting ice.

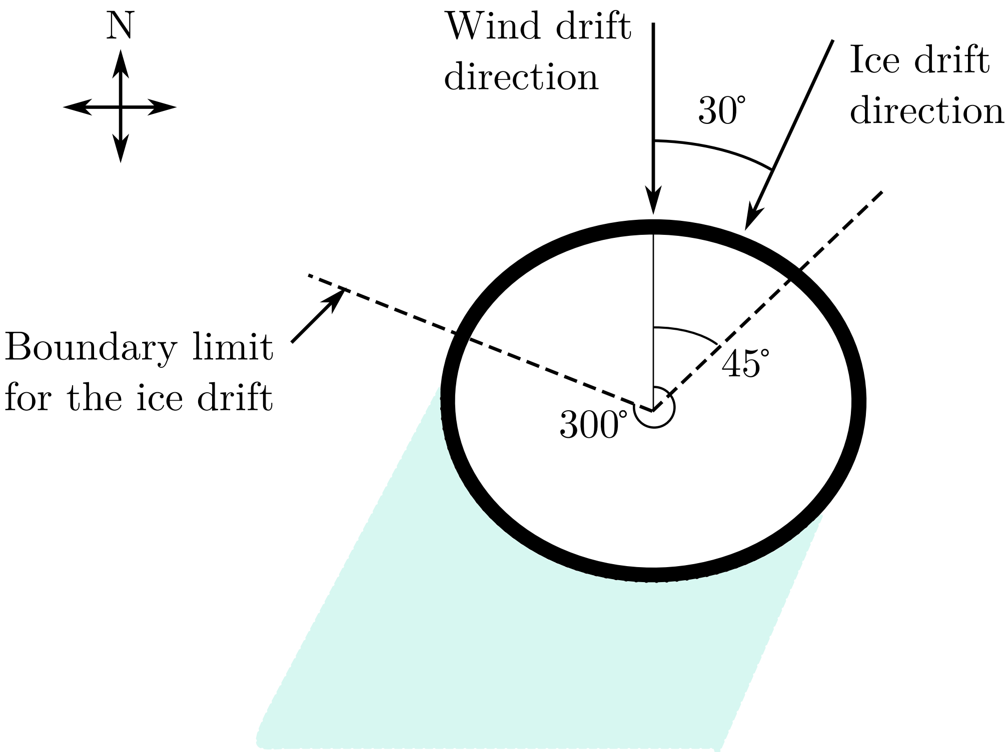


Figure 3.5: Schematic set-up for ice drift against the lighthouse.

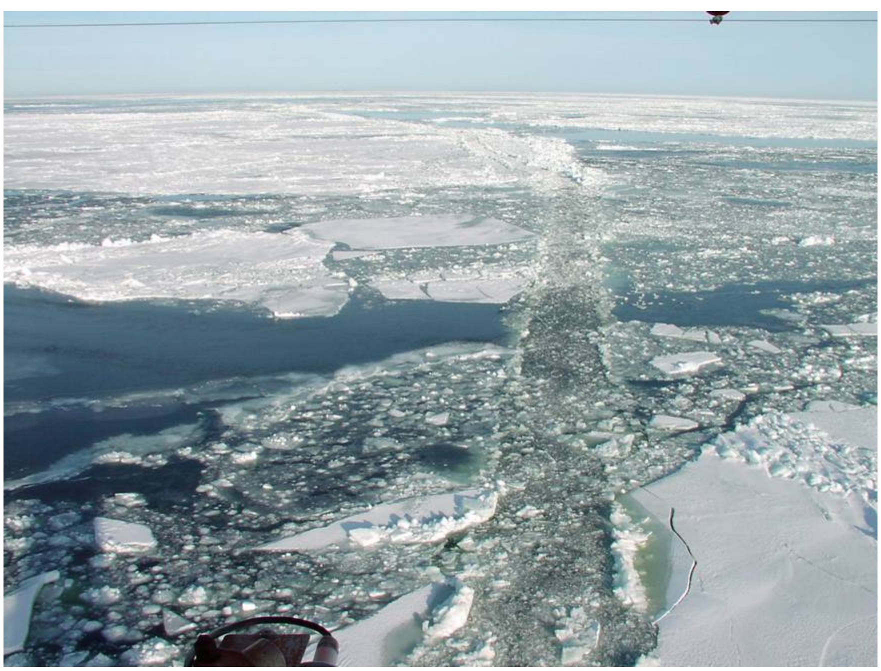


Figure 3.6: Track of the ice sheet. Photo taken the 21st February 2003. Photo courtesy of Kari Kolari.

3.2.3 Criterion for ice temperature

Ice is as mentioned a material which exists close to its natural melting point. Bjerkås and Skiple (2005) argued that the failure mode of the ice is closely related to the ice temperature. Most solids are seen to have increasing brittleness for decreasing temperature, as is the case for ice. Colder ice is seen to fail in a brittle manner, while warmer ice will be more ductile. With a more ductile ice sheet, the probability for FLC is higher. The model will use the ice growth when investigating the ice temperature profile.

Bjerkås et al. (2013) indicated that while the ice grows, the temperature profile is linear over the thickness. With little or no ice growth, Bjerkås et al. observed different temperature profiles, both irregular and C-shaped profiles. Different profiles are illustrated in Fig. 3.7, which is based on Bjerkås et al. (2013, Fig. 7). Note that the use of Fig. 3.7 in this thesis is for illustration only.

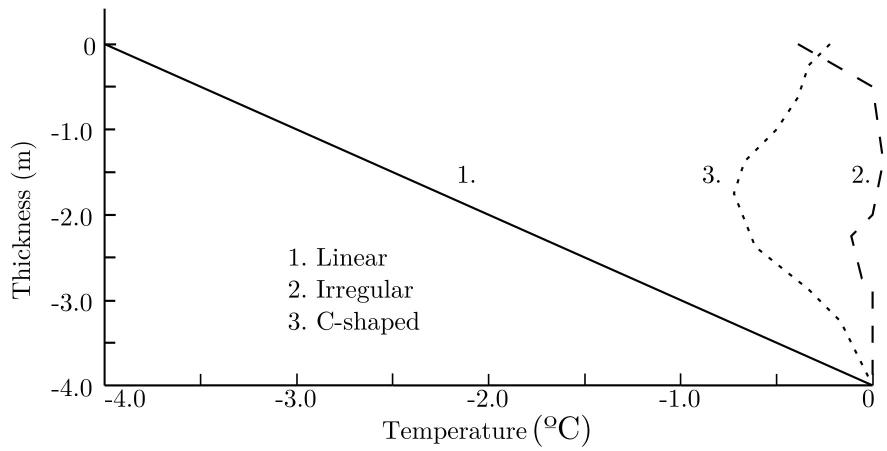


Figure 3.7: Different ice temperature profiles. Figure based on Bjerkås et al. (2013).

The temperature profile and ice core temperature is important when it comes to the dynamic response mode of a structure subjected to level ice. Bjerkås et al. (2013) observed that for warmer and more ductile ice, i.e. when the ice growth stops and temperature profile becomes C-shaped, FLC and resonant-like vibrations are more likely to occur. Having a growth which is less than 1 cm per week is assumed to be little or no growth in this thesis. This is checked by using Eqs. 3.1 and 3.2.

3.2.4 Criterion for waterline displacement

In order for FLC to initiate, the lighthouse must be displaced a certain distance before the vibrations are able to start. The relationship between the static load and the waterline displacement is:

$$F_C = k_{wl} \cdot \delta_{wl} \tag{3.15}$$

where:

- F_C is the static load (N).
- k_{wl} is the waterline stiffness of the structure (N/m).
- δ_{wl} is the waterline displacement of the structure (m).

F_C is found by using Eqs. 3.5 and 3.8. From these equations we see that the static force is depending on the ice thickness. A value for the waterline stiffness of the lighthouse is taken from Kärnä et al. (2004). They found the waterline stiffness by applying a load of 1 MN and reading the horizontal displacement. The stiffness was found to be:

$$k_{wl} = \frac{1 \cdot 10^6 \, N}{581.766 \cdot 10^{-6} \, m} = 1.72 \cdot 10^9 \; (N/m) \tag{3.16}$$

It is then assumed that the waterline displacement must be at least 1 mm before FLC can initiate. This is based on observations by Bjerkås et al. (2013). Then the waterline displacement onset criterion for FLC is:

$$\delta_{wl} = \frac{F_C}{k_{wl}} > 1.0 \; (mm) \tag{3.17}$$

3.2.5 Criterion for dynamic stability

The susceptibility to dynamic instability for the lighthouse is investigated. Looking into the ISO19906 (2010), a criterion for dynamic stability is found to be:

$$\xi_n \geq \frac{\phi_{nC}^2}{4\pi \cdot f_n \cdot M_n} \cdot h_{ice} \cdot \theta \tag{3.18}$$

where:

- ξ_n is the total damping of the eigenmode as a fraction of critical (-).
- ϕ_{nC} is the normalized modal amplitude at the ice action point (-).
- M_n is the true modal mass (kg).
- f_n is the natural frequency of the eigenmode (Hz).
- h_{ice} is the ice thickness (m).
- θ is a coefficient equal to $40 \cdot 10^6$ (kg/(ms)).

To ensure dynamic stability of a natural mode n, the structural damping must be larger than the opposite contribution to dynamic instability provided by the ice action. The procedure is then to use modal values and check if the structure is susceptible to dynamic instability or not. The dynamic load associated with this instability was described in section 3.1.4.

3.2.6 Criteria based on dimensionless groups

There have been many attempts to define boundaries between the different crushing modes and response regimes shown in Fig. 2.2. Palmer et al. (2009) suggest that the boundaries are related to ice velocity, ice thickness, contact width, natural frequency and using dimensionless groups, closely related to the theory of vortex-induced vibrations (VIV). The dimensionless groups suggested by Palmer et al. are:

$$\frac{V_{ice}}{f_1 \cdot h_{ice}} \tag{3.19}$$

and

$$\frac{V_{ice}}{f_1 \cdot D} \tag{3.20}$$

where:

- V_{ice} is the ice drift velocity (m/s).
- f_1 is the lowest natural frequency for the waterline translation (1/s).
- h_{ice} is the ice sheet thickness (m).
- D is the structural waterline diameter (m).

In addition to these equations, two graphs containing field data on different structures taken from Palmer et al. (2009, Figs. 3 and 4) are used. For Norströmsgrund lighthouse, the values are based on data from Bjerkås (2006). Eqs. 3.19 and 3.20 use the following lower and upper limits for FLC:

Dimensionless group	**Lower boundary**	**Upper boundary**
$V_{ice}/(f_1 \cdot h_{ice})$	0.0150	0.2500
$V_{ice}/(f_1 \cdot D)$	0.0100	0.0035

Table 3.1: Boundary values for dimensionless groups.

3.2.7 Criteria based on ice temperature, thickness and velocity

Bjerkås and Skiple (2005) point out that the temperature is an important factor. Observed data from Norströmsgrund shows that the occurrence probability of FLC increases with warmer ice temperatures. Based on measurements and corresponding analysis, transition velocities between CC and IC were proposed. These transition velocities depend on ice velocity, ice thickness and air temperature. The observed drift velocity for IC is between 0.02 and 0.08 m/s. Curves are then fitted to the upper values for observed IC, as shown in Figs. 3.8 and 3.9. Note that $v_{2,T}$ and $v_{2,h}$ in Figs. 3.8 and 3.9 are upper boundary limits for when IC can occur, similar to $v_{i,tr,2}$ in Fig. 2.2.

The upper boundary trend line for $v_{2,T}$ (Fig. 3.8) using air temperature and ice velocity is given as:

$$v_{2,T} = 0.004 \cdot T_{air} + 0.07 \tag{3.21}$$

where T_{air} is the ambient air temperature (°C). The temperature range for which the upper linear trend line $v_{2,T}$ is defined, is between -12 °C and 5 °C. There is supposedly a time delay between measured air temperature and current ice temperature. Averages over 24 hours of the air temperature were used during the collection of these data. Note the low number of IC observations below temperatures of -1 °C in Fig. 3.8.

The upper boundary trend line for the observed IC given a velocity and an ice thickness ($v_{2,h}$ in Fig. 3.9) is:

$$v_{2,h} = 0.03 \cdot h_{ice} + 0.045 \tag{3.22}$$

Where the ice thickness is between 0.2 and 1.2 m. Bjerkås and Skiple point out that due to lack of CC observations, the shaded areas in Figs. 3.8 and 3.9 should be studied further by collecting more data.

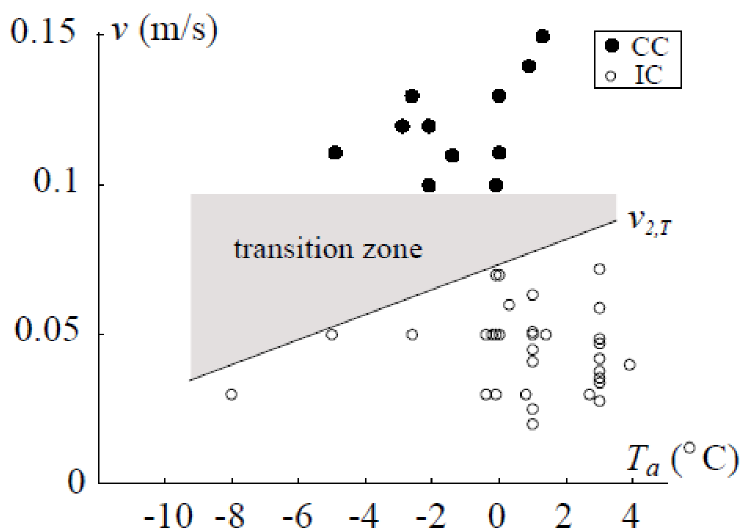


Figure 3.8: Distribution of CC and IC w.r.t. ice drift velocity vs. 24 hour average air temperature. Figure taken from Bjerkås and Skiple (2005).

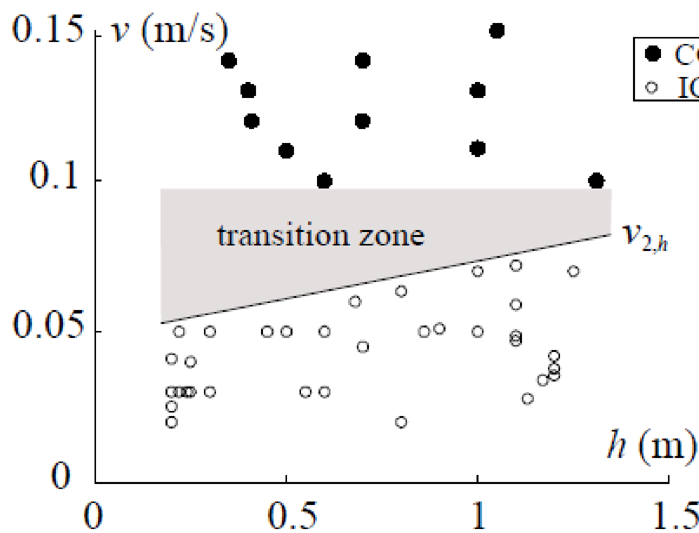


Figure 3.9: Distribution of CC and IC w.r.t. ice thickness versus ice velocity. Figure taken from Bjerkås and Skiple (2005).

3.3 Accumulated cycles

In order to calculate structural fatigue, the number of load cycles must be known. To find the number of cycles over a winter season, we need to know:

- How many events with FLC and SS vibrations there are.

- How long each FLC event lasts.

- The frequency which the structure oscillates with during these events.

- The ice sheet drift velocity.

On the lowest level, the mechanism that initiates the FLC is important. This is related to the ice failure mechanism. The ice velocity is important when it comes to determining the time, t, between each oscillation period due to FLC (Fig. 3.12). Assuming the structure oscillates with the lowest natural frequency, the number of cycles for a given event is estimated. Summing the number of cycles for each event in the season, the total number of cycles is found.

In the present thesis, it is assumed that the ice failure mechanism is a simple one, with a transverse global crack at a distance L_{crack} from the structure, shown in Fig. 3.10 below. When an ice sheet drifts against the structure, it will crush locally around the structure and create ice rubble. A transverse crack forms at the distance L_{crack} from the structure when the contact force is close to maximum level. After the global failure, the force level drops, the ice rubble clears out and the process will be repeated as long as the criteria for FLC are fulfilled. The distance to the transverse crack, L_{crack}, is a parameter which is widely discussed in the field of ice mechanics, and there is no established and accepted method to estimate it. Regarding ice crack length, L_{crack}, Määttänen (2001) states: "No physical reasoning exists to support characteristic ice failure length in crushing". Here, it is simply assumed that the L_{crack} in frequency locked-in crushing mode is two times the ice thickness, shown in Fig. 3.11. While the local ice fails by crushing, ice rubble accumulates on top of the ice sheet (or in some cases beneath it). When the pile becomes large enough, the weight of the rubble pile will cause the sheet to fail by global bending.

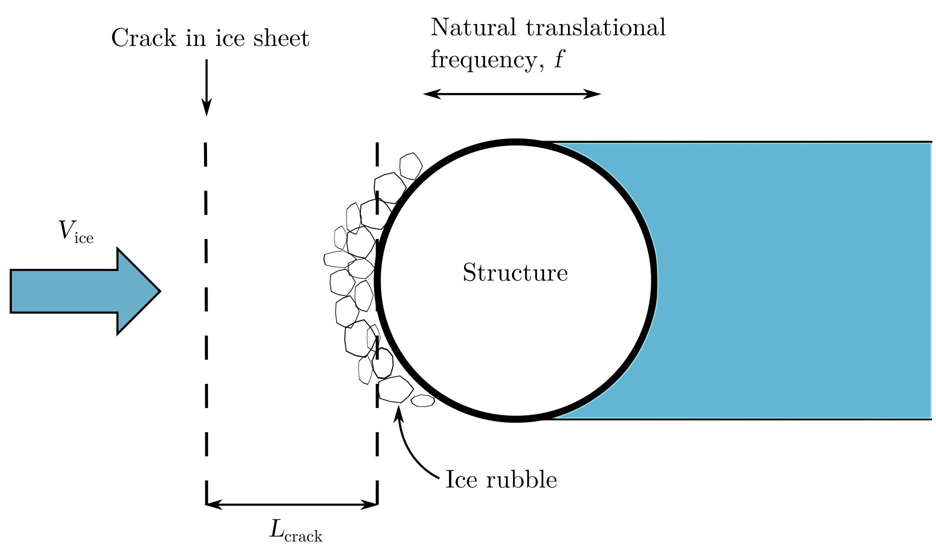


Figure 3.10: Illustration of simple failure mode, after Palmer et al. (2009).

The contact between the ice and the structure is lost, and the ice rubble cleared out. The time it takes for the contact to once again be established depends on both the length to the crack, L_{crack}, and the ice velocity, V_{ice}. The relationship is:

$$t = \frac{L_{crack}}{V_{ice}} = \frac{2 \cdot h_{ice}}{V_{ice}} \tag{3.23}$$

The time duration of each event the crushing frequency locks on to the natural frequency of the structure, is depending on many factors:

$$\theta_t = \theta_t(M_{ice}, M_{structure}, C, K, h_{ice}, V_{ice}) \tag{3.24}$$

where:

- θ_t is shown in Fig. 3.12.
- $M_{structure}$ is the mass of the structure.
- M_{ice} is the mass of the incoming ice sheet.
- C is the damping of the ice-structure system.
- K is the stiffness of the ice-structure system.

This is based on field observations by Bjerkås (2006). Bjerkås observed that frequency locked-in crushing lasted from 10 to 80 seconds for Norströmsgrund lighthouse. In this thesis the lock-in duration is set to 80 seconds, as the longest lock-in period will give a more conservative number of cycles. In other words, this is the time it takes for the rubble pile to become large enough to give global failure of the ice sheet.

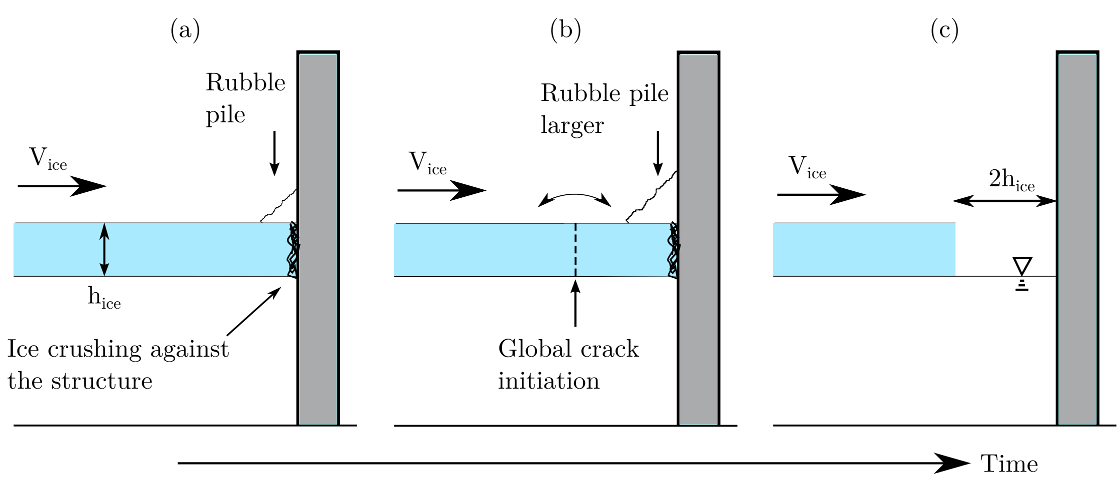


Figure 3.11: Bending failure against a vertical structure due to ice rubble.

The displacement for an event with FLC and SS vibrations is illustrated in Fig. 3.12. Note that the frequency of the cycles in Fig. 3.12 is not the same as for Norströmsgrund, nor is the waterline displacement, it is purely illustrative. Using frequency for mode number n, the period of one cycle is $1/f_n$. Having defined the time of FLC before the transverse crack forms as θ_t, the number of cycles per lock-in will be:

$$n_{cycles\,per\,lock-in} = \theta_t \cdot f_n \tag{3.25}$$

Knowing the lock-in time θ_t, and the time between lock-ins, t, the total number of lock-ins per event is found as:

$$n_{lock-ins\,per\,event} = \frac{T_{event}}{\theta_t + t} \tag{3.26}$$

Total number of cycles per event will then be:

$$n_{cycles\,per\,event} = n_{lock-ins\,per\,event} \cdot n_{cycles\,per\,lock-in} \tag{3.27}$$

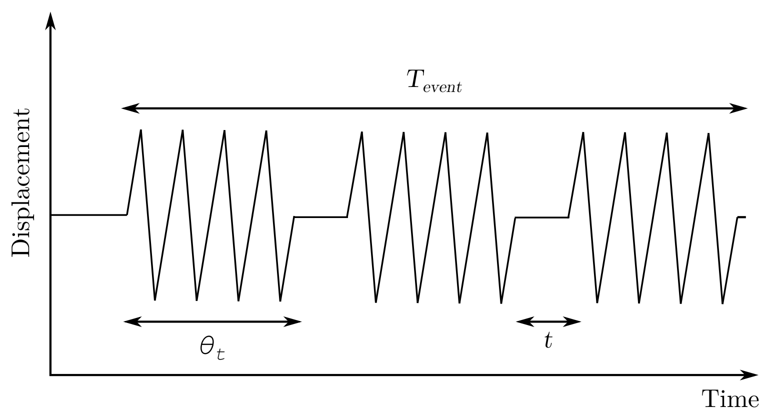


Figure 3.12: Illustrative model for a single event with SS vibrations.

3.4 Fatigue life

The theory in this section is based on Almar-Næss (1985). In the case of having cyclic loads acting on a structure, failure can occur even if the stress level is below yield strength. When the cumulative effect of many load cycles becomes large enough, the structure will fail due to fatigue, i.e. when the fatigue capacity has been reached. This is in some literature referred to as the fatigue life. Parameters related to the fatigue life are the load frequency, maximum loading, stress amplitude, material properties and the environment around the structure.

Failure by cyclic loads is defined by three categories; low-cycle, high-cycle and the fatigue limit. The categories are shown in Fig. 3.13, which is an SN-curve. The SN-curve shows the relation between the stress range, $\Delta\sigma$, and the number of cycles before failure, N, for the given stress range. The relation given in Eq. 3.28 is valid for the high-cycle range, further described below.

The low-cycle range is defined for cycles up to $\sim 10^5$ cycles. In this region the material undergoes cyclic plasticity, and the stress range in Eq. 3.28 is not applicable. This kind of failure can occur during extreme load scenarios, e.g. a tornado or an earthquake.

The high-cycle range is above $\sim 10^5$ cycles, and in this range the stress is essentially elastic. For most marine structures, like ships, offshore structures, risers etc., fatigue stresses are mainly in the high cycle range. This is assumed to be the case for the model of Norströmsgrund lighthouse. The SN-curve tends to follow a log-linear relationship in this range:

$$N(\Delta\sigma)^m = A \tag{3.28}$$

where:

- $\Delta\sigma$ is the stress amplitude (MPa).
- N is the number of cycles which gives failure for a given $\Delta\sigma$.
- m is a material dependent constant.
- A is a material dependent constant.

This equation becomes a straight line using log-log scale:

$$log\Delta\sigma = -\frac{1}{m} \cdot logN + \frac{1}{m} \cdot logA \tag{3.29}$$

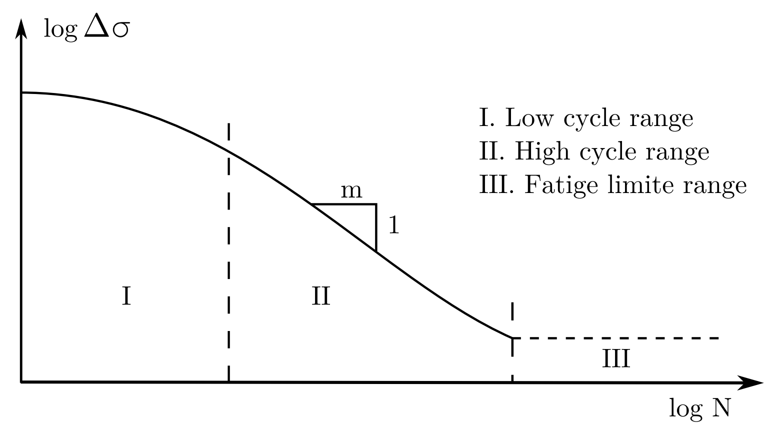


Figure 3.13: Illustration of a SN-curve with fatigue limit.

At very low stress ranges (range III in 3.13), some materials may have "infinite" life, i.e. a fatigue life significantly longer than the exposure to cyclic loads. The stress range is said to be below the fatigue limit, and the number of cycles to failure is significantly larger than any laboratory test can measure. This concept only applies for some materials in non-corrosive environments, e.g. steel in dry air. Another reason for steel having a fatigue limit, is that steel experiences strain-hardening (see Fig. 3.17). For large strains above yielding, the strength of the material increases.

The cyclic load is represented by the stress range, $\Delta\sigma$, which is shown in Fig. 3.14. Note that σ_m in Fig. 3.14 is the mean stress, and only tensile (positive) stress is considered. $\Delta\sigma$ will be the significant parameter when determining the fatigue life. $\Delta\sigma$ is defined as:

$$\Delta\sigma = \sigma_{max} - \sigma_{min} \tag{3.30}$$

where:

- $\Delta\sigma$ is the stress range (MPa).
- σ_{max} is the maximum stress (MPa).
- σ_{min} is the minimum stress (MPa).

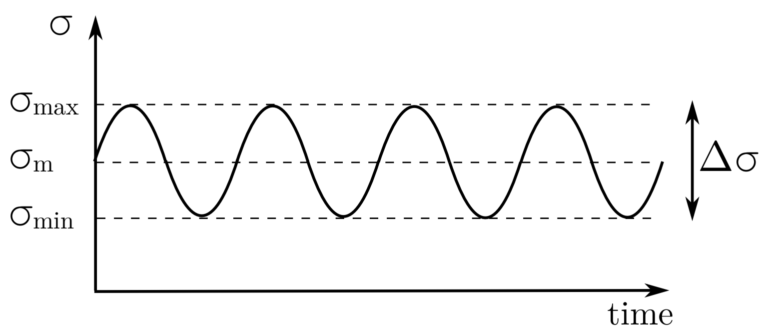


Figure 3.14: Fatigue load history. Figure based on Almar-Næss (1985, Fig. 1.10).

When the number of cycles for a given stress range is known, the cumulative fatigue damage can be estimated. The SN-curve is defined for a constant stress range and a constant mean stress. Since both the mean stress and the stress range are depending on the load conditions, which again depend on the environment surrounding the structure, the load history will not stay constant. To find the fatigue damage for an alternating load history, Miner-Palmgrens hypothesis is applied. This hypothesis gives the cumulative damage as the sum of the partial damage, and is often called the Miner-sum:

$$D = \sum_{i=1}^{k} \frac{n_i}{N_i} = \delta \tag{3.31}$$

where:

- n_i is the number of cycles.
- N_i is the number of cycles to failure at stress range $\Delta\sigma_i$
- k is the number. of partitions which the stress history is divided into.
- δ is a constant between 0 and 1.

Fig. 3.15 shows how the stress history is divided in a manner such that it can be used with Eq. 3.31. n_i is found from analysing the stress history, while N_i is found from SN-curves.

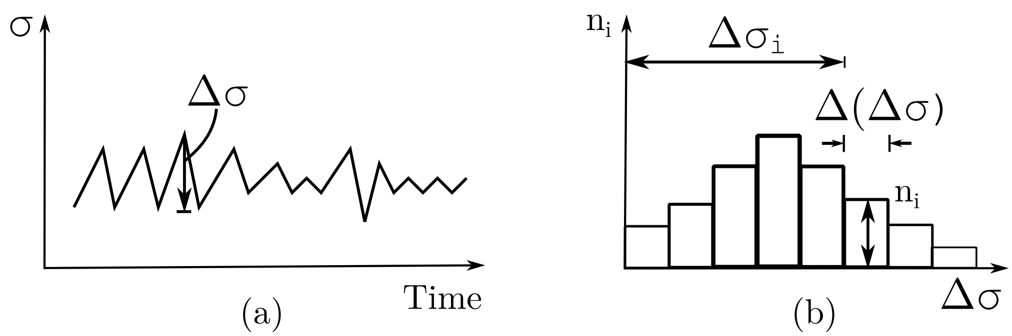


Figure 3.15: Fatigue load effects: a) Stress history. b) Frequency of stress ranges. Figure based on Almar-Næss (1985, Fig. 1.9).

3.4.1 Considered cross-section and material

Global failure due to fatigue is considered in this thesis. This is checked by investigation of fatigue in important parts of the structure. If the considered part fails, the structure will fail globally. The cross-section located 7.5 m above the sea bed (Fig. 3.16) is considered. The reinforcement is assumed to be ribbed bars. The type designation of this steel is **B500C**, while the concrete type is assumed to be **B35**. A typical stress-strain curve for ribbed steel is shown in Fig. 3.17a. Fig. 3.17b shows idealized and design stress-strain curve for ribbed bars. Ribbed bars are also referred to as rebars in this thesis.

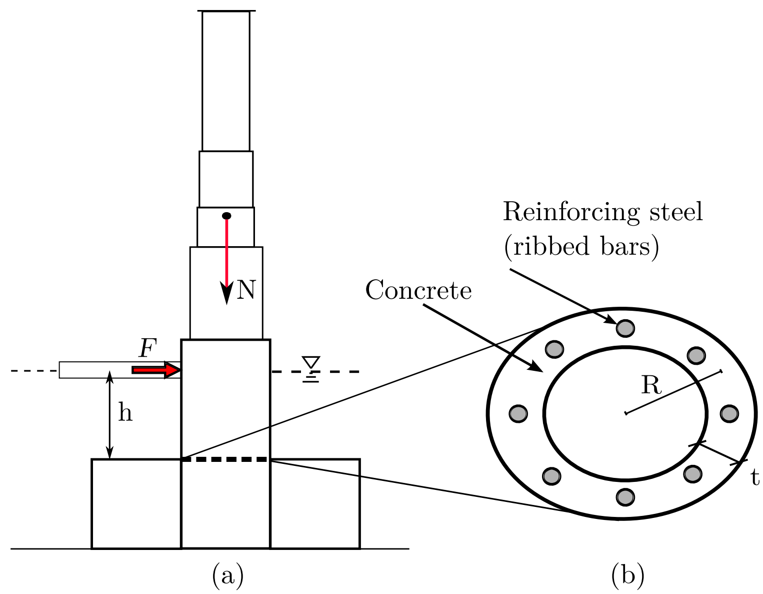


Figure 3.16: Considered cross-section with forces acting on it. Note that the dimensions are not to scale, nor is the number of ribbed bars.

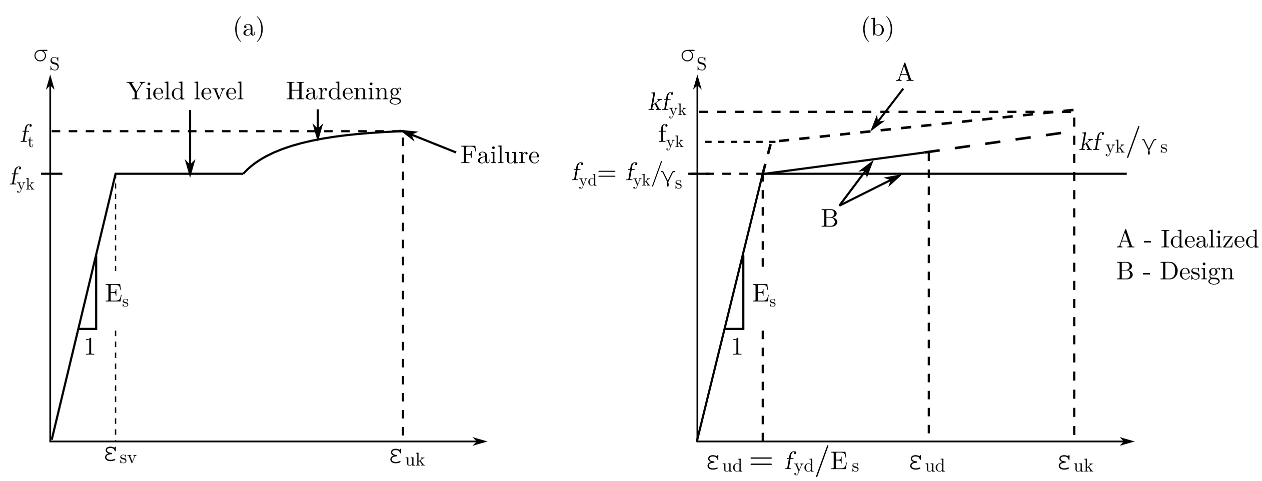


Figure 3.17: a) Typical stress-strain curve for ribbed steel. b) Design curve taken from EC2 (2008, Ch. 3.2.7).

For Fig. 3.17b the following is valid:

- f_{yk} is characteristic yield strength.
- E_s is modulus of elasticity, equal to $2 \cdot 10^5$ MPa for **B500C**.
- ϵ_{yd} design yield strain.
- ϵ_{uk} is tensile strain at rupture.
- ϵ_{ud} is design tensile strain at rupture (strain limit).

- γ_s is a material factor for reinforcing steel, here equal to 1.15.
- k = 1.04 (EC2, 2008, Ch. 3.2.7).

The characteristic SN-curve for the reinforcing steel is given by EC2 (2008), as shown in Fig. 3.18. Parameters related to the curve are given in Tab. 3.2. Unfortunately, not all

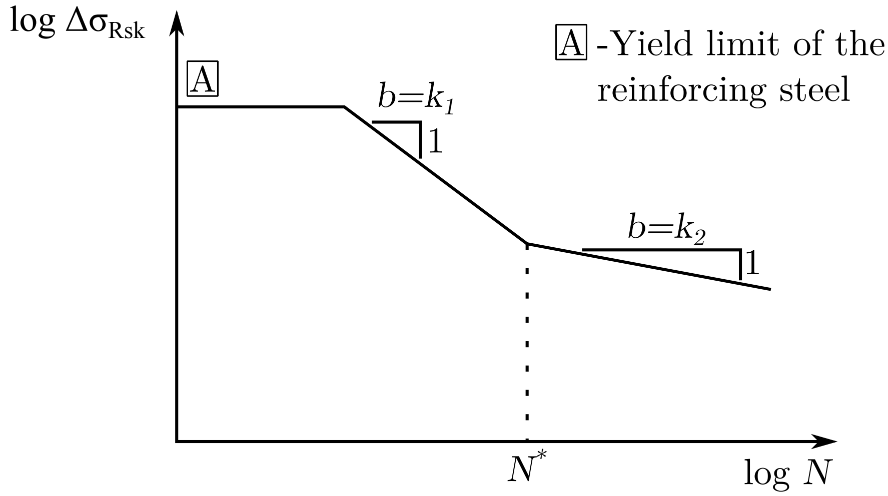


Figure 3.18: Characteristic SN-curve for rebars. Figure based on EC2 (2008, Fig. 6.30).

N^*	k_1	k_2	$\Delta\sigma_{Rsk}(N^*)$
10^6	5	9	162.5 (MPa)

Table 3.2: SN-curve parameters for ribbed steel. Values taken from EC2 (2008, Tab. 6.3N).

details concerning the cross-section are known. It is known that the material is concrete with ribbed bars reinforcing it, but the number of rebars is not known, nor is the rebar distribution. Therefore, the required reinforcement of the cross-section is estimated. This is based on using the design load, L_{design}, at the MWL, which is 2 MN/m for Norströmsgrund lighthouse (Bjerkås, 2013). The design moment acting on the cross-section is:

$$M_d = F \cdot h = L_{design} \cdot D \cdot h = 2\,MN/m \cdot 7.2\,m \cdot 6.7\,m \approx 96.5\,MNm \tag{3.32}$$

EC2 (2008) gives an expression for estimation of the cross-sectional steel as given in Eq. 3.33. Note that this procedure is for a rectangular cross-section, as a similar, straightforward approach for a circular pipe was not found. The required steel area for a rectangular cross-section is:

$$A_S = \frac{M_d}{f_{yd} \cdot z} \tag{3.33}$$

where:

- $f_{yd} = f_{yk}/\gamma_s$.
- z is the inner moment arm with z/d is equal to 0.835. d is the necessary effective height, and found using Eq. 3.34

$$d = \sqrt{\frac{M_d}{K \cdot f_{cd} \cdot b}} \tag{3.34}$$

where:

- d is necessary effective height.
- $K = 0.275$ (Sørensen, 2009, Tab. 4.3).
- $f_{cd} = 19.8$ N/mm^2 for **B500C**.
- b is the cross-sectional width.

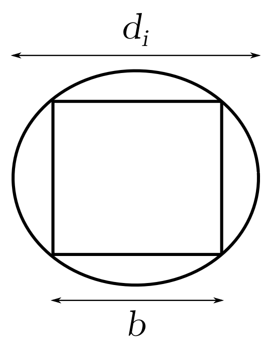


Figure 3.19: Illustration of b used to calculate A_s.

A circle with radius r_i is used when finding b. A square of size $b \cdot b$ is placed within the circle as shown in Fig. 3.19. Concerning distribution of the rebars, the requirement by EC2, NA.8.2(2) is used. Fig. 3.20 shows the distances. It states that the minimum distance between rebars in the same layer should be:

$$a_h \geq max\{2 \cdot \varnothing; d_g + 5\,mm; 20\,mm\} \tag{3.35}$$

Where $\varnothing$ is the rebar diameter and d_g is the maximum aggregate size. The concrete shall cover at least:

$$c_{nom} = c_{min} + \Delta c_{dev} \tag{3.36}$$

where Δc_{dev} is allowed deviation = 10 mm. The minimum cover c_{min} is max$\{c_{min,b}; c_{min,dur}; 10\,mm\}$. $c_{min,b}$ is max$\{\varnothing; 10\,mm\}$ and $c_{min,dur}$ is 25 mm.

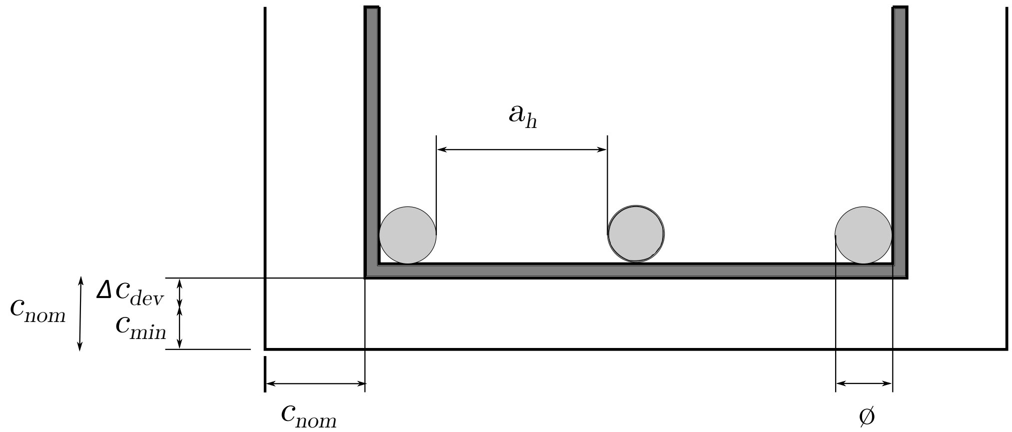


Figure 3.20: Required distances between concrete and rebars. Figure taken from EC2.

3.4.2 Stress range

Two forces acting on the cross-section are considered; The bending moment due to ice forces at the MWL and the structural weight of the lighthouse (Fig. 3.16). Following assumptions are made:

- Plane sections remain plane (Navier's hypothesis).
- The tensile strength of the concrete is neglected.
- No reduction of the effective cross-section is applied during the life time.
- Only positive stress ranges are considered in this model.

The stress action on the considered cross-section is in this model simply expressed as:

$$\sigma = \sigma_N + \sigma_M = \frac{N_{axial}}{A} + \frac{M}{I_z} \cdot y \tag{3.37}$$

where:

- A is the cross-sectional area (m^2).
- M is the moment acting on the cross-section (Nm).
- N_{axial} is the axial force acting on the cross-section (N).
- I_z is the 2^{nd} moment of area of the cross-section (m^4).
- y is the distance from the center to considered point (m).

The axial compressive force, N_{axial}, acting on the cross-section is due to the weight of the lighthouse above the cross-section. Note that M is time dependent, as it is indirectly dependent of the ice thickness (Eq. 3.5).

The previous section assumed a rectangle when calculating the required amount of steel. Now it will be assumed that the steel is distributed as a steel ring inside the concrete. The radius of this ring is assumed to be the the mean radius of the concrete pipe. Using the circumference and the area of steel, the thickness of the ring is easily found. The second moment of area is then calculated assuming a thin walled pipe, i.e. using the mean radius R. M is simply taken to be the ice force acting multiplied by the distance to the cross-section. The moment is then varying with varying ice force, which was described in section 3.1.4. The maximum stress is expressed as:

$$\sigma_{max} = \frac{F_{max} \cdot h}{\pi t R^2} - \frac{N_{axial}}{2\pi t R} \tag{3.38}$$

While the minimum stress will be:

$$\sigma_{min} = \frac{F_{min} \cdot h}{\pi t R^2} - \frac{N_{axial}}{2\pi t R} \tag{3.39}$$

where:

- F_{max} and F_{min} are found using Eq. 3.11.
- R is the mean radius of the cross-section (m).
- h is the distance from acting force to the cross-section (m).
- t is the thickness of the considered pipe (m).

3.5 SDOF system

An approach which has often been used when performing dynamic analysis, is to use a single-degree of freedom (SDOF) model. Using a SDOF model, it is relatively easy to evaluate the response. Higher and lower modes may influence the structure. Often, using a SDOF system is sufficient when analysing the dynamic response (Sodhi, 1988). The Norströmsgrund lighthouse is seen to be susceptible to the lowest modes of vibrations. The 1^{st} mode of vibration is shown in Fig. 3.21. This section will show how the equation for a SDOF system representing the MWL response is obtained. The dynamic equation for a forced oscillation is:

$$\mathbf{M}\ddot{\mathbf{U}}(t) + \mathbf{C}\dot{\mathbf{U}}(t) + \mathbf{K}\mathbf{U}(t) = \mathbf{F}(t) \tag{3.40}$$

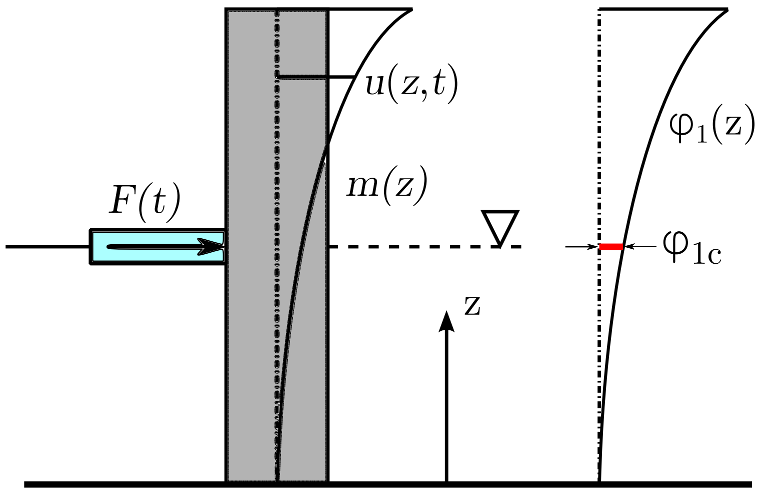


Figure 3.21: 1^{st} eigenmode of the lighthouse.

Here **M**, **C** and **K** are structural mass, damping and stiffness matrices respectively, while $\mathbf{U}$(t), $\dot{\mathbf{U}}$(t) and $\ddot{\mathbf{U}}$(t) are displacement, velocity and acceleration vectors. **F**(t) is the force vector acting on the structure. Given that the structure has n mode shapes, the eigenvectors can be used in a linear combination to express an arbitrary displacement. Each equation of equilibrium for the n different mode shapes are solved separately and the total solution is a sum of the contribution from each mode shape.

$$\mathbf{U}(t) = \sum_{i=1}^{n} \phi_i \cdot r_i(t) = \phi \cdot \mathbf{R}(t) \tag{3.41}$$

where:

- ϕ_i is the vibration mode number i.
- r_i(t) is the time dependent normal coordinate number i.
- ϕ is the matrix of n vibration modes.
- **R** is the designated normal coordinates vector.

3.5.1 Shape function

Kärnä and Turunen (1990) reported that the structure acts as a filter on the applied forces and no indications of resonance associated with other modes than the fundamental modes are observed during FLC. In other words, lowest eigenmodes are of importance as FLC is often seen to have a frequency close to these eigenfrequencies. Shape function for the 1^{st} eigenmode is taken from Guo (2012), shown in Fig. 3.22.

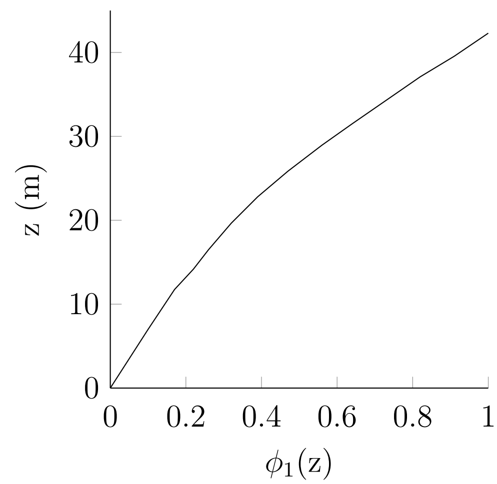


Figure 3.22: 1^{st} eigenmode by Guo (2012).

For the first eigenmode, the boundary conditions for the model of the gravity based lighthouse are assumed to be:

- Modal amplitude $\phi_1 = 0$ at $z = 0.0$ m.
- Modal amplitude $\phi_1 = 1$ at $z = 42.3$ m.

The shape function by Guo (2012) is seen to satisfy these boundary conditions. The value of interest, is the amplitude at the MWL. Two different values are found. One from the Fig. 3.22 and another from a report by Kärnä et al. (2004).

- $\phi_{1c} = 0.22000$ (Guo, 2012).
- $\phi_{1c} = 0.12123$ (Kärnä et al., 2004).

3.5.2 Mass

The modal mass M_n is calculated as:

$$M_n^* = \int m(z) \cdot \phi_n^2(z) dz \tag{3.42}$$

and

$$M_n = \frac{M_n^*}{\phi_n^2} \tag{3.43}$$

Here *m(z)* is the mass per unit length along elevation z, while ϕ_n is the mode shape function along the structure.

In addition to the mass from the structure itself, there is a contribution (added mass) from both the surrounding water and the incoming ice sheet. Here it is assumed that reported values from different researches are including these contributions.

3.5.3 Damping

The damping ratio for Norströmsgrund associated with the 1^{st} order eigenmode, is reported to be in the range of 0.02 - 0.04. This is based on values by different researches, e.g. Kärnä and Qu (2005) and Bjerkås et al. (2013). The damping ratio is defined as:

$$\xi_n = \frac{C_n}{2\sqrt{M_n \cdot K_n}} \tag{3.44}$$

where:

- ξ_n is the damping ratio (-).
- M_n is the mass (kg).
- K_n is the spring constant (N/m).
- C_n is the damping coefficient (kg/s).

The expression for damping in a SDOF model is written as:

$$C_n = 2 \cdot \xi_n \cdot \omega_n \cdot M_n \tag{3.45}$$

In addition to the damping provided by the structure itself, there are contributions to the damping due to the surrounding water and the ice rubble in vicinity of the waterline. It is assumed that this damping ratio accounts for the effects of ice and water around the lighthouse, as it is derived from measured oscillations.

3.5.4 Stiffness

Modal stiffness is depending on the modal mass M_n and defined as:

$$K_n = \omega_n^2 \cdot M_n \tag{3.46}$$

It is assumed that stiffness from the interaction between the structure and the ice sheet is included in K_n, since this effect was assumed included in M_n. Several researches report the lighthouse waterline stiffness of the lighthouse to be high (Kärnä and Qu, 2005).

3.5.5 SDOF system at waterline level

The purpose of using a SDOF in the present work is to compare computed and measured response. An accelerometer measured the response at 19.65 m. A SDOF model representing the response at the MWL is established. Transforming Eq. 3.40 using the result from an eigenvalue analysis, it is re-written as:

$$\mathbf{M}^*\ddot{\mathbf{U}}(t) + \mathbf{C}^*\dot{\mathbf{U}}(t) + \mathbf{K}^*\mathbf{U}(t) = \mathbf{Q}(t) \tag{3.47}$$

where

- $\mathbf{M}^* = \text{diag}(M_n)$.
- $\mathbf{C}^* = \text{diag}(2 \cdot \xi_n \cdot \omega_n \cdot M_n)$.
- $\mathbf{M}^* = \text{diag}(\omega_n^2 \cdot M_n)$.
- $\mathbf{Q} = \phi^T \cdot F_D$.
- $\omega_\mathbf{n} = 2 \cdot \pi \cdot f_n$ (angular frequency for the eigenmode n).

Using only the first mode of vibration, Eq. 3.47 is reduced to:

$$M_1^*\ddot{R}_1(t) + 2 \cdot \xi_1 \cdot \omega_1 \cdot M_1^*\dot{R}_1(t) + \omega_1^2 \cdot M_1^* R_1(t) = Q_1(t) \tag{3.48}$$

Here, the only load considered is acting at the MWL, thus reducing the force vector:

$$Q_{1c}(t) = \phi_{1c}^T \cdot F_D(t) \tag{3.49}$$

The displacement will be:

$$u_c(t) = \phi_{1c} \cdot R_1(t) \tag{3.50}$$

Eigenmode is scaled to unity at water level:

$$\psi(z) = \frac{\phi_1(z)}{\phi_{1c}} \ , \ \phi_{1c} = \phi(z_c) \tag{3.51}$$

Inserting Eqs. 3.50 and 3.51 into Eq. 3.48:

$$\phi_{1c}^2 \int m(z) \cdot \psi^2(z) dz \cdot \{\frac{1}{\phi_{1c}}[u_c(t) + 2 \cdot \xi_1 \cdot \omega_1 \cdot u_c(t)]\} = \phi_{1c} \cdot F_D(t) \tag{3.52}$$

Using Eqs. 3.42, 3.43, 3.45, 3.46 and 3.50, Eq. 3.52 is re-written to give the response at the waterline ($n = 1$ in our case):

$$m_n \ddot{u}_c(t) + c_n \dot{u}_c(t) + k_n u_c(t) = F_D(t) \tag{3.53}$$

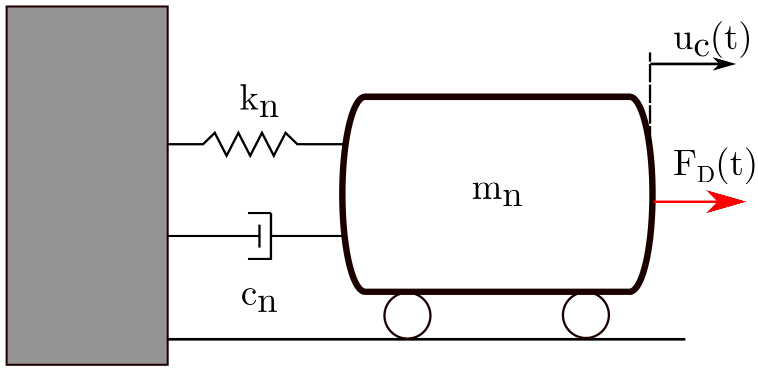


Figure 3.23: SDOF model for dynamic analysis at the waterline level.

Small letters are used in Eq. 3.53 to emphasize that m, c, and k are scalars. Solution to Eq. 3.53 w.r.t. u_c(t) is in this thesis found using the Newmark-beta method (Chopra, 2007). Solving equations 3.34 and 3.35 gives the waterline response:

$$\dot{u}_{i+1} = \dot{u}_i + [(1-\gamma)\Delta t] \cdot \ddot{u}_i + [\gamma \Delta t] \cdot \ddot{u}_{i+1} \tag{3.54}$$

and

$$u_{i+1} = u_i + (\Delta t) \cdot u_i + [(0.5-\beta)(\Delta t)^2] \cdot \ddot{u}_i + [\beta \cdot (\Delta t)^2] \cdot \ddot{u}_{i+1} \tag{3.55}$$

where the constant are set to $\beta = 1/4$ and $\gamma = 1/2$. This gives the constant acceleration method, which has an unconditionally stable solution. The force function was described in section 3.1.4.

4 Results

All the results have been based on using field data from the winter of 2002/2003, which was collected during the STRICE-campaign. The measured environmental parameters used as input for the model in this thesis were:

- Ambient air temperature.
- Wind direction.
- Wind speed.

These parameters have been measured as averages over 3-hour intervals from the 1^{st} of July 2002 to the 30^{th} of June 2003. The winter season has in this thesis been defined to last from the 20^{th} of October to the 26^{th} of May. The result of this was that the model only included ice forces acting on the lighthouse in this period. The main parameters related to the mechanics and dynamics of the ice-structure system was then calculated. Using these results, the number of vibration cycles and stress ranges at different points in time were found. The Miner-sum (Eq. 3.31) was estimated based on these results.

In addition, a single event with SS vibrations was used as comparison for the results produced by the simple SDOF model. This event occurred the 30^{th} March 2003, and was a known case of resonance. The following parameters were measured during this event:

- Total force acting on the load panels.
- Structural displacement and acceleration by a tilt-meter and an accelerometer.
- Ice thickness by an EM device.

The purpose of using a SDOF model was to see if a simple model could represent the measured response of the lighthouse structure. The input parameters for the SDOF system representing the waterline of Norströmsgrund were taken from Kärnä et al. (2004) and Guo (2012), see Tab. 4.1. These values were used (unless otherwise specified) as input for the SDOF model and other equations which required modal values.

Table 4.1: Reported values for SDOF system at the MWL.

Report	f_1(Hz)	ϕ_{1c} (-)	ξ_1 (-)	M_{11} (kg)
Kärnä et al. (2004)	2.89	0.12123	0.02	$1.72173E05$
Guo (2012)	2.60	0.22000	0.02	$1.70000E05$

4.1 Calculated ice properties

Estimation of the ice properties were made using the measured air temperature and wind drift vectors. These were measured at the Luleå airport from 1999 to 2003 by the Swedish Meteorological and Hydrological Institute (SMHI). This was carried out using 3 hour intervals. Since the lighthouse location is approximately 50 km southwest of Luleå, it was assumed that the environmental parameters were the same for both places at the time.

From Eq. 3.2 in section 3.1.1 the ice thickness was calculated based on the measured air temperatures and the freezing point of water. The ice thickness calculated using Eq. 4.1 and was averaged over 24 hours. The salinity in the GoB is very low ($< 0.4\%$), hence the freezing point was set to 0 °C, as it is for fresh water. Naturally, the calculated growth was largest in the coldest period, which was January. With rising temperatures, the growth decreased. Ice thawing was also neglected during the defined winter season. A result of this was if the air temperature was above the melting point, the thickness did not decrease and stayed constant during this time. This was the case around late February/early March. The effect of snow providing an isolating layer on top of the ice was neglected. The FDD was estimated to be 1380 this season.

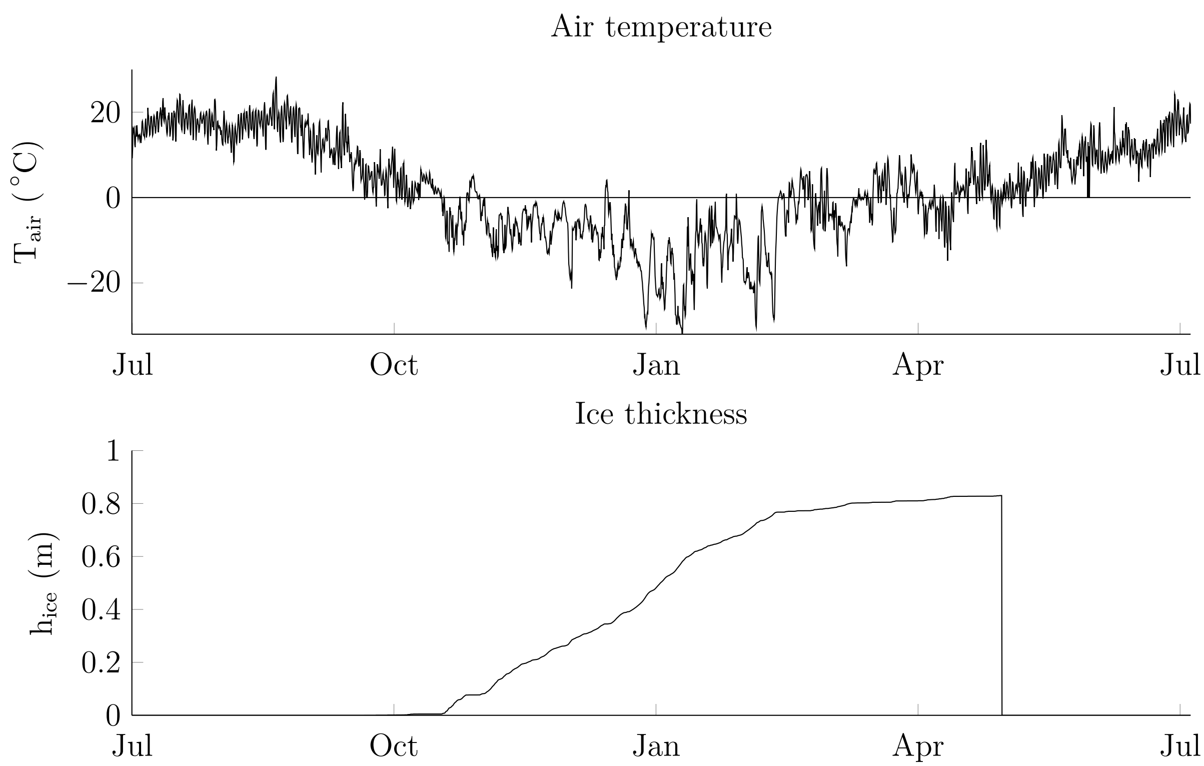


Figure 4.1: Calculated ice thickness given air measured temperature.

After a time with nearly zero growth in late April, the ice thickness was assumed to "melt" instantly (the drop in April, Fig. 4.1). This assumption was made after discussions with Bjerkås (2013). After April the 7^{th}, there were almost no field-observed vibrations with accelerations above 0.4 m/s^2. Thus it was not of interest to find an estimate of cycles after the reported stop of observed vibrations. As the model used the ice thickness as one of the main parameters, it was set to zero in order to exclude results in this time period.

The measured wind drift direction and velocity for the winter season of 2002/2003 was measured at 10 m height by SMHI as shown in Fig. A.1. A direction of 360 $^\circ$ meant that the wind was coming from the north, while 270 $^\circ$ meant that the wind was coming from the west. 0 $^\circ$ meant calm conditions. The ice drift direction was found by summation and averaging of the wind drift vectors. This was executed by adding the sine and cosine components of the respective vectors with amplitudes, finding the average and calculating the resulting vector. A time interval of 24 hours was used when summing and averaging vectors. The ice drift was then calculated by the use of the Ekmans assumptions, setting the relative angle between wind and ice drift to be 30 $^\circ$ clockwise. The ice velocity was found by using the measured wind speed and the coefficient (Eq. 3.3) defined by Zubov (1943). Resulting directions and velocities were found as shown in Fig. 4.2. The global pressure was calculated from late November and forward.

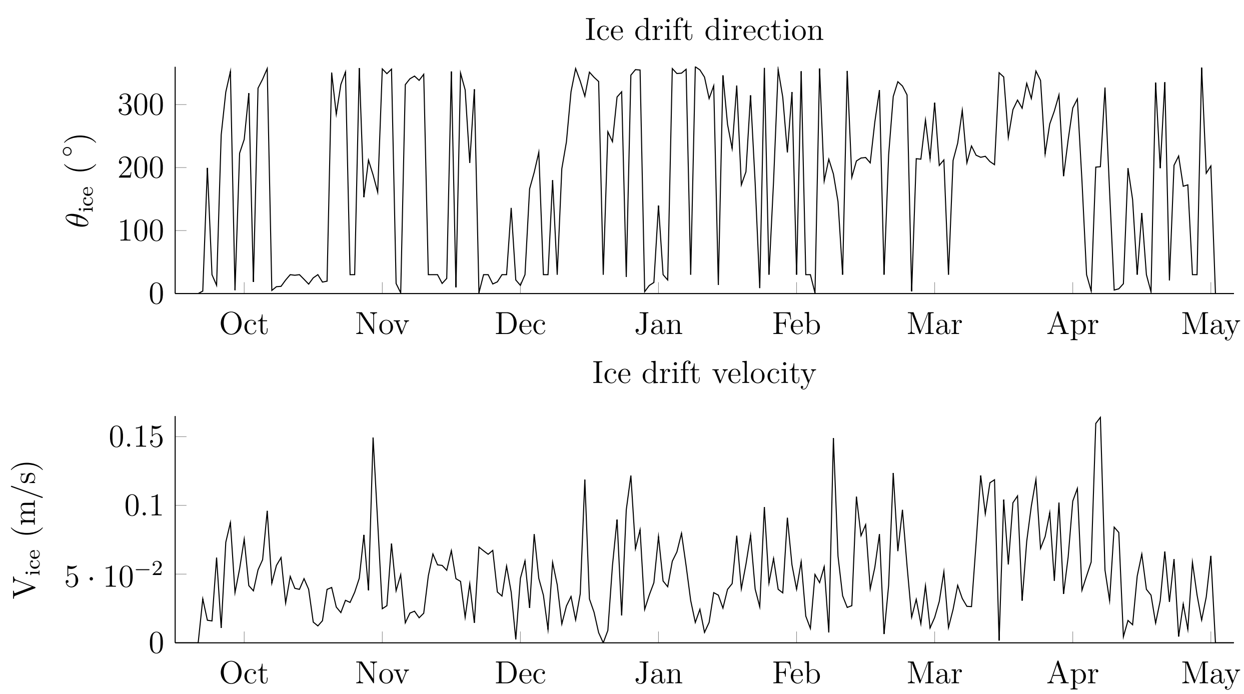


Figure 4.2: Daily averages of ice drift direction and velocity. Note that 360 $^\circ$ is North, while 0 $^\circ$ is calm conditions.

This was based on observing that the aspect ratio between structural diameter and ice thickness (Fig. 4.5) was below 40 in this time period and forwards.

Global ice pressure was found from Eq. 3.5 as shown in Fig. 4.3. The strength coefficient, C_R, used in Eq. 3.5, was chosen to be 1.8 MPa. The global pressure decreased from approximately 2.2 MPa in November to 1.5 MPa in January. Then the pressure converged toward 1.36 MPa in the late parts of the winter season.

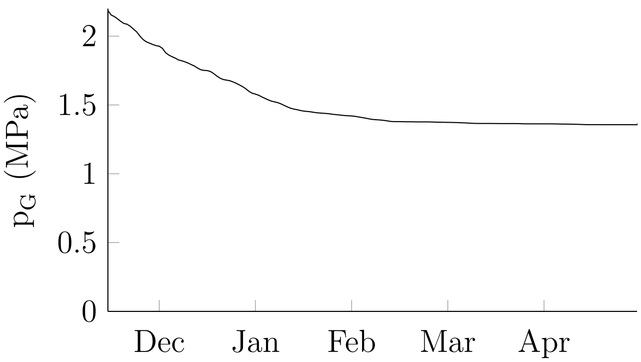


Figure 4.3: Global ice pressure.

4.2 Onset criteria for FLC

First, the susceptibility of the model to dynamic instability was checked. This was done using Eq. 3.18 defined in section 3.2.5. This equation used the mass, eigenfrequency and critical damping related to the model. Values used in Eq. 3.18 were taken from Tab. 4.1. The critical damping was then plotted vs. the ice-action as shown in Fig. 4.4. Both value sets indicated that the SDOF model of the waterline was susceptible to SS vibrations the entire season. The input parameter which differed most in value in the two sets, was the normalized amplitude. This gave a large difference in the ice action, as seen in Fig. 4.4.

The general onset criteria which applied for all methods presented in Tab. 4.2 were:

- Having ice crushing as failure mode (section 3.2.1).
- Having a compatible ice drift direction (section 3.2.2).
- Having a C-shaped ice temperature profile as a result of an ice growth less than 1 cm the proceeding week (section 3.2.3).
- Having a static waterline displacement of at least 1 mm (section 3.2.4).

Determining the ice failure mode was done by using the ratio D/h_{ice}. The ratio was found to decrease over the winter season as shown in Fig. 4.5. The ice thickness vector has been used together with the structural diameter to find the ratio between the two. The sudden drop in mid-April corresponded to setting the ice thickness to be zero from this point forward.

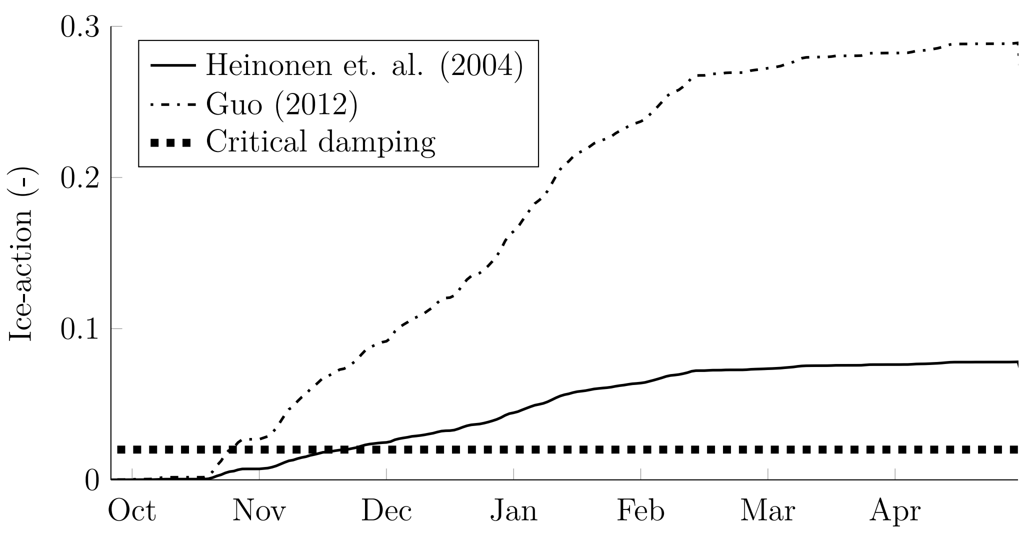


Figure 4.4: Susceptibility to dynamic instability.

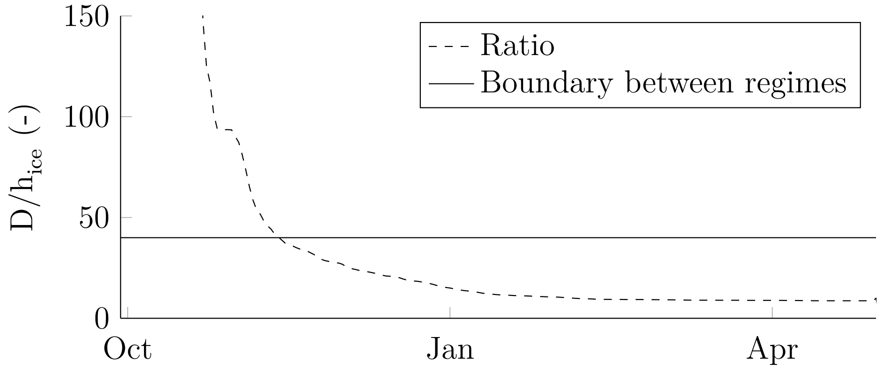


Figure 4.5: Ratio between structural diameter and ice thickness.

The ratio was seen to be below 40, which was the boundary between bending and crushing failure, in late November. The high values around October was due to a ice thickness which was nearly zero, and these values were not used as they were ruled out by the ratio criterion (section 3.2.1).

The onset criterion for drift direction stated that the ice drift had to be between 300 ° and 45 °. If the drift direction was not in this range, the model assumed that the ice sheet was not drifting. The daily average drift directions shown in Fig. 4.2 were used when checking if the ice drift was compatible.

The growth criterion was checked by comparing the ice thickness vector indices with intervals of one week. Due to the assumption of no thawing and a constant thickness in times of positive air temperatures, the thickness never decreased. It was straightforward to find the growth after one week, using the results shown in Fig. 4.1. If the difference was less than 1 cm, the ice temperature profile at that time was assumed to be C-shaped (Fig. 3.7), fulfilling the ice temperature criterion.

Having found the ice global pressure from Eq. 3.5 and the global force from Eq. 3.8, the static waterline displacement was investigated. By using Eq. 3.17, the static waterline displacement (depending on h_{ice}) was found as shown in Fig. 4.6. From Fig. 4.6 it was observed that the static waterline displacement was above 1 mm nearly the entire season, having used the design load, F_G, from ISO19906 (2010). The steepness of the curve in November was a result of the assumption of demanding ice crushing to be the failure mode. In other words, the static displacement was only checked in the range which D/h_{ice} was below 40. The expression for the global ice pressure (Eq. 3.5) was assumed valid from this point forward.

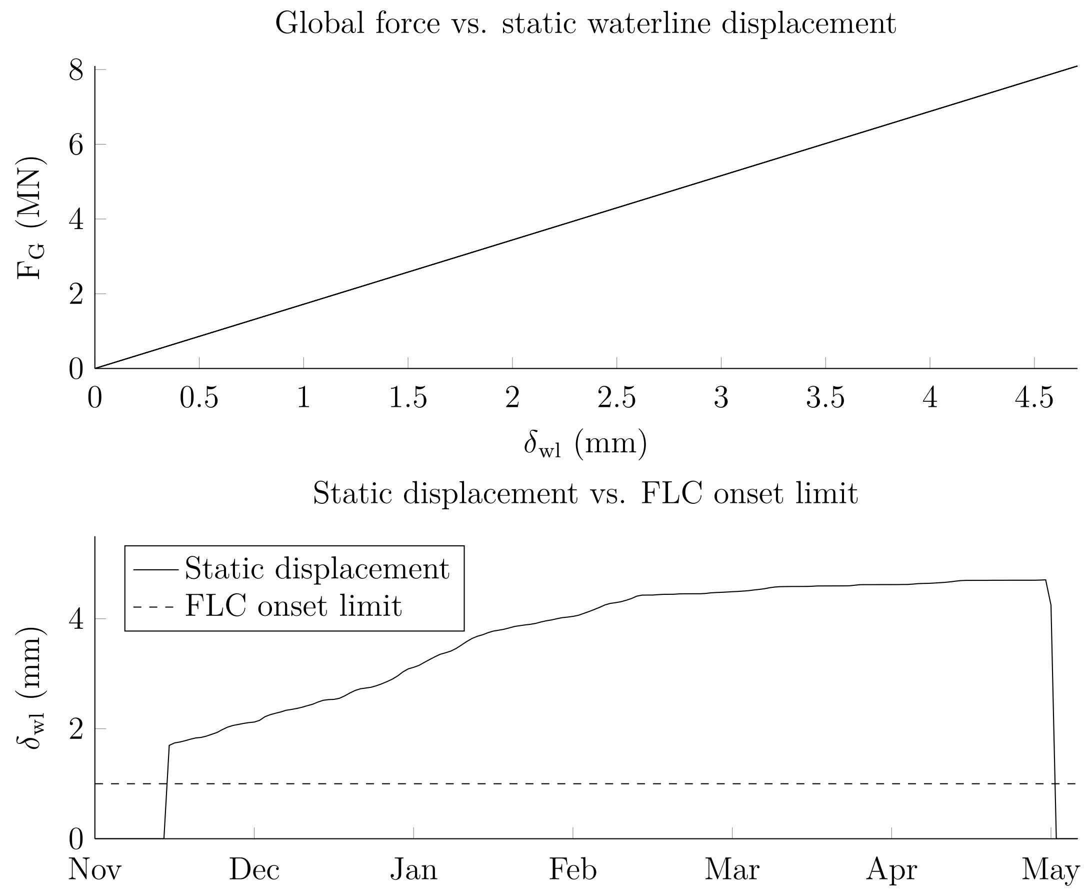


Figure 4.6: Waterline displacement as a function of global ice force.

4.3 Total number of cycles

Five different methods were used together with the general FLC onset criteria. The methods are presented in Tab. 4.2. These methods were used to find the total number of fulfilments of the onset criteria. In Fig. 4.7 the fulfilments at different times were plotted as points. As the ice thickness and the ice drift were taken as daily averages, the FLC conditions were assumed to last the entire day for which the criteria were fulfilled.

Table 4.2: Different methods used to find onset criteria.

	Method	**Eq.**	**Source**
1	Using dimensionless groups with diameter and velocity	3.20	Palmer et al. (2009)
2	Using dimensionless groups with ice thickness and velocity	3.19	Palmer et al. (2009)
3	Case specific method using ice thickness and ice velocity	3.22	Bjerkås and Skiple (2005)
4	Case specific method using temperature and ice velocity	3.21	Bjerkås and Skiple (2005)
5	Using dynamic instability criterion	3.18	ISO19906 (2010)

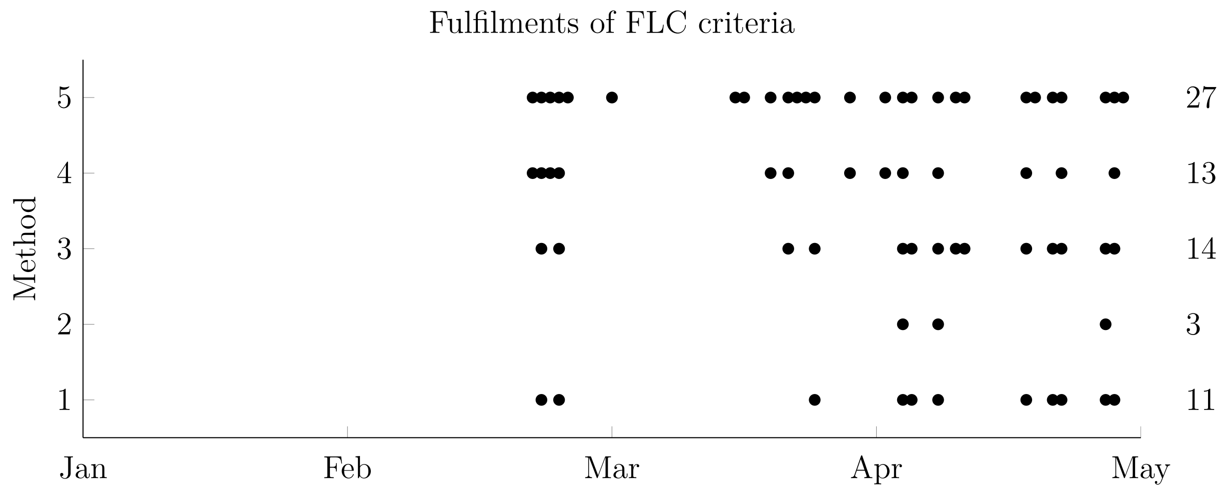


Figure 4.7: Fulfilments of IC criteria for different methods. The numbers on the right axis show the total number of fulfilments.

A result of this was the model calculated a high number of cycles. The natural frequency f_n was important when finding accumulated cycles. Here it was assumed that the lighthouse vibrated with the first mode eigenmode. A value of 2.89 Hz for f_1 was used (Tab. 4.1).
The five different methods in Fig. 4.8 were the same as defined in Tab. 4.2. For the five different methods, the accumulated vibrations over the season were calculated as shown in Fig. 4.8. For method 5, the accumulated number of vibration cycles was increasing almost constantly from the initiation in March to the end of winter season. Methods 1, 3 and 4 followed the same trend, except that they had a lower number of onset fulfilments (Fig. 4.7).

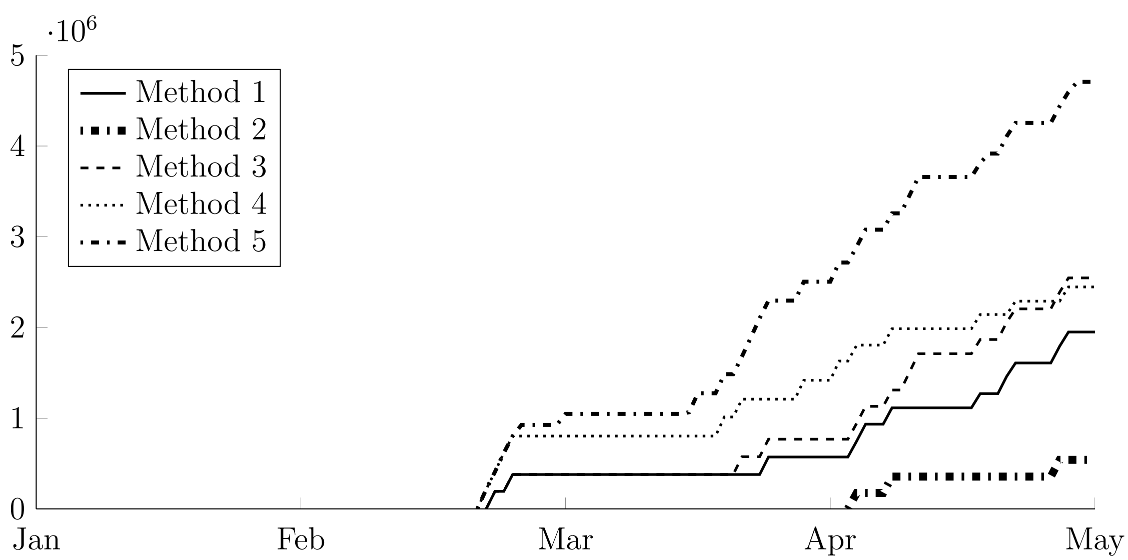


Figure 4.8: Accumulated number of cycles throughout the season.

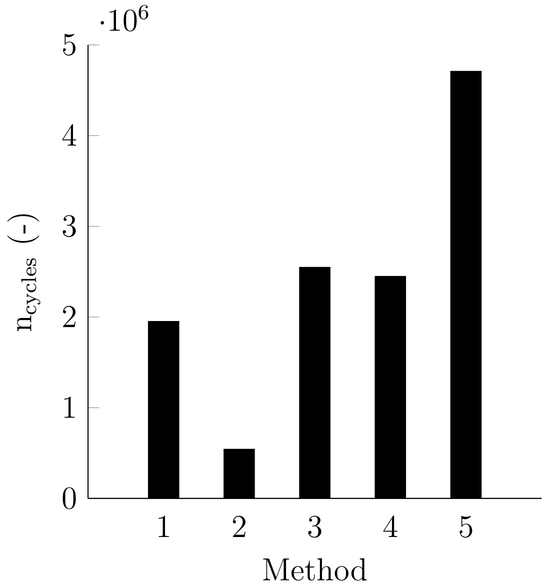


Figure 4.9: Total number of cycles for the different methods.

Method 2 gave only three days with fulfilled onset criteria, which was quite low compared to the other methods. The total numbers of accumulated cycles for 2002/2003 were found as shown in Fig. 4.9.

It was also of interest to see the number of cycles at separate points in time, shown in Fig. 4.10. The column heights were seen to vary at different points in time. This was due to the assumption of a varying lock-in time, which was depending on the ice velocity and the ice thickness (Eqs. 3.23, 3.26 and 3.27). Method 5 was seen to give a much higher number of vibration cycles than the other methods, while method 2 gave a low value compared to the others. The majority of cycles produced in the model was seen to be around April, which corresponded with in-situ measurements (Bjerkås et al., 2013).

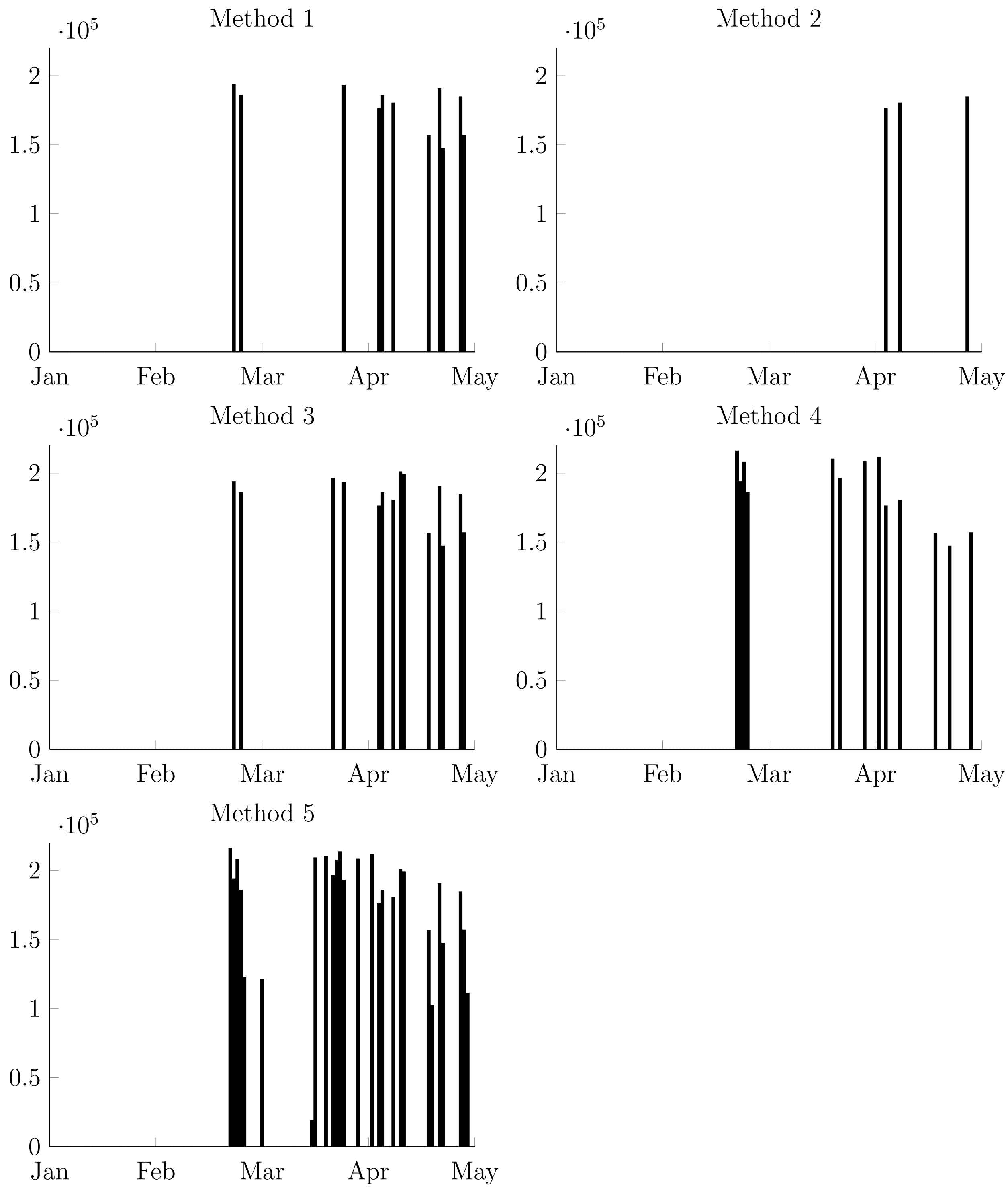


Figure 4.10: Cycles at different points in time for the different methods. The y-axis shows number of cycles.

4.4 Cross-section properties

Before the acting stress on the cross-section could be estimated, properties of the cross-section were needed. The amount of steel was estimated using the design load at the MWL. This load was defined in section 3.4.1 to be $M_d = 96.5$ MNm. The width of the cross-section, b, was determined by placing a square inside the pipe, as described in section 3.4.1. With $d_i = 3100\,mm$, the b was found to be 2192 mm. By using the design moment and the properties of **B500C** type steel, the effective height d was found using Eq. 3.34:

$$d = \sqrt{\frac{96.5\,MNm}{0.275 \cdot 19.88\,MN/m^2 \cdot 2.192m}} \approx 2843\,mm \tag{4.1}$$

Then the required steel area was obtained by Eq. 3.33:

$$A_S = \frac{96.5 \cdot 10^9\,Nmm}{\frac{500\,N/mm^2}{1.15} \cdot 0.835 \cdot 2843\,mm} = 169547\,mm^2 \tag{4.2}$$

With the cross-sectional area being:

$$A_{cs} = \pi \cdot (r_o^2 - r_i^2) = \pi \cdot (3600^2 - 3100^2)\,mm^2 \approx 1.052434 \cdot 10^7\,mm^2 \tag{4.3}$$

Trying a diameter of $\varnothing = 25$ mm, the minimum number of reinforcing steel bars was found to be:

$$n_{rebars} = \frac{A_s}{A_{rebar}} = \frac{93496\,mm^2}{\pi \cdot (0.5 \cdot 25)\,mm^2} \approx 191\,(-) \tag{4.4}$$

It was assumed that the rebars were placed in two layers. Each bar occupied:

$$\varnothing + a_h = 25\,mm + 50\,mm = 75\,mm \tag{4.5}$$

according to the requirements by EC2 (Fig. 3.20). Concerning cover of concrete, the minimum requirement was:

$$c_{nom} = c_{min} + \Delta c_{dev} = 25\,mm + 10\,mm = 35\,mm. \tag{4.6}$$

The distance and cover requirements were seen to be satisfied (calculations in App. A.3).

Having calculated the area of steel for the rectangle in Fig. 3.19, it was assumed that the rebars were replaced by a steel ring with the same total cross-sectional area. The thickness of this ring was found using the mean radius of the ring:

$$t_{steelring} = \frac{A_S}{2 \cdot \pi \cdot R} = \frac{169547}{2 \cdot \pi \cdot 3350} = 8.05\,mm \tag{4.7}$$

With the rebars replaced by a steel ring of diameter R and thickness $t_{steelring}$, the stress acting on the cross-section was estimated using Eqs. 3.38 and 3.39.

4.5 Fatigue damage

The fatigue life of the model was estimated using three different cases:

- Vibrations throughout the entire season given an ice thickness larger than 0.2 m. The winter season was assumed to be from February 1^{st} to April 7^{th}. This is later referred to as $\delta_{extreme}$.
- Using the onset criteria for FLC presented in this thesis (Tab. 4.2), referred to as δ_{onset1} etc.
- Utilization of the observed number of cycles for the winter season of 2002/2003 (Bjerkås et al., 2013), referred to as δ_{obs}.

First, the stresses acting on the circular cross-section were estimated. The alternating moment was obtained using the ice thickness and the force function for FLC conditions (Eq. 3.11). The q-factor in Eq. 3.11 was set to 0.5 to obtain the largest stress range, i.e. being conservative. Axial compression acting on the cross-section was found by calculating the weight of the lighthouse above the cross-section. The weight was found to be 5.56 MN, see Tab. A.1 in appendix A.2 for detailed calculation.

It was then checked if any of the calculated force levels exceeded the design load. The design load at the MWL was 2 MN/m, i.e. a global force equal to $F_{G,max} = 2\text{MN/m} \cdot 7.2$ m $= 14.4$ MN. Maximum estimated global force using Eqs. 3.2 and 3.8 was found to be $F_{G,max} = 8.1$ MN (Fig. 4.11), thus the lighthouse was seen to be dimensioned to withstand quite large loads. Since we only considered tension to be contributing to fatigue, it was also checked that the lowest force acting at MWL gave tension in the cross-section. If the compressive stress was of a larger magnitude than the tension stress, there would not have been any contribution to fatigue. The lowest load was observed to be 2.92 MN. It was assumed that there was no tension in the concrete, and the maximum tension in the steel for this load was calculated as:

$$\sigma_M = \frac{2.92 \cdot 10^6}{\pi \cdot 0.00805 \cdot 3.35^2} \approx 69\, MPa \tag{4.8}$$

The compressive stress provided by the lighthouse weight was quite small compared to this. It was assumed that the concrete was carrying compressive loads together with the reinforcing steel. The static, compressive stress was found to be:

$$\sigma_N = \frac{N}{A_{cs}} = \frac{5.56 \cdot 10^6}{10.52} \approx 0.53\, MPa \tag{4.9}$$

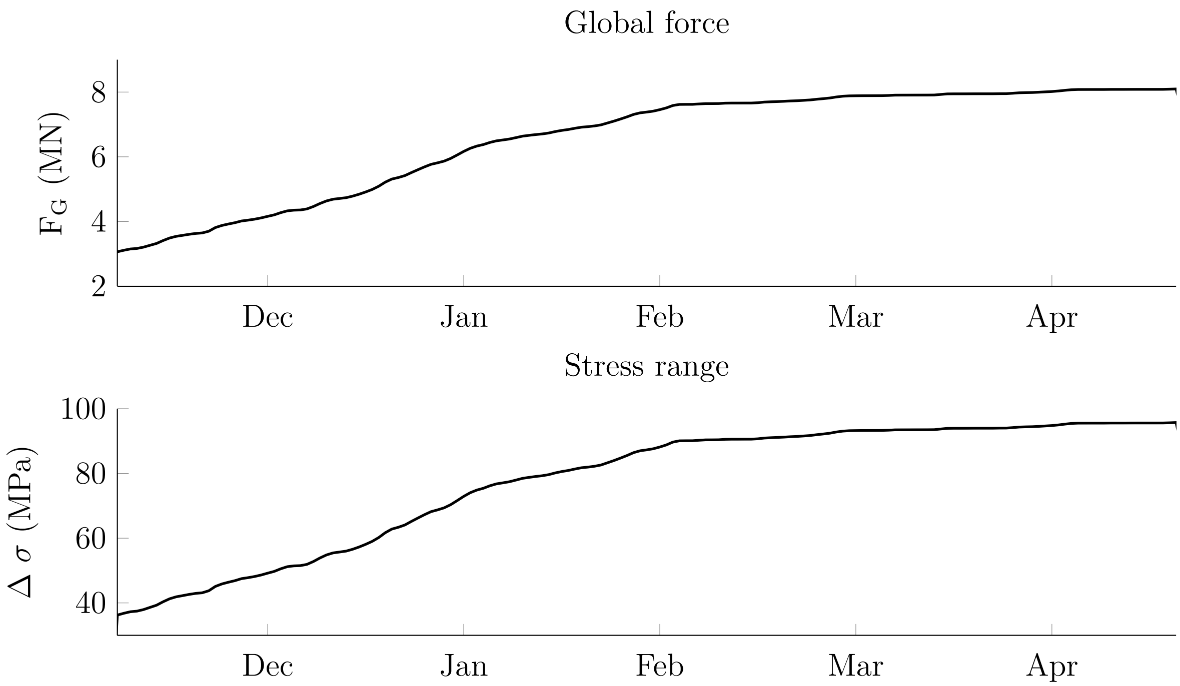


Figure 4.11: Global force (ISO19906, 2010) and calculated stress range. Both are depending on the ice thickness.

The tension stress due to bending was significantly larger than the compressive stress for the minimum load case. The SN-curve given by EC2 (section 3.4.1, Fig. 3.18) was used with the following criteria:

- $b = k_1$ if $\Delta\sigma > 162.5$ MPa.
- $b = k_2$ if $\Delta\sigma < 162.5$ MPa.

Observed number of cycles for the 2002/2003 season was 5090. All of these cycles occurred in late March/early April (Bjerkås et al., 2013, Fig. 10). At this time the ice thickness was seen to be ~ 0.9 m, and this thickness was assumed to be same for all cycles. This meant that all cycles had the same stress range, as the small variations in ice thickness (~ 0.01 m) would have given an insignificant variation in the stress range.

Having used three different approaches, the fatigue life, $t_{fatigue}$, was found as presented in Tab. 4.3.

Case	n_{cycles}	$\sum n_i/N_i$	$t_{fatigue}$	
$\delta_{extreme}$	40950144	0.152502	7	years
δ_{onset1}	1950172	0.014382	70	years
δ_{onset2}	541008	0.004199	239	years
δ_{onset3}	2546437	0.018848	54	years
δ_{onset4}	2446790	0.016535	61	years
δ_{onset5}	4707232	0.033132	31	years
δ_{obs}	5090	0.000203	4911	years

Table 4.3: Estimated fatigue damage by the different methods.

4.6 Measured and calculated structural response

One full-scale event was selected to compare the results from the SDOF model with. This event took place the 30^{th} March 2003 around noon, and Norströmsgrund lighthouse was seen to experience SS vibrations at the time. The time of the entire event lasted from 12:24:30 to 12:27:30, and the time history of the measured force and acceleration vectors are shown in Fig. 4.12. Around 12:26:00 the SS vibrations seemed to have stabilized, before they ended around 12:27:00. The response of the SDOF model described in section 3.5 was compared with the response from 12:26:00 to 12:26:40. The values presented by Kärnä et al. (2004) in Tab. 4.1 were used as input for the SDOF model. Initial conditions were set to:

- u_0 = 0.001 m according to the assumption of having a minimum displacement before FLC started (section 3.2.4).
- $\dot{u}_0$ = 0.0 m/s.
- $\ddot{u}_0$ = 0.0 m/s.

Following the guidelines of ISO19906 (2010), the q-factor in Eq. 3.9 was scaled. This factor influenced the force amplitude (Eq. 3.11). q was scaled such that the response at the waterline amounted to 1.4 times the highest ice velocity for FLC conditions, $v_{i,tr,2}$. Using f_1 = 2.89 Hz, $v_{i,tr,2}$ was found to be:

$$v_{i,tr,2} = \gamma_v \cdot f_1 = 0.060\, m \cdot 2.89\, 1/s = 0.1734\, m/s \tag{4.10}$$

Varying q the maximum velocities and accelerations of the model were found as listed in Tab. 4.4. As expected the highest velocity and acceleration were obtained using q = 0.5. The maximum velocity obtained for q = 0.50 was 0.1350 m/s, which was lower than the value suggested by ISO19906 (2010). The calculated acceleration for q = 0.50 was almost twice the measured one ($\sim$ 1.2 m/s^2, Fig. 4.12), ignoring the extreme values.

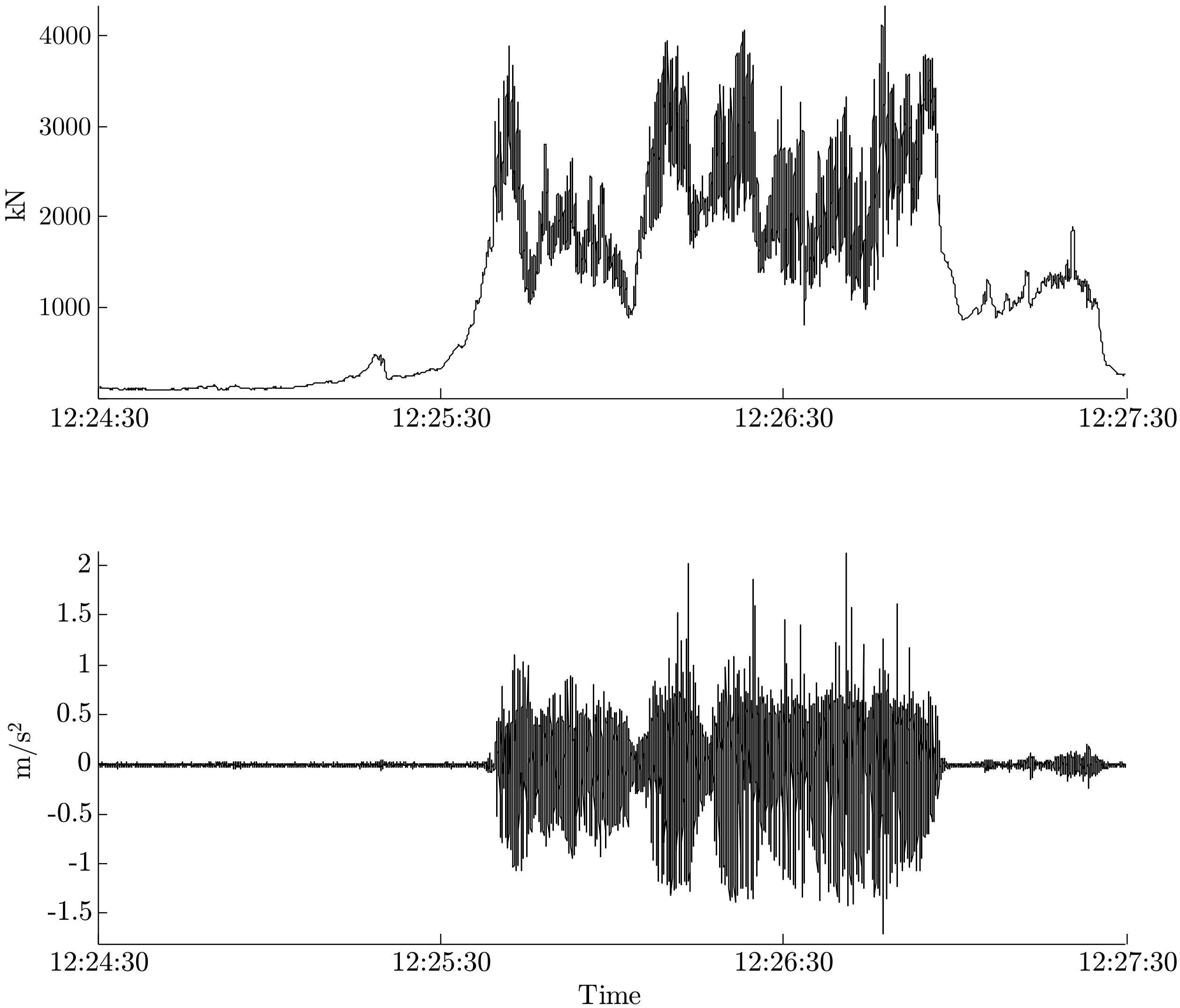


Figure 4.12: Measured forces by load panels and accelerations by accelerometer at 19.65 m for 30^{th} March 2003 event.

For $q = 0.1$, the response velocity was very low, and the acceleration was about half the measured one. It was a problem satisfying the velocity criterion by ISO19906 (2010) and at the same time obtaining an acceleration close to the observed acceleration.

Table 4.4: Maximum model response for different values of q

q	0.10	0.15	0.20	0.25	0.30	0.35	0.40	0.45	0.50
$\dot{u}_{max}\,(m/s)$	0.0273	0.0408	0.0542	0.0677	0.0811	0.0946	0.1081	0.1215	0.1350
$\ddot{u}_{max}\,(m/s^2)$	0.5228	0.7809	1.0390	1.2970	1.5551	1.8132	2.0712	2.3293	2.5873

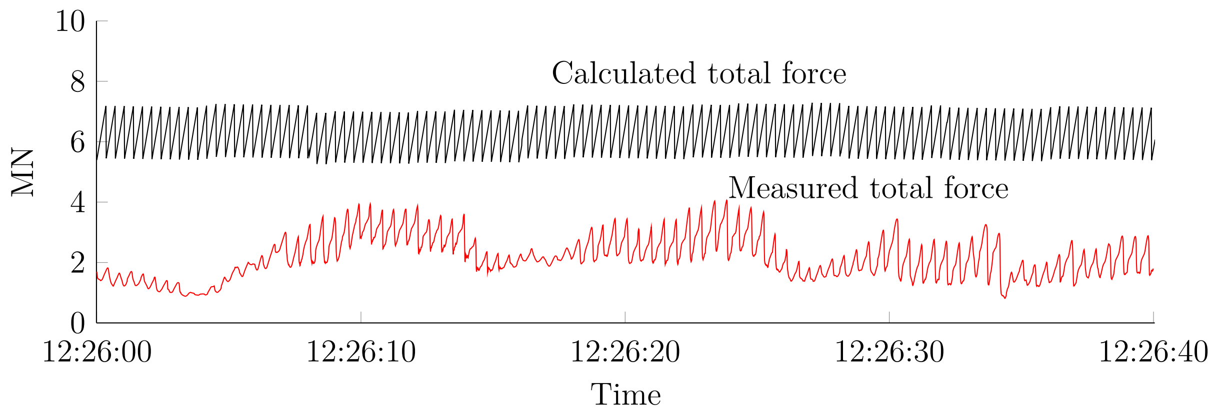


Figure 4.13: Calculated and measured forces for the 30 March 2003 event.

A value of $q = 0.5$ was too high concerning acceleration, while $q = 0.1$ was too low.
The force function used for the SDOF model was calculated using the measured ice thickness together with Eqs. 3.5 and 3.11. q was set to 0.25. This was basd on seeing that load amplitude magnitudes were seen to be corresponding well for this q-value (Fig. 4.13). Even though the model used the in-situ measured ice thickness (Fig. A.2), it was not able to reproduce the fluctuating force levels.

Acceleration of the model was then plotted vs. the measured acceleration. Note that the model calculated the acceleration at 14.20 m above sea bed, while the measured acceleration was taken at 19.65 m. The measured acceleration was seen to follow the load history. As the simple SDOF method used the Newmark-beta method with $\beta = 1/4$ and $\gamma = 1/2$, the solution was unconditionally stable. Small changes in the load history gave insignificant changes in the calculated acceleration. However, the acceleration amplitudes of the SDOF model was seen to be in the same range as the measured ones. Finally, single-sided amplitude spectra for both measured and computed accelerations were found using fast Fourier transform (FFT), see Fig. 4.15. A peak close to ~ 2.9 Hz was seen for the calculated acceleration, which was the input frequency. The measured response showed two distinct peaks. The first eigenfrequency was seen to be around 2.3 Hz, while the second one was at approx. 4.2 Hz.

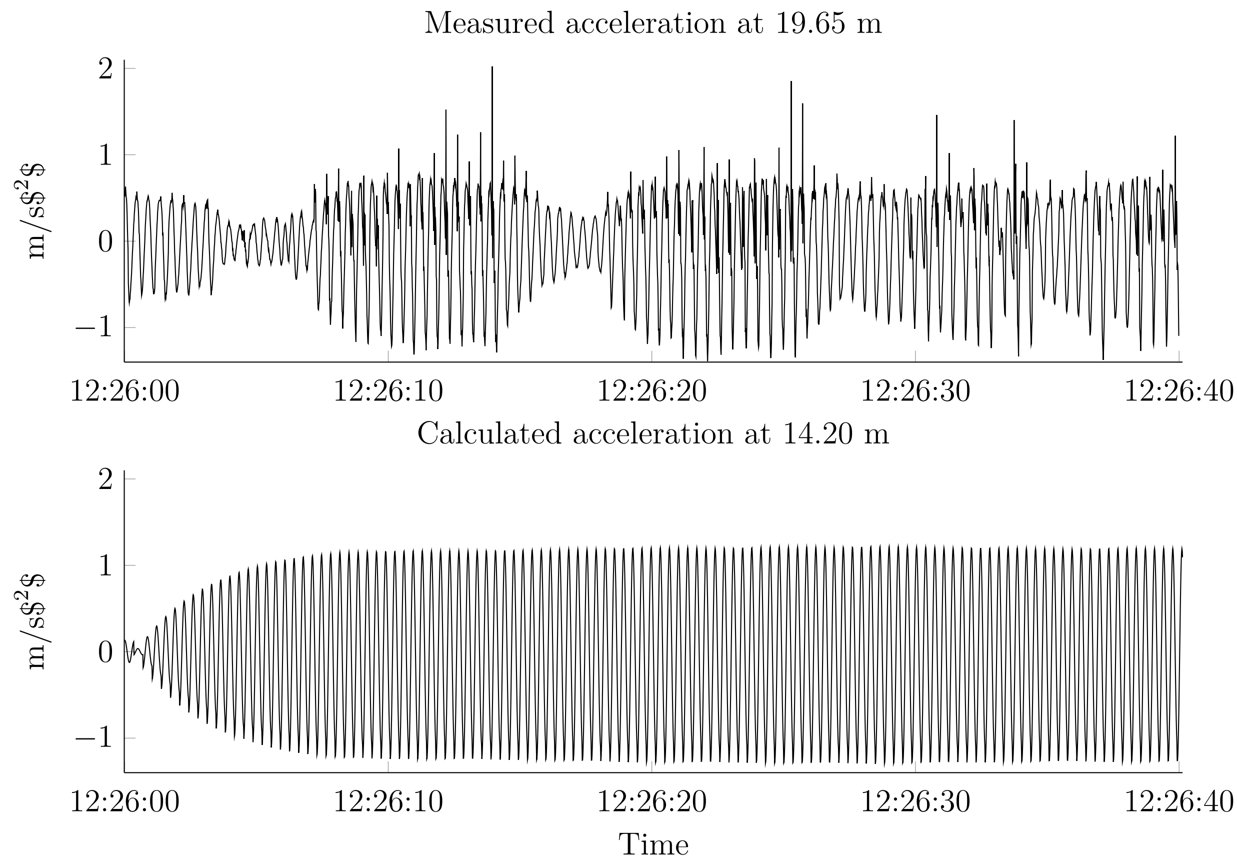


Figure 4.14: Calculated and measured accelerations for the 30 March 2003 event.

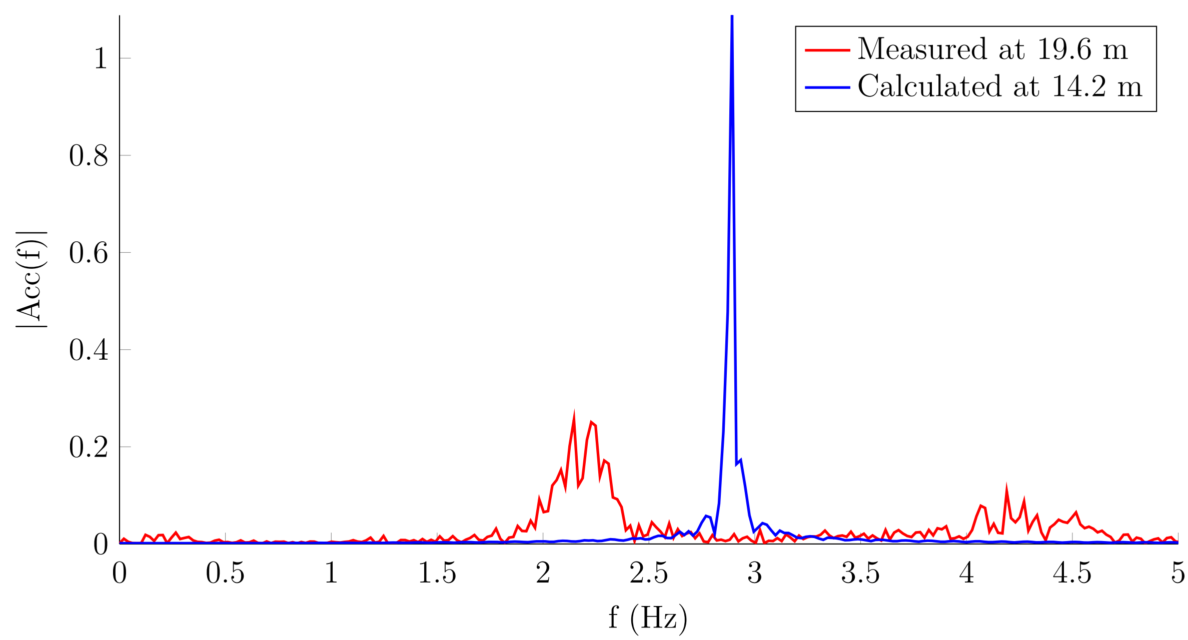


Figure 4.15: Single-sided amplitude spectra found using FFT.

5 Discussion

The purpose of the model developed in the present work was to estimate the fatigue damage due to ice action. Measured environmental parameters were used as input for the model, which calculated the mechanical properties of first-year sea ice. These properties were used together with onset criteria for frequency locked-in crushing. Structural properties of the lighthouse were combined with the FLC onset criteria to find accumulated cycles. Using the load history, the Miner-sum was utilized to estimate the fatigue life. In addition, a single event with FLC was selected from available full-scale data. Using reported SDOF values for the lighthouse, the response of a SDOF model was compared with the measured lighthouse response during FLC conditions.

5.1 Ice properties

The calculation of ice thickness and growth was based purely on the ambient air temperature. In reality there are several factors influencing the growth. Errors in the calculated ice thickness, may produce larger errors later in the model used.

Calculated ice sheet thickness was seen to be in range of the measured value for the 2003 season, with a calculated maximum thickness of ~ 0.8 m and observed maximum ~ 0.9 m (Bjerkås et al., 2013, Fig. 10). One error source was that the effects of having a layer of snow on top was neglected in the empirical relation by Zubov (1943). This could be included using a newer, analytical expression by Stefan (1981). Stefans expression included the thermal properties of snow and ice in the thickness calculation. Heat flux between ice, wind and ocean was also neglected in this thesis. The calculated growth stopped around early April (Fig. 4.1), which was corresponding well with what was observed this season (Bjerkås et al., 2013, Fig. 10, page 13).

Drift velocity was seen to be within the same range as values observed in the field. Bjerkås et al. (2013) observed the maximum drift velocities to be approximately 0.4 m/s for the 2003 season at Norströmsgrund. These were the peak values, while the average values were between 0.2 and 0.3 m/s. The maximum velocities obtained by the model was ~ 0.25 m/s. One reason for the model having lower velocities may be that the model used 24 hour averages, i.e. the average of 8 measurements.

The ice drift directions were found using 24 hour averages of the wind directions, and correcting for the geostrophic effect. However, the calculation did not take into account the enormous mass of the ice sheet. If the ice drifted toward south, and the wind suddenly started to blow against north, the ice would not have followed this change immediately due to its inertia of motion.

Calculated uni-axial compressive strength of the ice was seen to be in the range of reported values for first-year ice in subarctic regions, which is about 2 MPa (ISO19906, 2010). The season of 2002/2003 was calculated to have $\sim$ 1400 FDD. An amount of 1400 FDD is by ISO19906 (2010) defined to be in between temperate and subarctic conditions. The ice strain-rate was not used, which would have included the effect of a varying ice drift velocity. A stress/strain curve (Määttänen, 2001) could have been used instead of the expression given by ISO19906 (2010), which gave relatively large global forces. This method of calculating a global force was general, based on data collected on many locations. Estimation of the ice force (Eq. 3.5 in section 3.1.3) was directly depending on the ice thickness. An alternate method is to use the structural design load. A design load used for many Swedish lighthouses in the 1980s and 1990s, was given by Engelbrektson and Janson (1985). The line load was expressed as $F = 4 + 2.3 \cdot D$. A newer method to find an upper boundary for the pressure force, was presented by Bjerkås (2007). Bjerkås gave an expression which was only depending on the waterline diameter, $p = 2.05 \cdot D^{0.06}$. For Norströmsgrund lighthouse this gave $F = 4 + 2.3 \cdot 7.2 = 20.6$ MN and a pressure force of $p = 2.05 \cdot 7.2^{0.06} = 2.3$ MPa respectively. A pressure force of 2.3 MPa is equal to 14.96 MN, using an ice thickness of 0.9 m together with a diameter of 7.2 m. The design load for Norströmsgrund lighthouse was 2 MN/m, corresponding to a design load of 13 MN at the MWL given an ice thickness of 0.9 m. With several approaches to find the global force, the model could have used them parallel.

5.2 Onset criteria and cycles

The model utilized four general criteria, as described in section 3.2. These criteria were based on environmental conditions such as the ice drift and the ice temperature. Having defined basic criteria, different methods checking for possible FLC conditions (Tab. 4.2, section 4.3) were applied.

Early winter season, the model gave fulfilled FLC criteria. This was when the ice barely had started to form. Bjerkås and Skiple (2005) reported few observations of FLC for thickness less than 0.2 m. Thus, fulfilled onset criteria when the ice thickness was less 0.2 m were neglected. The drift direction criterion ruled out many potential FLC events, most likely because the inertia of motion of the ice was ignored. Using the measured ice drift would have yielded better results. On the other hand, the purpose of the model was to use available environmental data to produce estimates of the ice behaviour.

The growth criterion (section 3.2.3) was used as an indicator to check if the temperature profile was likely to be C-shaped. A C-shaped profile suggested that the ice was ductile, which implied a higher probability of having FLC. Even though there was little growth some periods of time mid season, there were not many fulfilments of onset criteria. One reason may be that for higher velocities the ice will be brittle, and thus the ice temperature must be warmer in order to compensate (Bjerkås and Skiple, 2005). With colder temperatures in this period, this may have been a reason for the low number of cycles.

The displacement criterion in section 3.2.4, which required at least 1 mm of displacement before onset FLC, used the global force defined by ISO19906 (2010). The waterline stiffness used was found from a FEM model of the lighthouse by Kärnä et al. (2004). Another way to find the stiffness could be to process the force and displacement measurements statistically.

Five different methods were used to see if criteria for onset of FLC were fulfilled (Tab. 4.2). Three were general, while two were case specific. Method 1 by ISO19906 (2010) was a generalized method and indicated that the lighthouse was very susceptible to resonant-like vibrations (Fig. 4.4). Due to the fact that the lighthouse is not rigid, deflections and vibrations have a significant effect (Kärnä et al., 2004).

The case specific methods were expected to give less cycles than the general ones. However, they did not. Both the case-specific methods gave similar results, 2446790 and 2546537 cycles (δ_{onset3} and δ_{onset4} in Tab. 4.3). Lowest number of cycles was obtained by the general method using the ice thickness and structural diameter. Even though the methods using the dimensionless groups were said to be general, they were not truly general here. This is due to the fact that they have used boundary values based on measured data from Norströmsgrund (Palmer et al., 2009; Bjerkås, 2006). The data set used may be too limited to provide a good estimate for the total number of cycles. A large difference in computed number of cycles was also seen for the two methods proposed by Palmer et al. (2009) (methods 2 and 3 in Fig. 4.9).

In any case, the model produced a much higher number of cycles than the observed value, which was 5090 cycles. The reason for this was that 24 hour averages were used. Having fulfilled the onset conditions, the model would calculate the number of cycles for an entire day. The model assumed that the FLC lasted for 80 seconds (Fig. 3.12). After 80 seconds, the FLC stopped, before it started again. This process continued as long as the FLC conditions were satisfied. This is naturally not the case for the full-scale model, but it is a simple approach. This lead to an over-estimation of the number of cycles for each FLC event. With an over-estimated number of cycles, the results were at least conservative. It should also be kept in mind that steady-state vibrations do not arise at all possible conditions (Kärnä, 2001).

5.3 Fatigue life

The resulting fatigue lives ranged from 7 to 4911 years (Tab. 4.3). A fatigue life of only 7 years is very unlikely. A fatigue life this short was calculated assuming continuous cycles the entire season. The stress range was calculated using an idealised cross-section, and only considering the steel. In addition, only fatigue due to tension was checked. Over time, the concrete will lose its integrity due to loading, environmental factors etc. The concrete will most likely experience fatigue due to compressive loads as well.

Surface cracks are either prefabricated or likely to arise in the reinforced steel. Cracks can grow if the loading cycles continue, thus weakening the structure over time. The steel was assumed to not have any initial cracks at all. There are always imperfections and conditions on the steel surface that may give crack growth. Cracks will grow even if the stress level is below yield. The crack growth rate is depending on the stress conditions on the crack tip. Tensile stresses will open the crack, while compressive loading may force the surface at the crack opening together. There is an interaction between the current stress and the preceding stress history (stress memory). Stress memory is not accounted for in the Miner-sum, and therefore there are large uncertainties in the fatigue strength calculation (Almar-Næss, 1985).

The environment around the lighthouse is corrosive, hence reducing the fatigue life. A combination of a highly corrosive environment and a cyclic load can lead to a very high crack growth rate (Almar-Næss, 1985). As the SN-curve applied in the model was for steel in dry air (EC2, 2008), the corrosive effects were neglected. Loads due to winds and waves were also neglected, as this was a study on the ice-structure interaction. Accidental loads were not accounted for. Further investigating the fatigue life, these loads must be implemented in the model.

No other published estimations of the lighthouse fatigue life have been found. Therefore it was difficult to judge if the obtained results were reasonable or not. Four of the methods used, gave results in the range of 30 to 70 years. The lighthouse has been operational since the 1970s. With this in mind, the model was for some methods seen to underestimate the fatigue life. A fatigue life of 4911 years, which was found using the observed number of cycles, may seem long. Fatigue due to cyclic loads acting on the steel ring was considered. Only checking for this kind of fatigue, a fatigue life this long is not unreasonable. This is based on the field-observed number of cycles. The winter of 2002/2003 was seen to be the most severe during the LOLEIF/STRICE campaigns (Bjerkås et al., 2013). With a lower number of cycles than $\sim$ 5000 for most seasons, the annual addition to fatigue damage will be relatively low.

5.4 Comparison of SDOF model and full-scale data

Many attempts have been made to establish models of the ice-structure interaction. Early modelling attempts were made by Kärnä and Turunen (1990). Kärnä and Turunens model used only the ice velocity and the fundamental frequency to find the vibration amplitude, i.e. a simple model using only the lowest natural mode. Kärnä (1992) developed the model further and modelled the soil-structure-ice interaction. Models have also been established in ABAQUS. Newer attempts to develop more sophisticated models have been made, e.g. using ABAQUS/CAE (Määttänen, 2001), including many vibration modes. Often, SDOF models have been used in comparison with more complex models.

The main purpose of using a simple SDOF model was to make a quick assessment of the lighthouse response. As mentioned above, several researchers have used SDOF models. Comparing the amplitude levels in Fig. 4.14, it was seen that the simple model used in this thesis was able to give a fair representation of the amplitude level. It should be kept in mind that the model was unconditionally stable.

Finding input values for the SDOF model was not easy. Values were reported by different researchers. The best SDOF values for the response at the MWL, were obtained from articles by Guo (2012) and Kärnä et al. (2004). Small variations in input values produce deviations. Only two sets of input values were used in the present work. To find a trend in the results, more input values should have been implemented.

The acceleration amplitudes of the model were in Fig. 4.14 seen to be smaller than the measured ones for the 30^{th} March 2003 event. One possible reason for this may be that he accelerometer was placed at 19.65 m above seabed in-situ, while the model calculated the response at 14.2 m. Lacking a modal mass at 19.65 was the reason for not calculating the model response here. Scaling the q-factor (Tab. 4.4) was seen to affect the model significantly. This factor directly changed the load amplitude (Eq. 3.11). Defined range for this factor was 0.1 - 0.5 by ISO19906 (2010).

The mean force level of the calculated and measured force was seen to differ. Having calculated the global force by using Eq. 3.8, produced higher values than measured in-situ. Some of the load panels were reported to be damaged and not recording the acting forces (Bjerkås et al., 2013). These panels covered the lighthouse from $0\,^{\circ}$ to $167\,^{\circ}$, while the calculated pressure used the entire waterline diameter. Random force fluctuations were seen in-situ, but the model did not include any random force level variation.

6 Conclusion

Having investigated onset conditions for FLC and the fatigue life of the lighthouse using a simple model, the following key conclusions were made:

- A straightforward model has been developed to predict onset conditions for FLC and steady-state vibrations. Estimated number of cycles ranged from 541008 to 40950144, depending on the approach used. The model overestimated the number of cycles compared to the observed number, which was 5090.

- The estimated fatigue life of the lighthouse was seen to vary greatly, depending on the approach. The extreme case gave a fatigue life of 7 years, while using the observed number of cycles yielded a fatigue life of 4911 years. A fatigue life of 7 years is highly unlikely.

- Of the five proposed methods to find possible FLC conditions, four of the methods yielded a fatigue life ranging from 31 to 70 years. The fifth method gave a fatigue life of 239 years. As the lighthouse has been operational for over 40 years, the model was seen to underestimate the fatigue life. Models used in design should always overestimate damages.

- The lighthouse was seen to be susceptible to frequency locked-in crushing.

- Calculated ice thickness was seen to be in range with the field-observed thickness. A calculated maximum of 0.81 m and a observed maximum of 0.9 m.

- Response of the SDOF model was close to the measured response of the 30^{th} March 2003 event. Simple SDOF models can be utilized when performing a first evaluation of the structural response.

6.1 Further work

To develop a more sophisticated model, the following steps could be made:

- Perform a more detailed cross-section analysis.
- Check the fatigue of the concrete due to compression.
- Utilize a Weibull distribution when estimating the fatigue life.
- Account for accidental and occasional loads.
- Verify the model using other, similar structures. The other lighthouses located in the Gulf of Bothnia are candidates.
- Collect more data and use the model for different seasons.
- Establish a FEM model of the lighthouse, or use existing models.
- Compare SDOF/FEM models with several known cases of FLC and resonant-like vibrations.
- Include higher-order eigenfrequencies when calculating the response.

Bibliography

Almar-Næss, A. (1985), *Fatigue Handbook - Offshore Steel Structures*, Tapir. ISBN 82-519-0662-8.

Bjerkås, M. (2006), Ice Actions on Offshore Structures, PhD thesis, Norwegian University of Science and Technology (NTNU), Trondheim.

Bjerkås, M. (2007), Review of measured full scale ice loads to fixed structures, *in* '26th International Conference on Offshore Mechanics and Arctic Engineering'.

Bjerkås, M. (2013). Personal communication.

Bjerkås, M., Lønøy, C. and Gürtner, A. (2013), 'Ice-induced vibrations and effects of ice temperature', *International Journal of Offshore and Polar Engineering* **1**(23), 1–6.

Bjerkås, M. and Skiple, A. (2005), 'Occurrence of continuous and intermitten crushing during ice-structure interaction', *Proc. 18th Conference on Port and Ocean Engineering under Arctic Conditions (POAC)* pp. 1131–1140.

Blenkarn, K. A. (1970), 'Measurement and analysis of ice forces on cook inlet structures', *Proceedings of the 2nd Offshore Technology Conference* .

Chopra, A. (2007), *Dynamics of Structures - Theory and Applications to Earthquake Engineering*, New Jersey: Person Prentice Hall.

EC2 (2008), *NS-EN 1992-1-1:204+NA:2008. Design of Concrete Structures. Part 1-1: General rules and rules for buildings.*

Ekman, V. (1902), 'Om jordrotationens inverkan på vindstrommar i hafvet', *Nyt magasin f. Naturv. B. 40* .

Engelbrektson, A. (1977), 'Dynamic ice loads on a lighthouse structure', *Port and Ocean engineering under Arctic Conditions 1977 (POAC77)* .

Engelbrektson, A. (1983), 'Observations of a resonance vibrating lighthouse structure in moving ice', *Port and Ocean engineering under Arctic Conditions 1983 (POAC83)* .

Engelbrektson, A. and Janson, J. E. (1985), 'Field observations of ice action on concrete structures in the baltic sea', *Concrete International* **7**(8), 48–52.

Erntsons, E. and Kjellgren, G. (1969), Estimation of ice pressure and the bending moment at rupture of the light-house tower nygrån., Technical report, Swedish Board of Shipping and Navigation.

Guo, F. (2012), 'Reanalysis of ice induced steady state vibration from an engineering perspective', *21st IAHR International Symposium on Ice, Dalian, China* .

ISO19906 (2010), 'International standard, petroleum and natural gas industries – arctic offshore structures'.

Kärnä, T. (1992), A procedure for dynamic soil-structure-ice interaction, *in* 'Proceedings of the Second International Offshore and Polar Engineering Conference', San Francisco, California, USA.

Kärnä, T. (1994), 'Mitigation of steady-state vibrations induced by ice', *Proc. of the Fourth (1994) International Offshore and Polar Engineering Conference* .

Kärnä, T. (2001), Simplified modelling of ice-induced vibrations of offshore structures, *in* 'Proceedings of the 16th International Symposium on Okhotsk Sea and Sea Ice', pp. 114–122.

Kärnä, T., Heinoen, J. and Luo, C. (2004), Dynamic behaviour of the norströmsgrund lighthouse, Technical report, VTT Technical Research Centre of Finland. Revision No. 1.0 Final.

Kärnä, T. and Jochmann, P. (2003), 'Field observations on ice failure modes', *Proceedings of the 17th International Conference on Port and Ocean Engineering under Arctic Conditions (POAC)* .

Kärnä, T. and Qu, Y. (2005), Analysis of the size effect in ice crushing – edition 1, Technical report, VTT Technical Research Centre of Finland. Internal Report.

Kärnä, T. and Turunen, R. (1990), 'A straightforward technique for analysing structural response to dynamic ice action', *Proc. of the 9th OMAE Conference, Houston* **IV**.

Lehmann, T. (2010), Prediction of the response from ice forces to a lighthouse structure, reanalysis of the nygrån lighthouse collapse incident in the light of new design codes, Master's thesis, Norwegian University of Technology and Science.

Määttänen, M. (2001), 'Numerical simulation of ice-induced vibrations in offshore structures', *The 14th Nordic Seminar on Computational Mechanics* .

Nansen, F. (1902), 'The oceanography of the north polar basin'.

Palmer, A., Yue, Q. and Guo, F. (2009), 'Ice induced vibrations and scaling', *Cold Regions Science and Technology* .

Sarpkaya, T. and Isaacson, M. (1981), 'Mechanics of wave forces on offshore structures'.

Sodhi, D. (1988), Ice-induced vibrations of structures, *in* 'IAHR Ice Symposium'.

Sodhi, D. (1991), 'Ice-structure interaction during indentation tests. in ice-structure interaction', *Proc. of IUTAM-IAHR Symposium* pp. 619–640.

Sørensen, S. I. (2009), *Betong-konstruksjoner - Beregning og dimensjonering etter Eurocode 2*, Tapir Akademisk Forlag. ISBN 978-82-519-2539-6.

Stefan, J. (1981), 'Ueber die theorie der eisbildung, insbesondere uber die eisbildung im polarmeere.', *Annalen der Physik* .

Timco, G. and Weeks, W. (2010), 'A review of the engineering properties of sea ice', *Cold Regions Science and Technology* **60**, 107–129.

Williams, J., Higgonson, J. and Rohrbough, J. (1968), 'Sea and air: The naval environment'.

Zubov, N. (1943), Artic ice. technical report, Technical report, Translated by U.S. Naval Oceanographic Office and American Meterological Society in 1952.

A Appendix

A.1 Figures

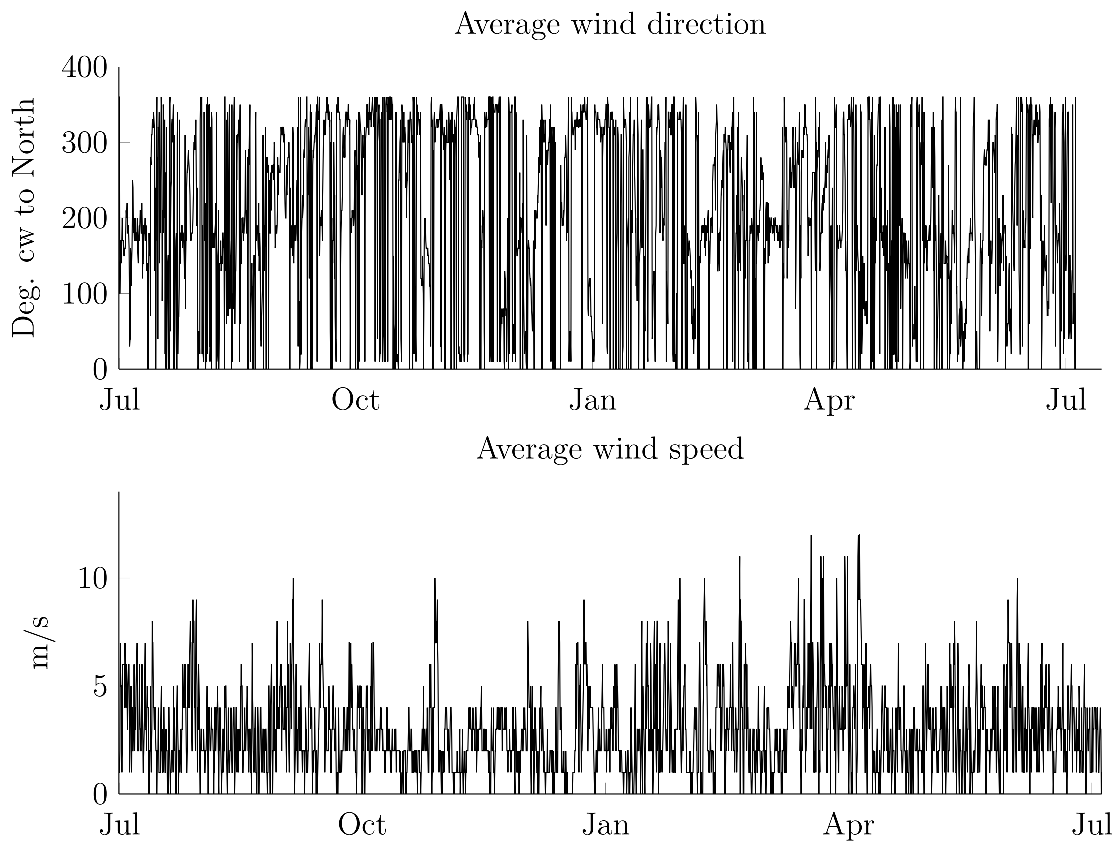


Figure A.1: Measured wind drift directions at Luleå Airport by SMHI.

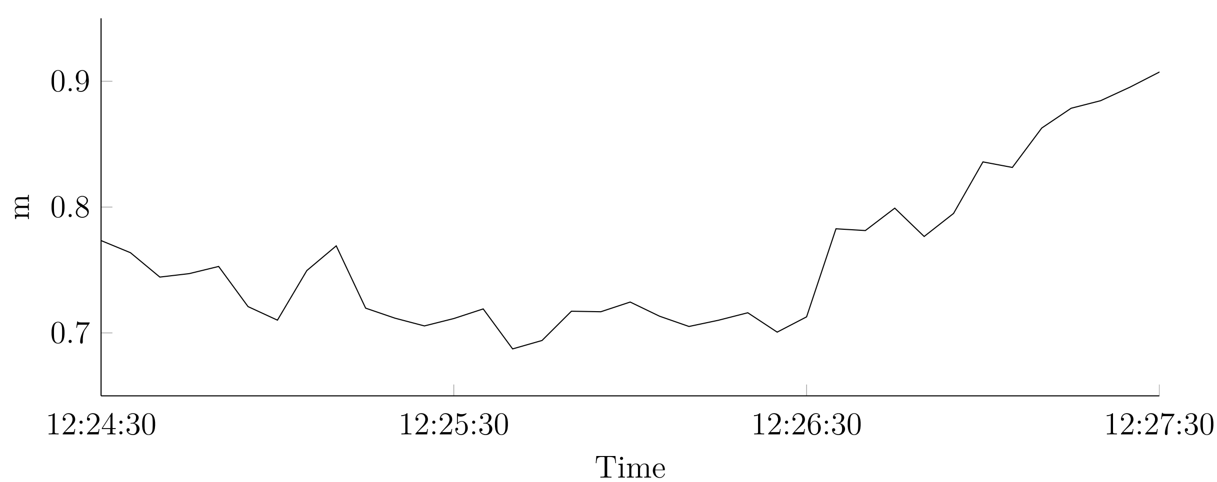


Figure A.2: Measured ice thickness for the 30.03.2003 event.

A.2 Tables

Table A.1: Weight of lighthouse acting on the considered cross-section.

$H_{section}$ (m)	r_{outer} (m)	$t_{section}$ (m)	$V_{section}$ (m^3)	$\rho_{section}$ (kg/m^3)	$M_{section}$ (kg)
9.00	3.60	0.50	94.72	2500	236798
3.00	2.65	0.55	24.62	2650	65249
0.20	2.10	2.10	2.77	2500	6927
3.00	2.50	0.40	17.34	2650	45955
0.02	2.10	2.10	0.28	2500	693
3.05	2.50	0.40	17.63	2650	46721
0.02	3.60	3.60	0.81	2500	2036
2.80	2.40	0.30	11.88	2650	31469
0.02	2.10	2.10	0.28	2500	693
2.80	2.40	0.30	11.88	2650	31469
0.02	3.60	3.60	0.81	2500	2036
2.70	2.30	0.20	7.46	2650	19781
0.02	2.10	2.10	0.28	2500	693
2.70	2.30	0.20	7.46	2650	19781
0.02	2.10	2.10	0.28	2500	693
2.70	2.30	0.20	7.46	2650	19781
0.02	3.00	3.00	0.57	2500	1414
2.70	2.30	0.20	7.46	2650	19781
0.10	4.30	4.30	5.81	2500	14522
				Summed mass (kg)	566490
				Summed weight (N)	5557264

A.3 Distribution of rebars

Having found the steel area and the number of rebars, it was checked that the distribution of rebars was satisfied using EC2 (2008). Having found the nr. of rebars to be $n_{rebars} = 191$. The concrete must cover $c_{nom} = 35$ mm. Minimum horizontal distance between bars is $a_h = 50$ mm. Fig. A.3 shows an illustration of the rebar distribution, assuming two layers.

Each bar required 75 mm space. Layer 1 had a distance of 8208 mm, while layer 2 had a distance of 768 mm. Total distance to distribute rebars over was then 15896 mm. Having 191 rebars which each require 75 mm, the required minimum distance to distribute rebars over was $191 \cdot 75$ mm $= 13370$ mm. The required miminum distance between rebars in the same layer, and distance between layers was seen to be satisfied according to EC2 (2008). The steel rebars were replaced by a steel ring. This steel ring had the same cross-sectional area as the total cross-sectional area of the reinforcing bars.

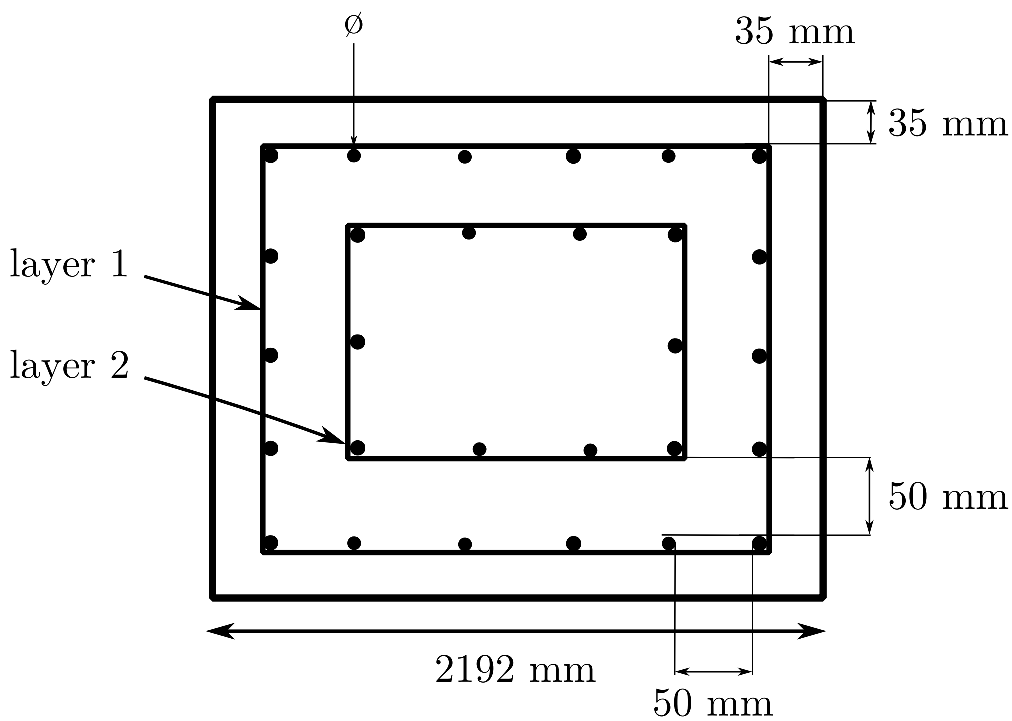
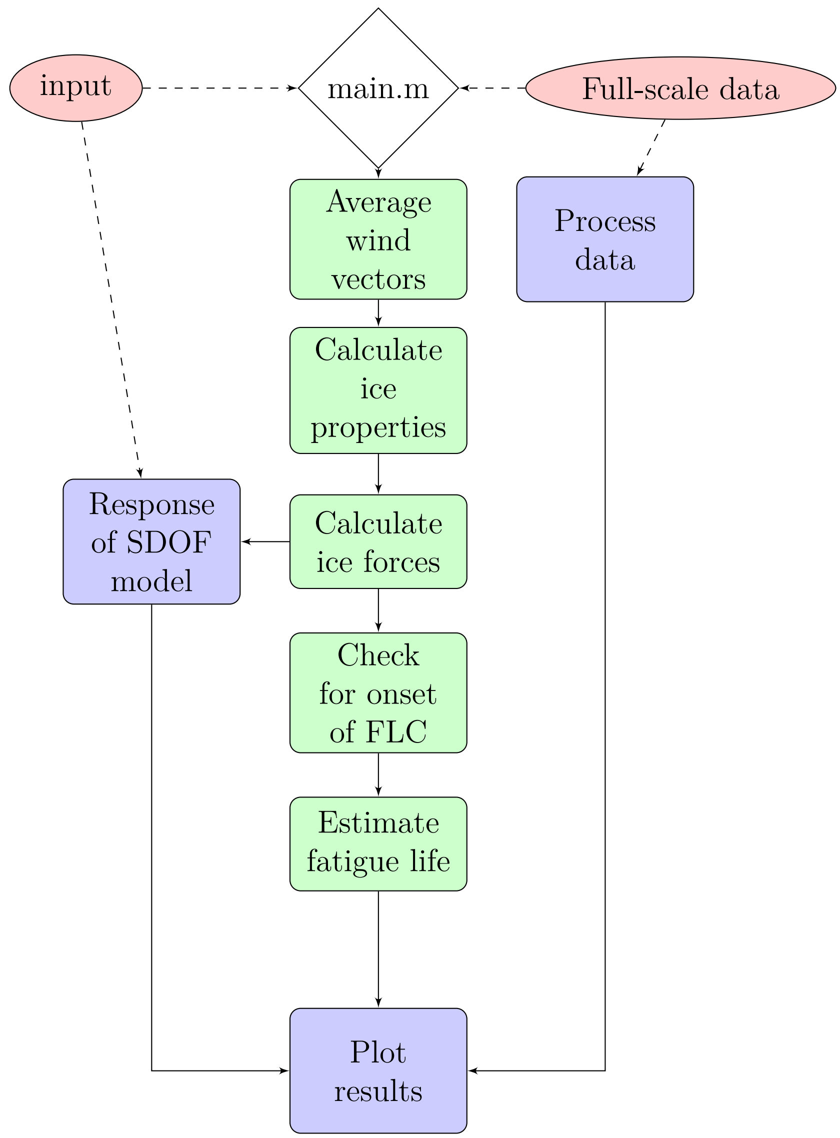


Figure A.3: Illustration of rebar distribution.

B MATLAB scripts

Below is a flow chart showing the structure of the MATLAB-program used in this thesis.



B.1 main.m

```matlab
% Author: Andreas Meese, Trondheim 21.02.2013
% 'Winter is coming'
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Script description:
% find number of cycles for a season given input parameters
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
tic
clc
clear all
close all
%% location of input file and subfunctions
% location of program (inputfile must be in same path as main.m)
folder = 'C:\Users\Andreas\Dropbox\NTNU\Master\Program\';
% location of outputfolder
outputloc = ...
    'C:\Users\Andreas\Dropbox\NTNU\Master\Report\latex\figures\Results';
% Make folders with following names in the output folder:
opfolders = ['\ice';'\ons';'\vib';'\frc';'\dyn';'\cmp'];
% name of inputfiles
inputfile = 'input.txt';
data1 = 'A_0203';
data2 = '03_3003_0400_122500_122800';
addpath(strcat(folder,'input'))
addpath(strcat(folder,'subfunctions'))  %related to general input
addpath(strcat(folder,'subfunctions2')) %related to the 30.03 event
addpath(strcat(folder,'plotfunctions')) %location of plotting functions

resolution = '-r600'; %resolution on plots if using print as function

%% Read input
% read input and the Luleaa data
input = rinput(outputloc,opfolders,inputfile,data1);
% load full scale data 30.03.03 event
event = loadevent(data2);

%% Wind statistics
% a 24 hr avg of the wind direction is to be used further to find ice
% drift
windprop = wind(input);

%% Calculate parameters based on ISO etc
% Calculate ice growth, ice velocity,drift direction and ice temperature
iceprop = iceproperties(input,windprop);

% Calculate global ffatiorce on a vertical structure
force = verticalforces(input,iceprop);

%% Find onset criteria for FLC and number of vibrations
FLC = onset(input,iceprop,force);
```

```
%% Estimate fatigue damage and fatigue life
fatigue = fatiguecalc(input,iceprop,force,FLC);

%% Dynamic response to FLC
dynamic2 = dynresponse2(input,event);

%% Process the 30.03.2003 event
flcevent = processevent(input,iceprop,event);


%% Plots
windplots(input,windprop,iceprop)
iceplots(input,iceprop,windprop);
onsetplot(input,iceprop,FLC)
vibrationplot(input,iceprop,FLC);
fatigueplot(input,fatigue);
forceplots(input,iceprop,force);
dynamicplots(input,iceprop,FLC,force,dynamic2)
comparisonplots(input,dynamic2,flcevent)

toc
```

B.2 input.txt

```
7.2
2.6 2.89 2.24 2.2
170000 2.6 0.22
171951 2.89 0.12123 3.88E09
0.02 0.04
42.3
14.2
1.0 0.1
140
0 3.50 7.0 11.75 14.18 16.5 19.65 22.8 25.85 28.95 31.5 34.3 37.1 39.55 42.3
0 0.05 0.1 0.17 0.22 0.26 0.32 0.39 0.47 0.56 0.64 0.73 0.82 0.91 1.0
1.72E09
0.001
5557264

3.6
0.5

2
5 9
162.5 1000000
100000
500
5090 0.9

0.020 30
```

```
300 45

0.0 1025.0
1.8

0.015 0.025
0.00095 0.0035
80

80 300
8
12

40 0.0119
0.1 0.5
0.06
0.25 0.5
0.001 0.0
90 130
2176 2183
```

B.3 rinput.m

```
% Subprogram for reading the input file the program for icemain.m
% Author: Andreas Meese, Trondheim 21.02.2012
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function[input] = rinput(outputloc,folders,filename,data1)

%open the input file
fileid=fopen(filename,'r');

%% Full scale measured environmental parameters
load(data1)
input.fs.temp =  (A_0203(:,5));              %measured air temperature
input.fs.windd = (A_0203(:,3));              %measured wind direction
input.fs.winds = (A_0203(:,4));              %measured wind speed

%% Structural properties
input.str.diam = fscanf(fileid,'%f',1);      %waterline diameter in meters
input.str.freq = fscanf(fileid,'%f',4);      %1st. translational freq
input.str.guo = fscanf(fileid,'%f',3);       %nodal mass (guo) and freq wl
input.str.strice = fscanf(fileid,'%f',4);    %SDOF M (kg), freq and K  (N/m)
input.str.xi1 = fscanf(fileid,'%f',2);       %SDOF damping% of critical
input.str.height = fscanf(fileid,'%f',1);    %tower height
input.str.wl = fscanf(fileid,'%f',1);        %waterline level (m)
input.str.Acontact = fscanf(fileid,'%f',2); %Reference A for pressure (m2)
input.str.Aface = fscanf(fileid,'%f',1);     %face area of structure
input.str.modeshape1 = fscanf(fileid,'%f %f',[15 2]);  %modeshape of sdof
input.str.kwaterline = fscanf(fileid,'%f',1); %waterline stiffness, strice
input.str.dwlbond = fscanf(fileid,'%f',1); %static displ. boundary FLC (m)
```

```
input.str.weightabovecs = fscanf(fileid,'%f',1); %weight above cs cons.

%% Parameters related to reinforced concrete and the steel
input.concrete.r_outer = fscanf(fileid,'%f',1); %outer diameter (m)
input.concrete.thick = fscanf(fileid,'%f',1);   %concrete thickness (m)

%% Parameters related to fatigue life
input.fat.designload=fscanf(fileid,'%f',1); %design load for waterline MN/m
input.fat.SN = fscanf(fileid,'%f',2); %k1 and k2 for ribbed steel
input.fat.klimitsigma = fscanf(fileid,'%f',2); %limit betw. k1 and k2. and
% limit between low and high-cycle range
input.fat.highlow = fscanf(fileid,'%f',1); %limit betw. high and low cycle
input.fat.rsyieldlim = fscanf(fileid,'%f',1); %yield for ribbed steel
% observed nr. of cycles and ice thickness at the time
input.fat.obsyclethick = fscanf(fileid,'%f',2); %nr. and m

%% Wind drift speed and direction
%1 is coefficient between ice and wind, 2 is geostrophic relation
input.drift.geostrophic = fscanf(fileid,'%f',2);
%upper and lower boundary for the ice drifts ability to move
input.drift.boundaries = fscanf(fileid,'%f',2);
%
%% Physical parameters
input.phys.water = fscanf(fileid,'%f',2);     %1 is t_freeze, 2 is density
input.phys.Cr = fscanf(fileid,'%f',1);        %ice strength coefficient
%
%% Input parameters for FLC criteria
%upper and lower dimensionless boundary for thickness
input.flc.Tdim = fscanf(fileid,'%f',2);
%upper and lower dimensionless boundary for diameter
input.flc.Ddim = fscanf(fileid,'%f',2);
%max lock in duration per event
input.flc.dur = fscanf(fileid,'%f',1);
%
%% processing properties
input.proc.length = fscanf(fileid,'%f',2);   %length of winter in days
input.proc.fmeas = fscanf(fileid,'%f',1);    %measurements per day
input.proc.ilength = [input.proc.length(1)*input.proc.fmeas ...
input.proc.length(2)*input.proc.fmeas];      %season length in indexes
input.fontsize = fscanf(fileid,'%f',1);      %fontsize on plots
%
%% Simulation properties for dynamic response to FLI
input.sim.time = fscanf(fileid,'%f',2);   %time length and time step
input.sim.q = fscanf(fileid,'%f',2);   %upper and lower parameter for Fmin
input.sim.gamma = fscanf(fileid,'%f',1);%ice velocity to response velocity
input.sim.newmark = fscanf(fileid,'%f',2);%beta and gamma in Newmarks met.
input.sim.initial = fscanf(fileid,'%f',2);%initial displ. and velocity
input.steadyevent = fscanf(fileid,'%f',2);%start and stop of steadystate
input.sevent = fscanf(fileid,'%f',2);     %start and stop for ss 30.03
%
%define the output paths
input.outputloc = strcat(outputloc,folders);

end
```

B.4 loadevent.m

```
% Subprogram for reading the input file the program for icemain.m
% Loading fullscale data from the 30.03.2003 event 12:24:30
% Author: Andreas Meese, Trondheim 21.02.2012
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function[full] = loadevent(data2)

%% Load environmental parameters
load(data2)

%% Load measured structural parameters 30.03.2003
% Event time information
full.event.start = SubEvent.Load_Time(1,:);
full.event.stop = SubEvent.Load_Time(end,:);
full.event.length = SubEvent.Load_Time(end,:)-SubEvent.Load_Time(1,:);

% Make timevector
n = length(SubEvent.Load_Time); %number of measurements
full.event.seconds = full.event.length(1)*60*60+ ...
    full.event.length(2)*60+full.event.length(3); %time in seconds
% Time vector
time0=(0:(full.event.seconds/(n)):full.event.seconds)';
full.time=time0(1:end-1);

% Accelerations
full.acc.lev19 = SubEvent.Acc(:,3); %sum of accelerations at 19.65 m
full.acc.lev37 = SubEvent.Acc(:,6); %sum of accelerations at 37.10 m

% Forces
full.totalforce.x = SubEvent.Total(:,3);
full.totalforce.y = SubEvent.Total(:,6);
full.totalforce.sum = SubEvent.Total(:,7);

% Inclination
full.inclination = SubEvent.Inc(:,1:6);

% Load panel forces
full.panelforce = SubEvent.Load_Pan(:,:);

% Load segment
full.loadseg.left = SubEvent.Load_Seg.L(:,1:4);
full.loadseg.right = SubEvent.Load_Seg.R(:,1:4);

%%% Ice thickness
%% thickness time
full.thickness.time = SubEvent.Thick_Time; %hours minutes seconds

% construct time vector for thickness
m = length(full.thickness.time); %number of measurements
```

```
full.thickness.tlength=full.thickness.time(end,:)...
    -full.thickness.time(1,:); %length in HH:MM:SS
full.tseconds = full.thickness.tlength(1)*60*60+...
    full.thickness.tlength(2)*60+full.thickness.tlength(3); %seconds
full.thickness.timevector=(0:(full.tseconds/(m)):full.tseconds)';
full.thickness.timevector=full.thickness.timevector(1:end-1);
%% Measured ice sheet thickness
% Underside of ice sheet layer by electro-magnetic measuring.
full.thickness.em = SubEvent.Thick_EM;
% Thickness by upward looking sonar
full.thickness.son = SubEvent.Thick_SON;
% Ice sheet thickness
full.thickness.t = SubEvent.Thickness;
% Top side of ice sheet layer (vertical pos)
full.thickness.las = SubEvent.Thick_LAS;
% Thickness by laser and em device
full.thickness.diff = SubEvent.Thick_LAS-SubEvent.Thick_EM;

%% Environmental parameters
full.env.time = SubEvent.Env_Time(:,1:3); %hours minutes seconds
full.env.temp = SubEvent.Ta(:,1:3); %

% Wind parameters
full.wind.speed = SubEvent.W.Speed(:,1:3);
full.wind.dir = SubEvent.W.Dir(:,1:3);

end
```

B.5 wind.m

```
% Subprogram for for main.m. Finds the 24 hrs averages of wind drift
% direction and velocities.
% Author: Andreas Meese, Trondheim 29.5.2012
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function[result] = wind(input)

P=8;    % 4 gives 2 avg per day, 8 gives 1 avg per day
L = length(input.fs.windd)/P;
% measured components
theta_w = circ_ang2rad(input.fs.windd);
v_wind = input.fs.winds;

result.avgwindcomp = zeros(2,L);  %daily avg. of cos and sin components
result.avgwind = zeros(L,2);      %col 1 is avg.speed, col 2 is avg.dir

% convert wind directions to radians

for j = 1:L
    if j == 1
        a = 1;
```

```
        b = P;
        sum1=0;
        sum2=0;
        for i = 1:P
            sum1 = sum1 + v_wind(i)*cos(theta_w(i)); %y component
            sum2 = sum2 + v_wind(i)*sin(theta_w(i)); %x component
        end
        result.avgwindcomp(1,j) = sum1/P; %cosine component of vector (y)
        result.avgwindcomp(2,j) = sum2/P; %sine component of vector (x)
    else
        a = a+P;
        b = b+P;
        sum1=0;
        sum2=0;
        for i = a:b
            sum1 = sum1 + v_wind(i)*cos(theta_w(i));
            sum2 = sum2 + v_wind(i)*sin(theta_w(i));
        end
        result.avgwindcomp(1,j) = sum1/P;
        result.avgwindcomp(2,j) = sum2/P;
    end
        result.avgwind(j,1) = sqrt((result.avgwindcomp(1,j)^2) + ...
        (result.avgwindcomp(2,j)^2));
    %std deviation of the 8 vectors
    result.windstdv(j) = std(input.fs.windd(a:b));
end

% find the average direction .
for j = 1:length(result.avgwind)
    if result.avgwindcomp(2,j) == 0 && result.avgwindcomp(1,j) == 0
        result.avgwind(j,2) = 0;
    elseif result.avgwindcomp(1,j) < 0
        result.avgwind(j,2) = ...
        atan(result.avgwindcomp(2,j)/result.avgwindcomp(1,j)) + pi();
    elseif result.avgwindcomp(2,j) < 0 && result.avgwindcomp(1,j) > 0
    %avg wind direction
    result.avgwind(j,2) = ...
        atan(result.avgwindcomp(2,j)/result.avgwindcomp(1,j)) + 2*pi();
    %avg wind speed
    end
end
result.alpha_rad = result.avgwind(:,2);
% mean angle
result.alpha_bar = circ_mean(result.alpha_rad);
% median
result.alpha_hat = circ_median(result.alpha_rad);
% std. deviation
% s is angular std
% s0 is circular std
[result.s_alpha,result.s0_alpha] = circ_std(result.alpha_rad);
%{
for j = 1:L/8
    if j == 1
        a = 1;
```

```
        b = 8;
        sum1=0;
        sum2=0;
        for i = 1:8
            sum1 = sum1 + v_wind(i)*cos(theta_w(i)); %y component
            sum2 = sum2 + v_wind(i)*sin(theta_w(i)); %x component
        end
        result.avgwindcomp(1,j) = sum1/8; %cosine component of vector (y)
        result.avgwindcomp(2,j) = sum2/8; %sine component of vector (x)
    else
        a = a+8;
        b = b+8;
        sum1=0;
        sum2=0;
        for i = a:b
            sum1 = sum1 + v_wind(i)*cos(theta_w(i));
            sum2 = sum2 + v_wind(i)*sin(theta_w(i));
        end
        result.avgwindcomp(1,j) = sum1/8;
        result.avgwindcomp(2,j) = sum2/8;
    end
        result.avgwind(j,1) = sqrt((result.avgwindcomp(1,j)^2) + ...
        (result.avgwindcomp(2,j)^2));
    %std deviation of the 8 vectors
    result.windstdv(j) = std(input.fs.windd(a:b));
end
%}
end
```

B.6 iceproperties.m

```
% Calculate ice thickness growth, ice velocity and drift direction
% for input temperature and for measured
% temperature. Subprogram to program icemain.m
% Author: Andreas Meese, Bergen 28.02.2012
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function[result] = iceproperties(input,windprop)

% INPUTS
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
T = input.fs.temp;           % Ambient air temperature
% Relation between wind speed and ice drift speed
IW_coeff = input.drift.geostrophic(1);
% Geostrophic relation between wind and ice
theta_g = input.drift.geostrophic(2)*pi()/180;
f_water = input.phys.water(1);    % freezing temperature of water
D = input.str.diam;               % structural width
f_meas = input.proc.fmeas;        % measuring frequency
P = 8;                            % 2 avg pr day, 8 gives 1 avg pr day

```

```
% OUTPUTS
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% Ice thickness and FDD
    L = length(T); % size of the measured temperature vector
    result.thick = zeros(1,L);
    result.FDD = zeros(1,L);

    for i = 1:L
        if i==1
            % only using temperatures below zero
            if T(i) ≤ f_water
            result.FDD(i) = (1/f_meas)*(f_water-T(i));

            else
            result.FDD(i) = 0;

            end

        else
            if T(i) ≤ f_water
            result.FDD(i) = result.FDD(i-1) + (1/f_meas)*(f_water-T(i));

            else
            result.FDD(i) = result.FDD(i-1) + 0;

            end
        end

    end

    for j = 1:input.proc.ilength(1)
        result.FDD(i) = 0;
    end
    for j = input.proc.ilength(2):length(result.FDD(i))
        result.FDD(i) = 0;
    end

    for i = 1:L

        if i==1
            if result.FDD(i) == 0
            result.thick(i) = 0;
            else
            a=1;
            b=-50;
            c=-8*result.FDD(i);
            %zubovs formula (centimeters)
            result.thick(i) = (-b-sqrt((b^2)-4*a*c))/(200*a);
            end

        else
            if result.FDD(i) == 0
            result.thick(i) = result.thick(i-1) + 0;
            else
```

```
                a=1;
                b=-50;
                c=-8*result.FDD(i);
                %zubovs formula (centimeters)
                result.thick(i) = (-b-sqrt((b^2)-4*a*c))/(200*a);
                end
            end
        end
        result.thick = (-1).*result.thick;

%% Ice speed and drift direction
% Calculated 24hr average of the wind -> 365 days w/ 8 measurings pr day
% convert to radians since matlab operates with radians
% the ice drift vector; row 1 is speed, row 2 is direction in degrees
result.icedrift = zeros(2,L/8);
for j = 1:L/P
    result.icedrift(1,j) = windprop.avgwind(j,1)*IW_coeff;
    %converting from deg to rad
    result.icedrift(2,j) = windprop.avgwind(j,2) + theta_g;
end

%% Define the start and end of the winter season
        for j = 1:input.proc.ilength(1)
        result.FDD(j)=0;
        result.thick(j)=0;
       % result.pg(j) = 0;
        result.aspectratio(j) = 0;
        %result.icedrift(:,j) = 0;
        end
        for j = input.proc.ilength(2):length(result.thick)
        result.FDD(j)=0;
        result.thick(j)=0;
       % result.pg(j) = 0;
        result.aspectratio(j) = 0;
        %result.icedrift(:,j) = 0;
        end

% Daily average ice thickness and air temp
result.ticeavg = zeros(1,L/P);
result.tempavg = zeros(1,L/P);
for j = 1:L/P
    if j == 1
        a = 1;
        b = P;
        sum1=0;
        sum2=0;
        for i = a:b
            sum1 = sum1 + result.thick(i);
            sum2 = sum2 + input.fs.temp(i);
        end
        result.ticeavg(1,j) = sum1/P;
        result.tempavg(1,j) = sum2/P;
    else
        a = a+P;
```

```
        b = b+P;
        sum1=0;
        sum2=0;
        for i = a:b
            sum1 = sum1 + result.thick(i);
            sum2 = sum2 + input.fs.temp(i);
        end
        result.ticeavg(1,j) = sum1/P;
        result.tempavg(1,j) = sum2/P;
    end
    % no ice thickness -> no drift velocity
    if result.ticeavg(1,j) == 0
       result.icedrift(:,j) = 0;
    end
end

%% Global pressure , daily avg.
    result.pg = zeros(1,L/P);
    result.aspectratio = zeros(1,L/P);
    m = -0.16;                %empirical coefficient
    h1 = 1.0;                 %reference thickness
    for j = 1:L/P
       if result.ticeavg(j) == 0;
           result.pg(j) = 0;
       else
           result.aspectratio(j) = input.str.diam/result.ticeavg(j);
           if result.aspectratio(j) > 40
               result.pg(j) = 0;
           else
               if result.ticeavg(j) < 1.0
                  n = -0.50 + result.ticeavg(j)/5;
               elseif result.ticeavg(j) ≥ 1.0
                  n = -0.30;
               end
               result.pg(j) = input.phys.Cr*((result.ticeavg(j)/h1)^n)*...
               ((input.str.diam/result.ticeavg(j))^m);
           end
       end
    end
end
```

B.7 verticalforce.m

```
% Calculate force on structure given structural properties, ice properties.
% Subprogram to program icemain.m
% Author: Andreas Meese, Bergen 28.02.2012
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function[result] = verticalforces(input,ice)

% INPUTS
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% assume biaxial stiffness
% Structural properties
D = input.str.diam;              %structural diameter (cylinder) (meters)
h_wl = input.str.wl;             %waterline height above seabed
A1 = input.str.Acontact(1);      % contact area 1 (m2)
A2 = input.str.Acontact(2);      % contact area 2 (m2)
t_ice = ice.ticeavg;               % Ice thickness. vector [1:2920]

%waterline stiffness of norstromsgrund by STRICE-report (karna, 2004)
k = input.str.kwaterline;

%length of vector
L = length(t_ice);

% OUPUTS
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%% ISO 19906 Vertical structure; page 169 equation A.8-21
% Nominal contact area ( structural width * ice thickness
An = D.*t_ice;
result.FGlobmax = zeros(1,L);
% Global ice action
    for j = 1:L
       result.FGlobmax(j) = ice.pg(j)*An(j);
    end

%%% Displacement of the ligthhouse due to static loading
%result.∆wl = zeros(1,L);

for j = 1:L
    result.∆wl(j) = (10^6).*result.FGlobmax(j)/k; %MN to N
end

% Over Turning Moment (Nm) and Base shear N
result.OTM = zeros(1,L);
result.BS = zeros(1,L);
for j = 1:L
    if j≥input.proc.length(1) && j ≤ input.proc.length(2)
        result.OTM(j) = result.FGlobmax(j)*h_wl;
        result.BS(j) = result.FGlobmax(j);
```

```
    end
end
end
```

B.8 onset.m

```
% Find nr. of events with FLC using onset criteria, structural properties,
% ice properties.
% Subprogram to program icemain.m
% Author: Andreas Meese, Bergen 28.02.2012
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function[result] = onset(input,iceprop,force)

t_ice = iceprop.ticeavg;            %ice thickness (24 hr avg)
v_ice = iceprop.icedrift(1,:);      %ice velocity 24hrs avg
theta_ice = iceprop.icedrift(2,:);  %ice drift directions 24hrs avg
T = iceprop.tempavg;                %air temp 24 hr avg
L = length(t_ice);                  %vector length 365
P = 8;                              %4 is 2 avg pr day, 8 is 1 avg pr day

% Vectors for different methods. Row 1 is fulfilment at index j (0 or 1),
% row 2 is nr. of vibrations at index j, row 3 is accumulated vibrations
% after index j
result.method1 = zeros(3,L);  %Dimensionless group by Palmer diameter
result.method2 = zeros(3,L);  %Dimensionless group by Palmer thickness
result.method3 = zeros(3,L);  %Bjerkas and Skiple method specific thickness
result.method4 = zeros(3,L);  %Bjerkas and Skiple method specific temp.
result.method5 = zeros(3,L);  %ISO onset criteria/dynamic instability
result.iceaction = zeros(1,L);%vector w/ calculated instability-value

%% PARAMETERS RELATED TO FLC-BREAKS DUE TO LOST CONTACT STRUCTURE/ICE
Tevent = (24*P/input.proc.fmeas)*3600;            %3 hour measurement freq
Vibr_1lockin = input.flc.dur*input.str.freq(2);   %nr of vibrations/lockin

%% PARAMETERS RELATED TO DYNAMIC STABILITY CRIETRION BY ISO
theta = 40*10^6;                    %kg/ms parameter ISO
xeta1 = input.str.xi1(1);           %damping ratio 0.02
phi1 = input.str.modeshape1(5,2);   %modal value from guo
M1 = input.str.guo(1);              %Modal mass guo
f1 = input.str.guo(2);              %1st natural freq. guo
phi2 = input.str.strice(3);         %modal value from strice
M2 = input.str.strice(1);           % mass from strice
f2 = input.str.freq(2);             %1st natural strice
xeta2 = input.str.xi1(2);           %damping ratio 0.04 (upper bound)

%% ICE action iso 19906 criterion
result.iceaction(1,:) = t_ice.*((phi1^2)/(4*pi*f1*M1))*theta;
result.iceaction(2,:) = t_ice.*((phi2^2)/(4*pi*f2*M2))*theta;

%% general criteria
result.driftcriteria = zeros(1,L);
```

```
result.driftcriterion = zeros(1,L);
result.uwlcriterion = zeros(1,L);
result.ratiocriterion = zeros(1,L);
result.growthcriterion = zeros(1,L);

% DRIFT DIRECTION CRITERION
q = input.drift.boundaries(1)*(pi/180);
w = (input.drift.boundaries(2)+360)*(pi/180);
e = input.drift.boundaries(2)*pi/180;
r = 0;

for j=1:L
   if j ≥ input.proc.length(1) && j ≤ input.proc.length(2)
    if theta_ice(j) ≥ q && theta_ice(j) ≤ w
       result.driftcriteria(j) = 1;
    elseif theta_ice(j) ≤ e && theta_ice(j) ≥ r
       result.driftcriteria(j) = 1;
    else
       result.driftcriteria(j) = 0;
    end
   end
end
% ICE GROWTH CRITERION
% ice growth must have been less than 1 mm the preceding week
% 1 week = 7*8 = 56 indices in vector
for j = 8:L
    if t_ice(j) > 0
        if t_ice(j)-t_ice(j-8) < 0.01
            result.growthcriterion(j) = 1;
        else
            result.growthcriterion(j) = 0;
        end
    else
        result.growthcriterion(j) = 0;
    end
end

% WATERLINE DISPLACEMENT CRITERION
for j = 1:L
    if force.∆wl(j) > input.str.dwlbond
        result.uwlcriterion(j) = 1;
    else
        result.uwlcriterion(j) = 0;
    end
end

% ASPECT RATION CRITERION
for j = 1:L
    if iceprop.aspectratio(j) < 40 && iceprop.aspectratio(j) > 0
        result.ratiocriterion(j) = 1;
    else
        result.ratiocriterion(j) = 0;
    end
end
```

```
result.generalcrit = zeros(1,L);
for j=1:L
      if result.driftcriteria(j) == 1 && result.uwlcriterion(j) == 1 ...
              && result.growthcriterion(j) == 1 && ...
              result.ratiocriterion(j) == 1
          result.generalcrit(j) = 1;
      else
          result.generalcrit(j) = 0;
      end
end

%% THE 5 DIFFERENT METHODS including the general criteria
for j = 1:L
    if result.generalcrit(j) == 1
         t = 2*t_ice(j)/v_ice(j);              %duration between lock-ins
         nrcycles = round(Tevent/(input.flc.dur+t)); %nr. lockins/event
        %% METHOD 1 and 2 ; DIMENSIONLESS GROUPS BY PALMER
        % diameter (method 1)
        if v_ice(j) > (input.flc.Ddim(1)*input.str.diam*input.str.freq(2))...
         && v_ice(j) < (input.flc.Ddim(2)*input.str.diam*input.str.freq(2))
            %fulfilment of flc-criterion
            result.method1(1,j) = 1;
            %nr of cycles at step j
            result.method1(2,j) = nrcycles*Vibr_1lockin;
        else
            result.method1(1,j) = 0;
            result.method1(2,j) = 0;
        end

       % thickness (method 2)
       if (v_ice(j)/(input.str.freq(2)*t_ice(j))) > input.flc.Tdim(1) ...
            && (v_ice(j)/(input.str.freq(2)*t_ice(j))) < input.flc.Tdim(2)
            %fulfilment of flc-criterion
            result.method2(1,j) = 1;
            %nr of cycles at step j
            result.method2(2,j) = nrcycles*Vibr_1lockin;
        else
            result.method2(1,j) = 0;
            result.method2(2,j) = 0;
       end

    %% BASED ON ICE VELOCITY AND THICKNESS by BJERKAS AND SKIPLE
    % thickness range 0.2 < h < 1.2m
    % BJERKAS CRITERIA FOR THICKNESS NR OF FULFILLMENTS
      %BJERKAS VELOCITY AND THICKNESS CRITERIA (method 3)
        if t_ice(j) > 0.2 && t_ice(j) < 1.2
            v2t = 0.03*t_ice(j) + 0.07; %equation by bjerkas/skiple
            if v_ice(j) > 0.02 && v_ice(j) < v2t
                %fulfilment of flc-criterion
                result.method3(1,j) = 1;
                %nr of cycles at step j
                result.method3(2,j) = nrcycles*Vibr_1lockin;
            else
```

```
                result.method3(1,j)=0;
                result.method3(2,j)=0;
            end
        end

        % BJERKAS CRITERIA FOR VELCOTY AND TEMPERATURE (method 4)
        if T(j)>-12 && T(j)<5
            v2t = 0.03*T(j) + 0.07;
            if v_ice(j)>0.02 && v_ice(j)<v2t
                %fulfilment of flc-criterion
                result.method4(1,j) = 1;
                %nr of cycles at step j
                result.method4(2,j) = nrcycles*Vibr_1lockin;
            else
                result.method4(1,j)=0;
                result.method4(2,j)=0;
            end
        end

       %% METHOD 5 ; SUSCEPTIBILITY to FREQUENCY LOCKIN ISO 19906 A.8.2.6

       if result.iceaction(1,j) > xeta1
           % fulfilments of FLC criteria
           result.method5(1,j) = 1;
           % nr of cycles at step j
           result.method5(2,j) = nrcycles*Vibr_1lockin;
       else
           result.method5(1,j) = 0;
           result.method5(2,j) = 0;
       end

     % Criteria not fulfilled
     elseif result.generalcrit(j) == 0
         result.method1(1,j) = 0;
         result.method1(2,j) = 0;
         result.method2(1,j) = 0;
         result.method2(2,j) = 0;
         result.method3(1,j) = 0;
         result.method3(2,j) = 0;
         result.method4(1,j) = 0;
         result.method4(2,j) = 0;
         result.method5(1,j) = 0;
         result.method5(2,j) = 0;

     end
end
for j = 1:L
        %% Finding accumulated vibrations at index j for each method
        if j==1
        result.method1(3,j) = result.method1(2,j);
        result.method2(3,j) = result.method2(2,j);
        result.method3(3,j) = result.method3(2,j);
        result.method4(3,j) = result.method4(2,j);
        result.method5(3,j) = result.method5(2,j);
```

```
        else
        result.method1(3,j) = result.method1(2,j) + result.method1(3,j-1);
        result.method2(3,j) = result.method2(2,j) + result.method2(3,j-1);
        result.method3(3,j) = result.method3(2,j) + result.method3(3,j-1);
        result.method4(3,j) = result.method4(2,j) + result.method4(3,j-1);
        result.method5(3,j) = result.method5(2,j) + result.method5(3,j-1);
        end
end

%% Total number of vibrations per season for different methods
result.total=[sum(result.method1(2,:)) sum(result.method2(2,:)) ...
    sum(result.method3(2,:)) sum(result.method4(2,:)) ...
    sum(result.method5(2,:))];

end
```

B.9 fatigue.m

```
% Find fatigue damage and structure's life time
% Subprogram to program icemain.m
% Author: Andreas Meese, Nidaros 27.06.2012
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function[result] = fatiguecalc(input,ice,force,FLC)

% natural frequency of waterline translation
f = input.str.freq(2);
% daily ice thickness
t_ice = ice.ticeavg;
% seconds in day
Tday = 24*3600;
% length of vector
L = length(t_ice);

%% Function for global ice pressure
coeff1 = -0.50;
coeff2 = -0.16;
h1 = 1.0;
icepressure = @(h) input.phys.Cr*((h/h1)^(coeff1+(h/5)))*...
                ((input.str.diam/h)^coeff2);

%% Cross-sectional properties
% STEEL
%required steel area (calculated external) %m^2
A_steel = 0.169547;
% location of the steel ring from centre of pipe %m
R = 3.35;
% circumference %m
O = 2*pi*R;
% thickness of the steel ring %m
t_st = A_steel/O;
```

```
% Second moment of area of the steel ring %m^4
I_z = pi*t_st*(R^3);
y_max = R+t_st;

% CONCRETE
% area of cs.
R_concr = input.concrete.r_outer - input.concrete.thick/2;
A_cs = 2*pi*R*input.concrete.thick;

%% Expression for the stress range ΔS

% axial compressive stress Pa, i.e. the weight of the concrete acting
% on the cross-section
sigma_N = input.str.weightabovecs/A_cs;

% varying force due to lock-in conditions
% attack arm of the force
arm = (input.str.wl-7.5);

% The stress range is given as a function of the force range
% the force range is defined by ISO19906 with ΔF = Fmax*(1-q)
% q is a varying factor
q = input.sim.q(2);  %factor from iso regarding lock-in force -> ΔF

% stress range in N/m2 (Pa)
Δ_S = @(F) ((F*(1-q)*arm)/(I_z))*y_max;

%%

% force due to ice % FMAX IS HERE IN MN
result.Fmax = force.FGlobmax;

%%

% row 1 is stress range, ΔS
% row 2 is Ni, nr. cycles to failure for ΔS
% row 3 is ni, nr. cycles for ΔS
% row 4 is fatigue damage, ni/Ni

% Ni is obtained from SN-curves

%% SN-curve parameters
% b (m)
k1 = input.fat.SN(1);
k2 = input.fat.SN(2);

% limit between k1 and k2
N_star = input.fat.klimitsigma(2);
sigma_star = input.fat.klimitsigma(1); %MPa

logA1 = log(N_star)+k1*log(sigma_star);
logA2 = log(N_star)+k2*log(sigma_star);
A1 = exp(logA1);
A2 = exp(logA2);
```

```matlab

%limit between high and low cycle regimes
HLlimit = input.fat.highlow;

% yield limit of ribbed steel
sigma_yield = input.fat.rsyieldlim;

%% Case 1 : Vibrations the entire season from October to April
result.case1 = zeros(4,L);

% Vibrations the entire day with natural fn when t_ice > 0.2 m
for j = 1:L
    if t_ice(j) ≥ 0.2
    %Δ_S
    result.case1(1,j) = Δ_S(result.Fmax(j));

    %determine if k1 or k2 is to be used
    if log(result.case1(1,j)) > log(sigma_star)
        b = k1;
        A = A1;
    elseif log(result.case1(1,j)) < log(sigma_star)
        b = k2;
        A = A2;
    end

    % N_i
    result.case1(2,j) = exp(log(A) - b*log(result.case1(1,j)));

    %n_i
    result.case1(3,j) = Tday*f;
    %damage
    result.case1(4,j) = result.case1(3,j)/(result.case1(2,j));
    end
end
result.damage.case1 = sum(result.case1(4,:));

%% Case 2 : Using calculated nr. of cycles by onset criteria
result.onset1 = zeros(4,L);

% ONSET 1
for j = 1:L
    if t_ice(j) ≥ 0.2

    % Δ_S
    result.onset1(1,j) = Δ_S(result.Fmax(j));

    % SN curve parameters
    if log(result.onset1(1,j)) > log(sigma_star)
        b = k1;
        A = A1;
    elseif log(result.case1(1,j)) < log(sigma_star)
        b = k2;
        A = A2;
    end
```

```
    % N_i
    result.onset1(2,j) = exp(log(A) - b*log(result.onset1(1,j)));
    %n_i
    result.onset1(3,j) = FLC.method1(2,j);
    %damage
    result.onset1(4,j) = result.onset1(3,j)/(result.onset1(2,j));
    end
end
result.damage.onset1 = sum(result.onset1(4,:));

% ONSET 2
result.onset2 = zeros(4,L);
for j = 1:L
    if t_ice(j) ≥ 0.2

    % Δ_S
    result.onset2(1,j) = Δ_S(result.Fmax(j));

    % SN curve parameters
    if log(result.onset2(1,j)) > log(sigma_star)
        b = k1;
        A = A1;
    elseif log(result.case1(1,j)) < log(sigma_star)
        b = k2;
        A = A2;
    end
    % N_i
    result.onset2(2,j) = exp(log(A) - b*log(result.onset2(1,j)));
    %n_i
    result.onset2(3,j) = FLC.method2(2,j);
    %damage
    result.onset2(4,j) = result.onset2(3,j)/(result.onset2(2,j));
    end
end
result.damage.onset2 = sum(result.onset2(4,:));

% ONSET 3
result.onset3 = zeros(4,L);
for j = 1:L
    if t_ice(j) ≥ 0.2

    % Δ_S
    result.onset3(1,j) = Δ_S(result.Fmax(j));

    % SN curve parameters
    if log(result.onset3(1,j)) > log(sigma_star)
        b = k1;
        A = A1;
    elseif log(result.case1(1,j)) < log(sigma_star)
        b = k2;
        A = A2;
    end
    % N_i
    result.onset3(2,j) = exp(log(A) - b*log(result.onset3(1,j)));
```

```
    %n_i
    result.onset3(3,j) = FLC.method3(2,j);
    %damage
    result.onset3(4,j) = result.onset3(3,j)/(result.onset3(2,j));
    end
end
result.damage.onset3 = sum(result.onset3(4,:));

% ONSET 4
result.onset4 = zeros(4,L);
for j = 1:L
    if t_ice(j) ≥ 0.2

    % Δ_S
    result.onset4(1,j) = Δ_S(result.Fmax(j));

    % SN curve parameters
    if log(result.onset4(1,j)) > log(sigma_star)
        b = k1;
        A = A1;
    elseif log(result.case1(1,j)) < log(sigma_star)
        b = k2;
        A = A2;
    end
    % N_i
    result.onset4(2,j) = exp(log(A) - b*log(result.onset4(1,j)));
    %n_i
    result.onset4(3,j) = FLC.method4(2,j);
    %damage
    result.onset4(4,j) = result.onset4(3,j)/(result.onset4(2,j));
    end
end
result.damage.onset4 = sum(result.onset4(4,:));

% ONSET 5
result.onset5 = zeros(4,L);
for j = 1:L
    if t_ice(j) ≥ 0.2

    % Δ_S
    result.onset5(1,j) = Δ_S(result.Fmax(j));

    % SN curve parameters
    if log(result.onset5(1,j)) > log(sigma_star)
        b = k1;
        A = A1;
    elseif log(result.case1(1,j)) < log(sigma_star)
        b = k2;
        A = A2;
    end
    % N_i
    result.onset5(2,j) = exp(log(A) - b*log(result.onset5(1,j)));
    %n_i
    result.onset5(3,j) = FLC.method5(2,j);
```

```
    %damage
    result.onset5(4,j) = result.onset5(3,j)/(result.onset5(2,j));
    end
end
result.damage.onset5 = sum(result.onset5(4,:));


%% Case 3 : Observed nr. of cycles
result.case3 = zeros(1,4);
%ni
result.case3(3) = input.fat.obsyclethick(1);
%ΔS
t_case3 = input.fat.obsyclethick(2);
force_case3 = input.str.diam*icepressure(t_case3);
result.case3(1) = Δ_S(force_case3);
% N_i
if log(result.case3(1)) > log(sigma_star)
    b = k1;
    A = A1;
    result.case3(2) = exp(log(A) - b*log(result.case3(1)));
elseif log(result.case3(1)) <  log(sigma_star)
    b = k2;
    A = A2;
    result.case3(2) = exp(log(A) - b*log(result.case3(1)));
end
% damage
result.case3(4) = result.case3(3)/result.case3(2);
result.damage.case3 = result.case3(4);

% THE FINAL CALCULATION : LIFE TIME
result.lifetime.case1 = ceil(1/result.damage.case1);
result.lifetime.onset1 = ceil(1/result.damage.onset1);
result.lifetime.onset2 = ceil(1/result.damage.onset2);
result.lifetime.onset3 = ceil(1/result.damage.onset3);
result.lifetime.onset4 = ceil(1/result.damage.onset4);
result.lifetime.onset5 = ceil(1/result.damage.onset5);
result.lifetime.case3 = ceil(1/result.damage.case3);
end
```

B.10 dynamic.m

```
% Calculate dynamic response of SDOF model
% given full scale measured parameters from 12:26:00 to 12:26:40 for
% 30.03.2003 event
% Subprogram to icemain.m
% Author: Andreas Meese, Tr.heim 16.05.2012
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function[result] = dynresponse2(input,event)

thickness = event.thickness.t(13:22);  %measured thickness vector

f1 = input.str.freq(2);
M1 = input.str.strice(1)/(input.str.strice(3)^2);
%time length of simulation
t = 40;
%time between each measurement
TSTEP = t/length(thickness);
%time step simulation
tstep = input.sim.time(2);
%lower and upper q values (coefficient)
q = 0.25;%input.sim.q(1);
% iso = FLC.method5(1,:);      %vector with 0 and 1's (critieria fulfilm.)

%% maximum force at time of flc event 30.03.2003
    pg = zeros(1,length(thickness));
    staticmax = zeros(1,length(thickness));
    staticmin = zeros(1,length(thickness));
    ∆f = zeros(1,length(thickness));
    m = -0.16;                %empirical coefficient
    h1 = 1.0;                 %reference thickness
    for j = 1:length(pg)
       if input.str.diam/thickness(j) > 40
           pg(j) = 0;
       else
           if thickness(j) < 1.0
              n = -0.50 + thickness(j)/5;
           elseif thickness(j) ≥ 1.0
              n = -0.30;
           end
           pg(j) = input.phys.Cr*((thickness(j)/h1)^n)*...
           ((input.str.diam/thickness(j))^m);

           staticmax(j) = pg(j)*thickness(j)*input.str.diam*10^6;
           staticmin(j) = staticmax(j)*(1-q);
           ∆f(j) = staticmax(j)*q;
       end
    end

%timevector and force vector establishment
```

```
result.timevector = 0:tstep:t;
% result.timevector = 0:t/length(event.thickness.timevector):t;
J = length(result.timevector);

%% Load history given measured thickness
% result.F = zeros(1,J);
temp1 = 1;
temp2 = ceil(TSTEP/tstep);
steget = ceil(TSTEP/tstep);
for i = 1:size(staticmax,2)
result.zet1(temp1:temp2) = staticmax(i);
result.zet2(temp1:temp2) = staticmin(i);
result.zetdF(temp1:temp2) = staticmax(i)*q;
temp1 = temp1+steget;
temp2 = temp2+steget;

end

floor(J/floor(TSTEP/tstep));
result.F = zeros(1,length(result.zet1));
for i=1:J
    result.F(i) = result.zet1(i) + ...
        result.zetdF(i)*((sawtooth(2*pi*f1*result.timevector(i))-1)/2);
end

% %response vectors establishment
result.u = zeros(1,J); result.udot = result.u;
result.u2dot = result.u;
%
dt = tstep;                       %sample time
Fs = 1/dt;                        %sample frequency
L = length(result.u2dot);         %length of signal

%fft vectors establishment
result.fftacc = zeros(1,L);
result.freq = zeros(1,L);
result.fftamp = zeros(1,L);
%
%% Dynamic response of a SDOF model of the waterline
% Newmarks method to solve the equation of forced motion
xi = input.str.xi1(1);        %fraction of critical damping
wn = 2*pi*f1;                 %circular frequency
P = result.F(1,:);            %force vector in Newtons
M = M1;                       %system mass
K = M*wn^2;                   %system stiffness
C = 2*xi*wn*M;                %system damping
dt = input.sim.time(2);       %time step
beta = input.sim.newmark(1);  %beta in newmarks method
gamma = input.sim.newmark(2); %gamma in newmarks method

u0 = input.sim.initial(1);    %initial displacement
udot0 = input.sim.initial(2); %initial velocity

kgor = K + gamma/(beta*dt)*C + M/(beta*dt^2);
```

```
a = M/(beta*dt) + gamma*C/beta;
b = 0.5*M/beta + dt*(0.5*gamma/beta -1)*C;

dp = diff(P);

result.u(1) = u0;
result.udot(1) = udot0;
result.u2dot(1) = 1/M*(P(1)-K*u0-C*udot0);

for i = 1:(length(result.timevector)-1)
    ∆F = dp(i) + a*result.udot(i) + b*result.u2dot(i);
    du_i = ∆F/kgor;
    dudot_i = gamma/(beta*dt)*du_i - gamma/beta*result.udot(i) + ...
        dt*(1-0.5*gamma/beta)*result.u2dot(i);
    du2dot_i = 1/(beta*dt^2)*du_i - 1/(beta*dt)*result.udot(i) - ...
        0.5/beta*result.u2dot(i);
    result.u(i+1) = du_i + result.u(i);
    result.udot(i+1) = dudot_i + result.udot(i);
    result.u2dot(i+1) = du2dot_i + result.u2dot(i);
end

%% Fourier transform of acceleration
% for j = 1:2
NFFT = 2^nextpow2(L); % Next power of 2 from length of y
abc = result.u2dot(1,:);
result.fftacc = fft(abc,NFFT)/L;
result.freq = Fs/2*linspace(0,1,NFFT/2+1);
% amplitude of the one sided spectrum
result.fftamp = 2*abs(result.fftacc(1:NFFT/2+1));
end
```

B.11 flcevent.m

```
% Process the full scale data of 30.03.2003
% ANM 22.05.2013

function [flcev] = processevent(input,ice,event)

% Convert from seconds to index in timevector
∆t = event.time(end)/length(event.time);
steady_start = ceil(input.steadyevent(1)/∆t);
steady_stop = ceil(input.steadyevent(2)/∆t);
%timevector
flcev.time = event.time(steady_start:steady_stop);

%% Ice thickness
%measured
% measured thickness 30.03.2003
flcev.thick.tmeas = event.thickness.t;
% measured timevector
flcev.thick.timemeas = event.thickness.timevector;
```

```

% %calculated thickness
% calculated thickness 30.03.2003
% flcev.thick.calc = ice.thick(2176:2183);
% flcev.thick.calctime = 1:1:input.proc.fmeas; %1 to 8, 1 is midnight

%% Forces
flcev.entireforce =  event.totalforce.sum;
flcev.totalforce = event.totalforce.sum(steady_start:steady_stop);

%% Displacements in
flcev.displ = event.inclination(steady_start:steady_stop,:);

%% Accelerations
flcev.entireacc = [event.acc.lev19,event.acc.lev37];
flcev.acc = [event.acc.lev19(steady_start:steady_stop),...
    event.acc.lev37(steady_start:steady_stop)];

%% Fourier transform of accelerations at 19.6m and 37.1m
% make frequency vector given length of acceleration vectors and
% frequency/timestep in time vector
dt = event.time(2)-event.time(1);          %sample time
Fs = 1/dt;                                 %sample frequency
L = length(event.acc.lev19(steady_start:steady_stop));   %length of signal

NFFT = 2^nextpow2(L); % Next power of 2 from length of y
flcev.fftacc19 = fft(event.acc.lev19(steady_start:steady_stop),NFFT)/L;
flcev.fftacc37 = fft(event.acc.lev37(steady_start:steady_stop),NFFT)/L;
flcev.freq = Fs/2*linspace(0,1,NFFT/2+1);
% amplitude of the one sided spectrum
flcev.amp19 = 2*abs(flcev.fftacc19(1:NFFT/2+1));
flcev.amp37 = 2*abs(flcev.fftacc37(1:NFFT/2+1));
end
```

B.12 windplots.m

```
% Subprogram to the program icemain.m
% Author: Andreas Meese, tr.heim 17.06
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%subplot(2,1,1)

function [] = windplots(input,wind,iceprop)

months = ['Jul';'Oct';'Jan';'Apr';'Jul'];
%A(1,1:6) = 0
%axis([XMIN XMAX YMIN YMAX])
copy_to_folder = input.outputloc(1,:); %same folder as iceproperties
fontsize=input.fontsize;
L = length(input.fs.windd);

%% WIND PROPERTIES
```

```
% measured wind directions
figure(666)
subplot(2,1,1)
hold on
set(gca,'FontSize',fontsize)
set(gca,'XTick',1:722.5:L)
set(gca,'XTickLabel',months)
%axis([0 L 0 14])
ylabel('Deg. cw to North')
title('Average wind direction')
plot(input.fs.windd,'Color','k')
box off
set(gcf, 'PaperPositionMode', 'manual');
set(gcf, 'PaperUnits', 'centimeters');
set(gcf, 'PaperPosition', [0 0 16 12]);
hold off

subplot(2,1,2)
circ_plot(circ_ang2rad(input.fs.windd),'hist',[],20,true,true,...
    'linewidth',2,'color','r')

% averaged wind direction with std
figure(667)
hold on
ylabel('Deg. cw to North')
xlabel('Day nr. after 01.07.2002')
title('Average wind direction')
plot(wind.avgwind(:,2).*(180/pi),'Color','k')
plot((wind.avgwind(:,2)-wind.s_alpha).*(180/pi),'Color','r')
plot((wind.avgwind(:,2)+wind.s_alpha).*(180/pi),'Color','r')
legend('24hr avg dir','std','std')
box off
set(gcf, 'PaperPositionMode', 'manual');
set(gcf, 'PaperUnits', 'centimeters');
set(gcf, 'PaperPosition', [0 0 16 12]);
hold off
```

B.13 iceplots.m

```
% Subprogram to the program icemain.m
% Author: Andreas Meese, Bergen 28.02.2012
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%subplot(2,1,1)
%% EMIR SIN MAATE
function [] = iceplots(input,ice,wind)

months = ['Jul';'Oct';'Jan';'Apr';'Jul'];
months2 = ['Sep';'Oct';'Nov';'Dec';'Jan';'Feb';'Mar';'Apr';'May'];
months3 = ['Nov';'Dec';'Jan';'Feb';'Mar';'Apr';'May'];
%A(1,1:6) = 0
%axis([XMIN XMAX YMIN YMAX])
```

```
copy_to_folder = input.outputloc(1,:);
fontsize=input.fontsize;
L = length(ice.thick);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% Air temperature and ice thickness
figure(1)
subplot(2,1,1)
hold on
plot(input.fs.temp,'Color','k')
set(gca,'FontSize',fontsize)
set(gca,'XTick',1:722.5:L)
set(gca,'XTickLabel',months)
axis([0 L -32 30])
title('Air temperature')
ylabel('T_{air} (\,^{\circ}C)')
% Add a horizontal line for the input temperature
line([0;length(input.fs.temp)],...
    [input.phys.water(1);input.phys.water(1)],'Color',[0 0 0])
%legend('Average','Freezing temp','Location','Best')
box off
hold off

subplot(2,1,2)
hold on
plot(ice.thick,'Color','k')
set(gca,'FontSize',fontsize)
set(gca,'XTick',1:722.5:L)
set(gca,'XTickLabel',months)
% xlabel('Time')
axis([1 L 0 1.0])
title('Ice thickness')
ylabel('h_{ice} (m)')
box off
matlab2tikz('filename',strcat(copy_to_folder,'\airvsthick.tex'),...
    'showInfo',false,'height','\figureheight','width','\figurewidth')

%% ice drift
driften = ice.icedrift(2,:);
for j = 1:length(driften)
    if driften(j) > 2*pi
        driften(j) = driften(j) - 2*pi;
    end
end
figure(2)
subplot(2,1,1)
hold on
set(gca,'FontSize',fontsize)
set(gca,'XTick',60:30:300)
set(gca,'XTickLabel',months2)
axis([75 305 0 360])
ylabel('\theta_{ice} (\,^{\circ})')
title('Ice drift direction')
plot(driften*180/pi,'Color','k')
```

```
box off
set(gcf, 'PaperPositionMode', 'manual');
set(gcf, 'PaperUnits', 'centimeters');
set(gcf, 'PaperPosition', [0 0 16 12]);
hold off

subplot(2,1,2)
hold on
set(gca,'FontSize',fontsize)
set(gca,'XTick',60:30:300)
set(gca,'XTickLabel',months2)
axis([75 305 0 0.165])
ylabel('V_{ice} (m/s)')
title('Ice drift velocity')
plot(ice.icedrift(1,:),'Color','k')
box off
set(gcf, 'PaperPositionMode', 'manual');
set(gcf, 'PaperUnits', 'centimeters');
set(gcf, 'PaperPosition', [0 0 16 12]);
hold off
matlab2tikz('filename',strcat(copy_to_folder,'\icedrift.tikz')...
    ,'showInfo',false,'height','\figureheight','width','\figurewidth')


%% global pressure for sea ice
figure(4)
hold on
set(gca,'FontSize',fontsize)
set(gca,'XTick',120:30:300)
set(gca,'XTickLabel',months3)
axis([134 299 0 2.2])
plot(ice.pg,'Color','k')
%title('Global pressure for sea ice')
ylabel('p_{G} (MPa)')
% Add a horizontal line for the input temperature
line([0;length(input.fs.temp)],...
    [input.phys.water(1);input.phys.water(1)],'Color',[0 0 0])
% legend('Average','Freezing temp','Location','SouthEast')
set(gcf, 'PaperPositionMode', 'manual');
set(gcf, 'PaperUnits', 'centimeters');
set(gcf, 'PaperPosition', [0 0 20 10]);
matlab2tikz('filename',strcat(copy_to_folder,'\globalpressure.tikz')...
    ,'showInfo',false,'height','\figureheight','width','\figurewidth')
hold off

%% aspect ratio between D and hice
figure(47)
hold on
set(gca,'FontSize',fontsize)
set(gca,'XTick',1:90:L/8)
set(gca,'XTickLabel',months)
axis([89 300 0 150])
ylabel('D/h_{ice} (-)')
plot(ice.aspectratio,'Color','k','LineStyle','--')
```

```
line([0;300],[40;40],'Color',[0 0 0],'LineStyle','-')
legend('Ratio','Boundary between regimes',...
    'Location','NorthEast')
matlab2tikz('filename',strcat(copy_to_folder,'\aspect.tikz')...
    ,'showInfo',false,'height','\figureheight','width','\figurewidth')
hold off

%% modeshape
figure(101)
plot(input.str.modeshape1(:,2),input.str.modeshape1(:,1),'k')
box off
% Create xlabel
xlabel('\phi_{1}(z)','FontSize',12);
% Create title
% title('1^{st} mode shape','FontSize',12);
stringen = ...
    'C:\Users\Andreas\Dropbox\NTNU\Master\Report\latex\figures\theory';
% Create ylabel
ylabel('z (m)','FontSize',12);
matlab2tikz('filename',strcat(stringen,'\modeshape1.tex')...
    ,'showInfo',false,'height','\figureheight','width','\figurewidth')


end
```

B.14 onsetplot.m

```
function [] = onsetplot(input,ice,FLC)

fontsize=input.fontsize;
L = length(FLC.generalcrit);
Lvector = 1:1:L;
months = ['Jul';'Oct';'Jan';'Apr';'Jul'];
months2 = ['Sep';'Oct';'Nov';'Dec';'Jan';'Feb';'Mar';'Apr';'May'];
months4 = ['Jan';'Feb';'Mar';'Apr';'May'];

copy_to_folder = input.outputloc(2,:);
%% FULFILMENTS OF FLC CRITERIA AT TIME j
figure(5005)
hold on
set(gca,'FontSize',fontsize)
title('Fulfilments of FLC criteria')
plot(FLC.method1(1,:)*0.7,'.','Color','k')
plot(FLC.method2(1,:).*0.8,'.','Color','k')
plot(FLC.method3(1,:).*0.9,'.','Color','k')
plot(FLC.method4(1,:),'.','Color','k')
plot(FLC.method5(1,:)*1.1,'.','Color','k')
ylabel('Method')
% xlabel('Time')
set(gca,'XTick',180:30:330)
set(gca,'XTickLabel',months4)
```

```
set(gca,'YTick',0.7:0.1:2)
set(gca,'YTickLabel',['1';'2';'3';'4';'5'])
axis([180 300 0.65 1.15])
text(305,0.7,num2str(sum(FLC.method1(1,:))))
text(305,0.8,num2str(sum(FLC.method2(1,:))))
text(305,0.9,num2str(sum(FLC.method3(1,:))))
text(305,1.0,num2str(sum(FLC.method4(1,:))))
text(305,1.1,num2str(sum(FLC.method5(1,:))))
set(gcf, 'PaperPositionMode', 'manual');
set(gcf, 'PaperUnits', 'centimeters');
set(gcf, 'PaperPosition', [0 0 15 10]);
matlab2tikz('filename',strcat(copy_to_folder,...
    '\criteriafulfilments.tikz'),'showInfo',false,...
    'height','\figureheight','width','\figurewidth')
box off
hold off

figure(50)  % FERDIG
hold on
box off
set(gca,'XTick',60:30:300)
set(gca,'XTickLabel',months2)
set(gca,'FontSize',fontsize)
%title('Dynamic stability criterion by ISO 19906:2010')
plot(FLC.iceaction(2,:),'Color','k','LineStyle','-','Linewidth',1)
plot(FLC.iceaction(1,:),'Color','k','LineStyle','-.','Linewidth',1)
line([0 L],[0.02 0.02],'Color','k','LineStyle','--','Linewidth',3)
set(gcf, 'PaperPositionMode', 'manual');
set(gcf, 'PaperUnits', 'centimeters');
set(gcf, 'PaperPosition', [0 0 25 16]);
axis([85 299 0 0.3])
legend('Heinonen et. al. (2004)', 'Guo (2012)','Critical damping',...
    'Location','NorthWest')
% xlabel('Time')
ylabel('Ice-action (-)')
matlab2tikz('filename',strcat(copy_to_folder,'\isocriteria.tikz')...
,'showInfo',false,'height','\figureheight','width','\figurewidth')
hold off
end
```

B.15 vibrationplot.m

```
function [] = vibrationplot(input,ice,FLC)

fontsize=input.fontsize;
L = length(FLC.generalcrit);
months = ['Jan';'Feb';'Mar';'Apr';'May'];
copy_to_folder = input.outputloc(3,:);
ul_vibrbars = 2.2E5;
linewidth = 1;
%% ACCUMULATED NUMBER OF VIBRATIONS
```

```
figure(7)
hold on
set(gca,'FontSize',fontsize)
set(gca,'XTick',180:30:300)
set(gca,'XTickLabel',months)
axis([180 300 1000 5.0E6])
plot(FLC.method1(3,:),'Color','k','LineWidth',linewidth)
plot(FLC.method2(3,:),'-.','Color','k','LineWidth',linewidth+2)
plot(FLC.method3(3,:),'--','Color','k','LineWidth',linewidth)
plot(FLC.method4(3,:),':','Color','k','LineWidth',linewidth)
plot(FLC.method5(3,:),'-.','Color','k','LineWidth',linewidth+0.5)
% xlabel('Time')
% title('Accumulated nr. of frequency locked-in vibrations')
legend('Method 1','Method 2','Method 3','Method 4','Method 5', ...
    'Location','NorthWest')
set(gcf, 'PaperPositionMode', 'manual');
set(gcf, 'PaperUnits', 'centimeters');
set(gcf, 'PaperPosition', [0 0 20 15]);
matlab2tikz('showInfo', false,'filename',strcat(copy_to_folder,...
    '\accumulatedvib.tikz')...
    ,'height','\figureheight','width','\figurewidth')
box off
hold off


%% TOTAL NUMBER OF VIBRATIONS

y = [FLC.total(1) FLC.total(2) FLC.total(3) ...
    FLC.total(4) FLC.total(5)];
x =1:1:5;
figure(8); bar(x,y,0.4,'k');
set(gca,'FontSize',fontsize)
hold on
grid off
%title('Number of vibrations for one season')
ylabel('n_{cycles} (-)')
xlabel('Method')
% axis([0.5 5.5 0 3.2E7])
box off
set(gcf, 'PaperPositionMode', 'manual');
set(gcf, 'PaperUnits', 'centimeters');
set(gcf, 'PaperPosition', [0 0 16 12]);
matlab2tikz('showInfo', false,'filename',strcat(copy_to_folder,...
    '\totalflc.tikz')...
    ,'height','\figureheight','width','\figurewidth')


%% VIBRATIONS IN TIME

bar1 = FLC.method1(2,:);
bar2 = FLC.method2(2,:);
bar3 = FLC.method3(2,:);
bar4 = FLC.method4(2,:);
bar5 = FLC.method5(2,:);
```

```
x = 1:1:L;

figure(90);
subplot(3,2,1)
hold on
baren=bar(x, bar1, 'FaceColor', 'k', 'EdgeColor', 'k');
% plot(input.fs.temp,'Color','k')
set(gca,'FontSize',fontsize)
set(gca,'XTick',180:30:300)
set(gca,'XTickLabel',months)
axis([180 300 0 ul_vibrbars])
title('Method 1')
% annotation('textbox', [.2 .2 .2 .2], 'String', 'Method 1');
% ylabel('Nr. of vibrations [-]')

subplot(3,2,2)
hold on
baren=bar(x, bar2, 'FaceColor', 'k', 'EdgeColor', 'k');
% plot(input.fs.temp,'Color','k')
set(gca,'FontSize',fontsize)
set(gca,'XTick',180:30:300)
set(gca,'XTickLabel',months)
axis([180 300 0 ul_vibrbars])
title('Method 2')
% annotation('textbox', [.2 .7 .2 .2], 'String', 'Method 2');

subplot(3,2,3)
hold on
baren=bar(x, bar3, 'FaceColor', 'k', 'EdgeColor', 'k');
% plot(input.fs.temp,'Color','k')
set(gca,'FontSize',fontsize)
set(gca,'XTick',180:30:300)
set(gca,'XTickLabel',months)
axis([180 300 0 ul_vibrbars])
title('Method 3')
% annotation('textbox', [.7 .4 .1 .1], 'String', 'Method 3');
% ylabel('Nr. of vibrations [-]')
% xlabel('Time')

subplot(3,2,4)
hold on
baren=bar(x, bar4, 'FaceColor', 'k', 'EdgeColor', 'k');
% plot(input.fs.temp,'Color','k')
set(gca,'FontSize',fontsize)
set(gca,'XTick',180:30:300)
set(gca,'XTickLabel',months)
axis([180 300 0 ul_vibrbars])
% annotation('textbox', [.7 .7 .1 .1], 'String', 'Method 4');
% ylabel('Nr. of vibrations [-]')
% xlabel('Time')
title('Method 4')
% set(gcf, 'PaperPositionMode', 'manual');
% set(gcf, 'PaperUnits', 'centimeters');
% set(gcf, 'PaperPosition', [0 0 20 15]);
```

```

subplot(3,2,5)
hold on
baren=bar(x, bar5, 'FaceColor', 'k', 'EdgeColor', 'k');
% plot(input.fs.temp,'Color','k')
set(gca,'FontSize',fontsize)
set(gca,'XTick',180:30:300)
set(gca,'XTickLabel',months)
axis([180 300 0 ul_vibrbars])
% annotation('textbox', [.7 .7 .1 .1], 'String', 'Method 4');
% ylabel('Nr. of vibrations [-]')
% xlabel('Time')
title('Method 5')
set(gcf, 'PaperPositionMode', 'manual');
set(gcf, 'PaperUnits', 'centimeters');
set(gcf, 'PaperPosition', [0 0 20 20]);
matlab2tikz('filename',strcat(copy_to_folder,'\vibrationsintime.tikz')...
    ,'showInfo',false,'height','\figureheight','width','\figurewidth')
end
```

B.16 fatigueplot.m

```
% Subprogram to the program icemain.m
% Author: Andreas Meese, Bergen 28.02.2012
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function [] = fatigueplot(input,fatigue)
months = ['Nov';'Dec';'Jan';'Feb';'Mar';'Apr';'May'];
fontsize = input.fontsize;
linewidth = 1;
copy_to_folder = ...
   'C:\Users\Andreas\Dropbox\NTNU\Master\Report\latex\figures\results\fat';
%% STRESS RANGE AND GLOBAL FORCE
figure(97)
subplot(2,1,1)
hold on
set(gca,'FontSize',fontsize)
set(gca,'XTick',130:30:330)
set(gca,'XTickLabel',months)
plot(fatigue.Fmax,'Color','k','LineWidth',linewidth)
axis([137 299 2 9])
% xlabel('Time')
ylabel('F_{G} (MN)')
title('Global force')

subplot(2,1,2)
hold on
set(gca,'FontSize',fontsize)
set(gca,'XTick',130:30:330)
set(gca,'XTickLabel',months)
axis([137 299 30 100])
```

```
title('Stress range')
% xlabel('Time')
ylabel('\Delta \sigma (MPa)')
plot(fatigue.case1(1,:),'Color','k','LineWidth',linewidth)
set(gcf, 'PaperPositionMode', 'manual');
set(gcf, 'PaperUnits', 'centimeters');
set(gcf, 'PaperPosition', [0 0 20 15]);
box off
hold off
matlab2tikz('showInfo', false,'filename',...
    strcat(copy_to_folder,'\DsigmaFglob.tikz')...
    ,'height','\figureheight','width','\figurewidth')

end
```

B.17 forceplots.m

```
% Subprogram to the program icemain.m
% Author: Andreas Meese, Bergen 28.02.2012
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function [] = verticalplots(input,ice,verticalforce,selected)
months = ['Jul';'Oct';'Jan';'Apr';'Jul'];

months = ['Jul';'Oct';'Jan';'Apr';'Jul'];
months2 = ['Sep';'Oct';'Nov';'Dec';'Jan';'Feb';'Mar';'Apr';'May'];
fontsize = 14;
copy_to_folder = input.outputloc(4,:);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% Forces on vertical structure
figure(10)
subplot(2,1,1)
hold on
plot(verticalforce.FGlobmax,'k')
set(gca,'FontSize',fontsize)
set(gca,'XTickLabel',months)
set(gca,'XTick',1:722.5:2920)
axis([0 2920 0 12])
ylabel('Ice load [MN]')
title('Ice load - ISO 19906- Vertical Global force')
%legend('Average','Location','NorthEast')
set(gcf, 'PaperPositionMode', 'manual');
set(gcf, 'PaperUnits', 'centimeters');
set(gcf, 'PaperPosition', [0 0 16 12]);
hold off

subplot(2,1,2)
hold on
set(gca,'FontSize',fontsize)
set(gca,'XTick',1:722.5:2920)
set(gca,'XTickLabel',months)
```

```
axis([0 2920 0 0.9])
title('Ice thickness')
ylabel('Thickness [m]')
plot(ice.thick,'k')
hold off
%legend('Ice - vertical','Location','NorthEast')
set(gcf, 'PaperPositionMode', 'manual');
set(gcf, 'PaperUnits', 'centimeters');
set(gcf, 'PaperPosition', [0 0 16 12]);
matlab2tikz('filename',strcat(copy_to_folder,'\verticalforces.tikz')...
    ,'showInfo',false,'height','\figureheight','width','\figurewidth')

%% Force vs. waterline displacement using k from STRICE (2004)
figure(42)
subplot(2,1,1)
hold on
set(gca,'FontSize',fontsize)
%set(gca,'XTick',1:722.5:2920)
%set(gca,'XTickLabel',months)
%title('Ice thickness')
title('Global force vs. static waterline displacement')
axis([0 1000*max(verticalforce.Δwl) 0 max(verticalforce.FGlobmax)])
ylabel('F_{G} (MN)')
xlabel('\Δ_{wl} (mm)')
plot(1000.*verticalforce.Δwl,verticalforce.FGlobmax,'k')
hold off
%legend('Ice - vertical','Location','NorthEast')
set(gcf, 'PaperPositionMode', 'manual');
set(gcf, 'PaperUnits', 'centimeters');
set(gcf, 'PaperPosition', [0 0 16 12]);

subplot(2,1,2)
hold on
set(gca,'FontSize',fontsize)
set(gca,'XTick',60:30:300)
set(gca,'XTickLabel',months2)
title('Static displacement vs. FLC onset limit')
axis([120 305 0 5.5])
ylabel('time')
ylabel('\Δ_{wl} (mm)')
plot(1000.*verticalforce.Δwl,'k')
line([0;length(verticalforce.Δwl)],...
    [1;1],'Color',[0 0 0],'LineStyle','--')
legend('Static displacement','FLC onset limit','Location','NorthWest')
set(gcf, 'PaperPositionMode', 'manual');
set(gcf, 'PaperUnits', 'centimeters');
set(gcf, 'PaperPosition', [0 0 16 12]);
matlab2tikz('filename',strcat(copy_to_folder,...
    '\forcewldisplacement.tikz'),'showInfo',false,'height',...
    '\figureheight','width','\figurewidth')
end
```

B.18 dynamicplots.m

```matlab
% Subprogram to the program icemain.m
% Author: Andreas Meese, Bergen 28.02.2012
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function [] = dynamicplots(input,ice,FLC,force,dynamic)
months = ['Jul';'Oct';'Jan';'Apr';'Jul'];
%seconds = char('0','10','20','30','40','50','60','70','80');
seconds = char('0','2','4','6','8','10');
copy_to_folder =input.outputloc(5,:);

fontsize = input.fontsize;
L = length(dynamic.F);
g = L/(input.sim.time(1)/10);

% Load history
figure(12)
hold on
plot(dynamic.F(1,:),'k')
set(gca,'FontSize',fontsize)
set(gca,'XTickLabel',seconds)
set(gca,'XTick',0:200:round(g))
%set(gca,'YtickLabel',)
%set(gca,'YTick',[])
% a = dynamic.F(1:round(g));
% [maxV maxI] = max(a);
% [minV minI] = min(a);
% text(maxI,dynamic.F(maxI),strcat(num2str(maxV), ' MPa'),...
%     'VerticalAlignment','bottom','HorizontalAlignment','right',...
%     'Fontsize',input.fontsize)
% text(minI+100,dynamic.F(minI),strcat(num2str(minV),' MPa'),...
%     'VerticalAlignment','top','HorizontalAlignment','left',...
%     'Fontsize',input.fontsize)
% axis([0 g dynamic.Fmin-1 dynamic.Fmax+1])
ylabel('Ice load [MN]')
xlabel('Time [s]')
title('Load history')

hold off
set(gcf, 'PaperPositionMode', 'manual');
set(gcf, 'PaperUnits', 'centimeters');
set(gcf, 'PaperPosition', [0 0 16 12]);
matlab2tikz('filename',strcat(copy_to_folder,'\loadhistory.tikz')...
    ,'showInfo',false,'height','\figureheight','width','\figurewidth')

% Displacement

figure(13)
subplot(3,1,1)
hold on
```

```
box off
plot(dynamic.u(1:round(g)).*1000,'k')
set(gca,'FontSize',fontsize)
set(gca,'XTickLabel',seconds)
set(gca,'XTick',0:200:round(g))
% axis([0 g 0 5])
ylabel('mm')
title('Displacement')
set(gcf, 'PaperPositionMode', 'manual');
set(gcf, 'PaperUnits', 'centimeters');
set(gcf, 'PaperPosition', [0 0 16 12]);
hold off

subplot(3,1,2)
hold on
box off
plot(dynamic.udot(1:round(g)),'k')
set(gca,'FontSize',fontsize)
set(gca,'XTickLabel',seconds)
% set(gca,'XTick',0:200:round(g))
xlabel('Time (s)')
ylabel('m/s')
title('Velocity')
% legend('Strice','Guo','Location','NorthEast')
% axis([0 g -0.81 0.81])
set(gcf, 'PaperPositionMode', 'manual');
set(gcf, 'PaperUnits', 'centimeters');
set(gcf, 'PaperPosition', [0 0 16 12]);

subplot(3,1,3)
hold on
box off
plot(dynamic.u2dot(1:round(g)),'k')
set(gca,'FontSize',fontsize)
set(gca,'XTickLabel',seconds)
set(gca,'XTick',0:200:round(g))
xlabel('Time [s]')
ylabel('m/s^{2}')
title('Acceleration')
% legend('Strice','Guo','Location','NorthEast')
% axis([0 g -0.81 0.81])
set(gcf, 'PaperPositionMode', 'manual');
set(gcf, 'PaperUnits', 'centimeters');
set(gcf, 'PaperPosition', [0 0 16 12]);
matlab2tikz('filename',strcat(copy_to_folder,'\displveloacc.tikz')...
    ,'showInfo',false,'height','\figureheight','width','\figurewidth')


% Plot single-sided amplitude spectrum for sdof acceleration.
figure(14)
hold on
box off
set(gca,'FontSize',fontsize)
plot(dynamic.freq,dynamic.fftamp,'k','LineWidth',2)
```

```
title('Single-Sided Amplitude Spectrum of Acceleration')
axis([0 5.3 0 0.6])
xlabel('Frequency [Hz]')
ylabel('|Acc(f)|')
% legend('Strice','Guo','Location','NorthEast')
set(gcf, 'PaperPositionMode', 'manual');
set(gcf, 'PaperUnits', 'centimeters');
set(gcf, 'PaperPosition', [0 0 16 12]);
matlab2tikz('filename',strcat(copy_to_folder,'\fftacc.tikz')...
    ,'showInfo',false,'height','\figureheight','width','\figurewidth')

end
```

B.19 comparisonplots.m

```
% Subprogram to the program icemain.m
% Author: Andreas Meese, Bergen 28.02.2012
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function [] = comparisonplots(input,dynamic,flcev)
months = ['Jul';'Oct';'Jan';'Apr';'Jul'];
%seconds = char('0','10','20','30','40','50','60','70','80');
seconds = char('0','2','4','6','8','10');
timevector = char('12:26:00','12:26:10','12:26:20',...
    '12:26:30','12:26:40');
timevector2 = char('12:24:30','12:25:30','12:26:30','12:27:30');
timevector3 = char('12:24:30','12:25:00','12:25:30','12:26:00',...
    '12:26:30','12:27:00','12:27:30');
a = size(timevector);
b = size(timevector2);
copy_to_folder = input.outputloc(6,:);

% 30.03.2003 event force and acceleration
figure(395)
subplot(2,1,1)
hold on
box off
set(gca,'XTick',1:5051:length(flcev.entireforce)+1)
set(gca,'XTickLabel',timevector2)
set(gca,'FontSize',input.fontsize)
% set(gca,'Fontname','cmr10')
% title('Measured total force')
ylabel('kN','interpreter','latex')
axis([1 length(flcev.entireforce)+1 0 max(flcev.entireforce)])
set(gcf, 'PaperPositionMode', 'manual');
set(gcf, 'PaperUnits', 'centimeters');
% set(gcf, 'PaperPosition', [0 0 16 12]);
plot(flcev.entireforce,'k')

subplot(2,1,2)
hold on
```

```
box off
set(gca,'XTick',1:5051:length(flcev.entireforce)+1)
set(gca,'XTickLabel',timevector2)
set(gca,'FontSize',input.fontsize)
plot(flcev.entireacc(:,1),'k')
xlabel('Time','interpreter','latex')
ylabel('m/s$^{2}$','interpreter','latex')
axis([1 length(flcev.entireforce)+1 -1.8 2.15])
hold off
% title('Measured acceleration at 19.6 m above sea bed')
set(gcf, 'PaperPositionMode', 'manual');
set(gcf, 'PaperUnits', 'centimeters');
% set(gcf, 'PaperPosition', [0 0 16 12]);
% set(gca,'Fontname','cmr10')
name = 'entireevent';
name_pdf = strcat(name,'.pdf');
export_fig(name_pdf,'-pdf','-transparent','-painters');
copyfile(name_pdf,copy_to_folder);
delete(name_pdf);
clear name_pdf;

%% FORCE FUNCTIONS
figure(15)
hold on
box off
L = length(flcev.totalforce);
set(gca,'FontSize',input.fontsize)
set(gca,'XTick',1:L/(a(1)-1):L+1)
set(gca,'XTickLabel',timevector)
plot(dynamic.F(1,:)./(10^6),'k')
plot(flcev.totalforce./(10^3),'r')
ylabel('MN')
xlabel('Time')
axis([0 (length(flcev.totalforce))+5 0 10])
% title('Comparison between measured and computed forces')
text(1440,8.3,'Calculated total force')
text(2000,4.5,'Measured total force')
% legend('Calculated','Measured','Location','best')
set(gcf, 'PaperPositionMode', 'manual');
set(gcf, 'PaperUnits', 'centimeters');
set(gcf, 'PaperPosition', [0 0 16 12]);
matlab2tikz('filename',strcat(copy_to_folder,'\forcecomparison.tikz')...
    ,'showInfo',false,'height','\figureheight','width','\figurewidth')
hold off

%% ACCELERATIONS
figure(17)
subplot(2,1,1)
hold on
box off
set(gca,'FontSize',input.fontsize)
%set(gca,'XTick',1:722.5:L)
%set(gca,'XTickLabel',months)
plot(flcev.acc(:,1),'k','LineStyle','-')
```

```
ylabel('m/s$^{2}$')
% xlabel('Time')
axis([0 length(dynamic.u2dot) -1.4 2.1])
L = length(flcev.acc);
set(gca,'FontSize',input.fontsize)
set(gca,'XTick',1:L/(a(1)-1):L+1)
set(gca,'XTickLabel',timevector)
title('Measured acceleration at 19.65 m')
set(gcf, 'PaperPositionMode', 'manual');
set(gcf, 'PaperUnits', 'centimeters');
set(gcf, 'PaperPosition', [0 0 16 12]);

subplot(2,1,2)
hold on
box off
set(gca,'FontSize',input.fontsize)
%set(gca,'XTic:k',1:722.5:L)
%set(gca,'XTickLabel',months)
plot(dynamic.u2dot,'k','LineStyle','-')
ylabel('m/s$^{2}$')
xlabel('Time')
axis([0 length(dynamic.u2dot) -1.4 2.1])
L = length(flcev.acc);
set(gca,'FontSize',input.fontsize)
set(gca,'XTick',1:L/(a(1)-1):L+1)
set(gca,'XTickLabel',timevector)
title('Calculated acceleration at 14.20 m')
set(gcf, 'PaperPositionMode', 'manual');
set(gcf, 'PaperUnits', 'centimeters');
set(gcf, 'PaperPosition', [0 0 16 12]);
matlab2tikz('filename',strcat(copy_to_folder,'\accelcomparison.tikz')...
    ,'showInfo',false,'height','\figureheight','width','\figurewidth')
hold off

%% FFT OF ACCELERATIONS
% Plot single-sided amplitude spectrum.
figure(18)
hold on
box off
set(gca,'FontSize',input.fontsize)
plot(flcev.freq,flcev.amp19,'r','LineStyle','-','LineWidth',1)
plot(dynamic.freq,dynamic.fftamp,'b','LineStyle','-','LineWidth',1)
% title('Single-Sided Amplitude Spectrum of Acceleration')
axis([0,5,0,max(dynamic.fftamp)])
xlabel('f (Hz)')
ylabel('|Acc(f)|')
box off
legend('Measured at 19.6 m','Calculated at 14.2 m')
set(gcf, 'PaperPositionMode', 'manual');
set(gcf, 'PaperUnits', 'centimeters');
set(gcf, 'PaperPosition', [0 0 16 12]);
matlab2tikz('filename',strcat(copy_to_folder,'\fftcomparison.tikz')...
    ,'showInfo',false,'height','\figureheight','width','\figurewidth')

```

```
%% Ice thickness
figure(19)
hold on
box off
set(gca,'FontSize',input.fontsize)
plot(flcev.thick.tmeas,'k','LineStyle','-')
% line([1;length(flcev.thick.tmeas)],...
%     [flcev.thick.calc(5);flcev.thick.calc(5)],'Color',[0 0 0])
% title('Measured ice thickness during 30.03.2003 event')
ylabel('m')
xlabel('Time')
L = length(flcev.thick.tmeas);
set(gca,'XTick',1:12:37)
set(gca,'XTickLabel',timevector2)
axis([1 37 0.65 0.95 ])
% legend('Measured','Calculated','Location','NorthWest')
set(gcf, 'PaperPositionMode', 'manual');
set(gcf, 'PaperUnits', 'centimeters');
set(gcf, 'PaperPosition', [0 0 16 12]);
matlab2tikz('filename',strcat(copy_to_folder,'\thickevent.tikz')...
    ,'showInfo',false,'height','\figureheight','width','\figurewidth')
hold off
end
```