NTNU
Norwegian University of
Science and Technology

Methods for Ice-Model Updating Using a Mobile Sensor Network

Anne Mai Ersdal

Master of Science in Engineering Cybernetics
Submission date: June 2011
Supervisor: Lars Imsland, ITK

Norwegian University of Science and Technology
Department of Engineering Cybernetics

Faculty of Information Technology, Mathematics and Electrical Engineering
Department of Engineering Cybernetics

PROJECT DESCRIPTION SHEET

Name of the candidate: Anne Mai Ersdal

Thesis title (Norwegian):

Thesis title (English): Methods for ice model updating using a mobile sensor network

Background

As offshore oil- and gas production enters arctic seas, the presence of ice becomes a substantial challenge in dynamic positioning of vessels. An important part of a DP system in the arctic, is a system for doing "ice management", which likely will include the use of observer techniques including appropriate measurements for constructing an overall picture of the ice situation, and to be able to predict movement and impact of ice. Possible sensors for ice measurements include video/ radar measurements from UAVs and sensors placed on the ice by e.g. UAVs, that is, sensors that can be located based on information in an ice management observer. This project work will look at how model information can be used for “optimal” information gathering from UAVs.

The master thesis builds in part on the project work, “Measurement selection based on observability Gramians”.

Work description

1. Give a brief overview over the challenges with DP in ice, and how an observer scheme for ice dynamics can be help in this regard.
2. Give a brief account of ice features and approaches to modeling of ice.
3. Implement the ice model of Hibler. Set it up on a simple grid. Make illustrative simulations and attempt to validate (for instance using results in the literature).
4. Implement an Ensemble Kalman Filter based on the Hibler ice model, using measurements that could conceivably come from an UAV system.
5. Investigate method(s) (for instance based on observability Gramians or state covariance in a Kalman Filter) that can be used to decide where measurements should be taken.

Start date: 10 January, 2011 **Due date:** 04 July, 2011

Supervisor: Lars Imsland

Co-advisor(s): Joakim Haugen, ITK, NTNU

Trondheim, __28.06.2011__________

Lars Imsland
Supervisor

Methods for Ice-Model Updating Using a Mobile Sensor Network

TTK4900 Master thesis

Author
Anne Mai Ersdal
June 30, 2011

Supervisor
Professor Lars Imsland
Co-advisor
Joakim Haugen

NTNU
Department of Engineering Cybernetics

Abstact

The need for dynamic positioning to function in ice-infested waters is growing as the offshore oil and gas industry enters the Arctic. The ice introduces great challenges, some of which can be resolved through proper ice management. This requires good knowledge of the surrounding ice-environment.

This thesis deals with the question of achieving a good state estimator for a sea-ice model. The dynamic thermodynamic sea-ice model of Hibler III (1979) is implemented, and it is shown through simulations that it reacts in a realistic manner to varying air temperature. The states of this model are estimated with an ensemble Kalman filter, and it is shown that different states can be estimated very well by ensemble Kalman filters based on different measurement configurations. This implemented nonlinear sea-ice model and state estimator is meant to serve as a platform where methods designed to select measurement configurations best suited for state estimation can be tested.

A suggestion for a method which chooses measurement configurations on-line is presented. The idea is that this method allows for different measurement configurations to be applied at different time steps, all based on which one that provides the best estimate at the current time. Unfortunately there was no time to implement this method and test it on the previously mentioned test platform; it must be kept in mind that it is merely a theoretical suggestion which must be further tested.

Acknowledgments

This thesis concludes my education within Master of Technology, Engineering Cybernetics at NTNU. It has been five challenging and hectic, but also fun and wonderful years which has though me a great deal, both academically and about life itself.

First of all, I would like to show my gratitude to my supervisor Professor Lars Imsland for all his encouragement and guidance in writing this thesis. I would like to thank him for always being available, and for always taking the time to answer all my questions and offering his advise. Thank you, it has been indispensable. I am also grateful to my co-advisor Joakim Haugen, for all the help, inputs and guidance he has offered, and also to Brage, Eirik, Jonas, Karsten, Tor Aksel and Øyvind for inputs, discussions, laughs and fellowship.

I would also like to thank my parents and sister for their unconditional support throughout my years at NTNU, and especially my father for introducing me to Engineering Cybernetics and for always helping me with my math homework.

Last, but not least, I want to thank Ørjan for always being there to support, encourage and comfort me, and for always being so helpful and understanding.

June 30, 2011
Anne Mai Ersdal

Contents

List of Figures

List of Tables

Nomenclature

Acronyms

DP	Dynamic positioning
EKF	Extended Kalman filter
EnKF	Ensemble Kalman filter
EOG	Empiric observability gramian
KF	linear Kalman filter
MDO	Measure for degree of observability
PDE	Partial differential equation
SFD	Spatial finite differences

Greek letters

$\boldsymbol{\beta}$		Thermodynamic changes of sea-ice states
$\boldsymbol{\gamma}$		Singular vectors
$\dot{\boldsymbol{\epsilon}}$	$[\mathrm{s}^{-1}]$	Sea-ice strain-rate tensor
ζ	$[\mathrm{N}/\mathrm{s}]$	Nonlinear bulk viscosity
η	$[\mathrm{N}/\mathrm{s}]$	Nonlinear shear viscosity
θ		Water turning angle
κ_i	$[\mathrm{W}/\mathrm{Km}]$	Thermal conductivity of ice
λ		Eigenvalues
μ_i		Measure for information about state i in output
ξ_A	$[\mathrm{s}^{-1}]$	Diffusion, ice compactness
ξ_h	$[\mathrm{M}/\mathrm{s}]$	Diffusion, ice thickness
ρ_a	$[\mathrm{kg}/\mathrm{m}^3]$	Air density
ρ_i	$[\mathrm{kg}/\mathrm{m}^3]$	Sea-ice density
ρ_w	$[\mathrm{kg}/\mathrm{m}^3]$	Sea-water density
$\boldsymbol{\sigma}$	$[\mathrm{N}]$	Sea-ice stress tensor

$\boldsymbol{\varsigma}$		Singular values
$\boldsymbol{\tau}_a$	[N]	Forces due to air stresses
$\boldsymbol{\tau}_w$	[N]	Forces due to water stresses
$\boldsymbol{\Upsilon}$	$[\mathrm{m/s}]$	Planar ice velocity
$\boldsymbol{\upsilon}$		Eigenvectors
ϕ		Air turning angle
$\boldsymbol{\chi}_w$		Boundary conditions
$\boldsymbol{\psi}$		Mechanical deformation of sea-ice states

Latin letters

A		Ice compactness
C_a		Air drag coefficient
C_w		Water drag coefficient
d		Cell size
D_1		Diffusion coefficient, harmonic diffusion
D_2		Diffusion coefficient, biharmonic diffusion
$\boldsymbol{e}_s$		Scaled estimate error
f	$[\mathrm{s}^{-1}]$	Coreolis parameter
F	[N]	Internal ice forces
g	$[\mathrm{m/s^2}]$	Gravity acceleration
h	[m]	Mean ice thickness
h_0	[m]	Mean ice thickness
H	[m]	Cutoff thickness, dividing thick and thin ice
$\boldsymbol{J}$		Sea-ice model states
$\boldsymbol{k}$		Unit vector normal to the sea surface
k_a	$[\mathrm{W/Km^2}]$	Heat exchange coefficient
K		Kalman gain
L	$[\mathrm{J/kg}]$	Latent heat of freezing
m	[kg]	Sea-ice mass
N		Ensemble member, EnKF
P	[N]	Sea-ice strength
$\boldsymbol{P}^-$		A priori Kalman filter covariance matrix
$\boldsymbol{P}^+$		A posteriori Kalman filter covariance matrix
Q_w	$[\mathrm{W/m^2}]$	Heat flux from sea water
$\boldsymbol{S}$		Scaling matrix
S_A	$[\mathrm{s}^{-1}]$	Thermodynamic term, ice compactness
S_h	$[\mathrm{m/s}]$	Thermodynamic term, ice thickness
T_a	[°C]	Air temperature
T_f	[°C]	Freezing point of sea water
$\boldsymbol{U}$	$[\mathrm{m/s}]$	Planar ice velocity

$\boldsymbol{U}_g$	[$^{\mathrm{m}}/_{\mathrm{s}}$]	Geostrophic wind
$\boldsymbol{U}_w$	[$^{\mathrm{m}}/_{\mathrm{s}}$]	Geostrophic ocean current
$\boldsymbol{u}$	[$^{\mathrm{m}}/_{\mathrm{s}}$]	Ice velocity, x-direction
$\boldsymbol{v}$	[$^{\mathrm{m}}/_{\mathrm{s}}$]	Ice velocity, y-direction
r		Ratio of principle axes
$\boldsymbol{v}$		Measurement noise
$\boldsymbol{w}$		Process noise
$\boldsymbol{W}_{o,linear}$		Observability gramain
$\boldsymbol{W}_o$		Empiric observability gramain
$\boldsymbol{x}$		System state vector
$\hat{\boldsymbol{x}}^-$		A priori estimate of system state vector
$\hat{\boldsymbol{x}}^+$		A posteriori estimate of system state vector
$\hat{\boldsymbol{X}}$		Ensemble estimate of system state vector

Chapter 1

Introduction

The need for Dynamic Positioning (DP) systems to function in ice-infested areas is rapidly growing, especially as the offshore oil and gas production enters the arctic seas. The presence of ice is a factor not included in DP systems nowadays, and this is a challenge which must be solved in order for DP to function in the presence of ice.

This chapter is an introduction to this topic, including a general presentation of DP systems and incentives for DP in ice-environments. Previous experience with DP in the Arctic will also be presented, along with a brief introduction to what will be addressed in this thesis and an overview of how the report is organized.

1.1 Dynamic positioning systems

A Dynamic Positioning system is a control system for station-keeping and low-speed maneuvering of vessels (Fossen, 2002). It was first tested in the 1970s, and has since then developed to become standard equipment on both large ships and oil rigs. The purpose of a DP system is to control the horizontal motion of a ship in surge, sway and yaw. It has traditionally been used to keep a ship stationary with fixed heading, or to move a ship from one location to another at low speed (Fossen, 2002). This is accomplished solely by the means of the ship's thrusters.

DP is also used for assisting mooring line position-keeping, known as thruster-assisted position mooring (PM) systems (Sørensen et al., 2001). Here, most of the position-keeping is done by attaching the ship to multiple fixed objects, for instance on the seabed, by so-called mooring lines. The DP system is employed for minimizing the tension on the mooring lines during severe environmental conditions.

The movement of a vessel in open water will be influenced by wind, waves and currents. The forces caused by these environmental disturbances are called *environmental loads*, and must be compensated for by the DP controller in order for the DP system to function properly. According to Fossen (2002), these are the most important design requirements for a DP controller to function under bad weather conditions.

There is a wide area of application for DP systems. Sørensen et al. (2001) define some of them as:

Offshore oil and gas industry
Offshore service vessels, drilling rigs, drilling ships, shuttle tankers, cable and pipe layers et cetera.

Shipping
Guidance systems coupled to automatic tracking control systems, automatic docking systems and precise positioning when operating in confined waters.

Cruise and yacht
Station keeping in areas where anchors are not allowed due to vulnerable coral reefs and precise positioning in confined waters and harbors.

Fisheries
Control of ships during fishing for precise positioning, reduced fuel consumption and intelligent selective fishing.

1.2 Dynamic positioning in the Arctic

The main incentive for the usage of DP in the Arctic is by far the oil and gas industry. The demand for oil and gas has grown rapidly over the past three decades, and it is still steadily increasing. As many of the major oil and gas reservoirs in the middle east, western Africa and the Gulf of Mexico already are producing at maximum level, some of them depleted, the need for discovering and exploiting new oil and gas reservoirs is crucial. Gautier et al. (2009) state that approximately 13% of the undiscovered oil reservoirs and 30% of the undiscovered gas reservoirs in the world are found in the

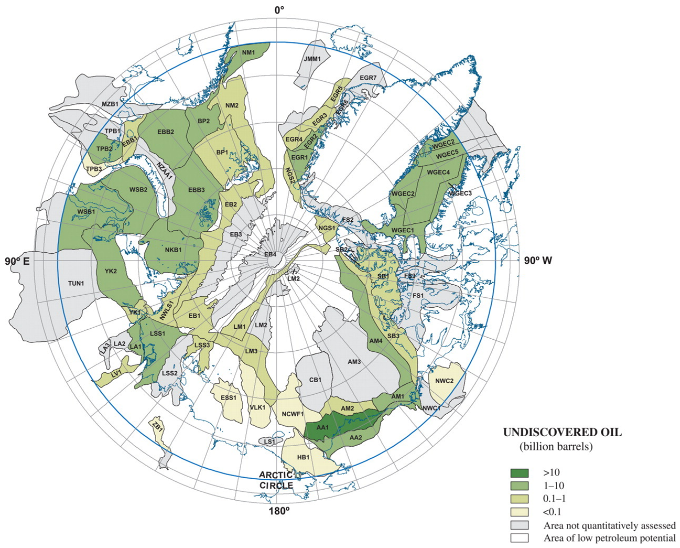


Figure 1.1: Undiscovered oil in the Arctic (Gautier et al., 2009).

Arctic, most of which are offshore. DP is a major part of recovering oil and gas from subsea reservoirs, during well drilling, installation of subsea equipment, geological surveys and so on. Figures 1.1 and 1.2 give a graphical representation of the mean estimated undiscovered oil and gas deposits in the Arctic.

Other compelling reasons for developing DP for Arctic conditions are short term operations, perhaps particularly for exploration and scientific means, and shipping activities. DP in ice covered seas as found in the Arctic is therefore of great current interest.

1.3 Previous experience with DP in Arctic regions

Over the last decade some successful DP operations in ice covered areas have been fulfilled. The first major DP operation was carried out in 1999 in the Sakhalin region in the Sea of Okhotsk off the east coast of Russia (Keinonen et al., 2000). Since then the oil and gas industry in this area has developed, and DP operations with ice breakers are currently taking place there. But the ice concentration in the Sakhalin region is quite different from the ice concentration in potential petroleum exploration areas. In it possible that the ice in these areas consist of multi year ice inclusion, whilst the ice in

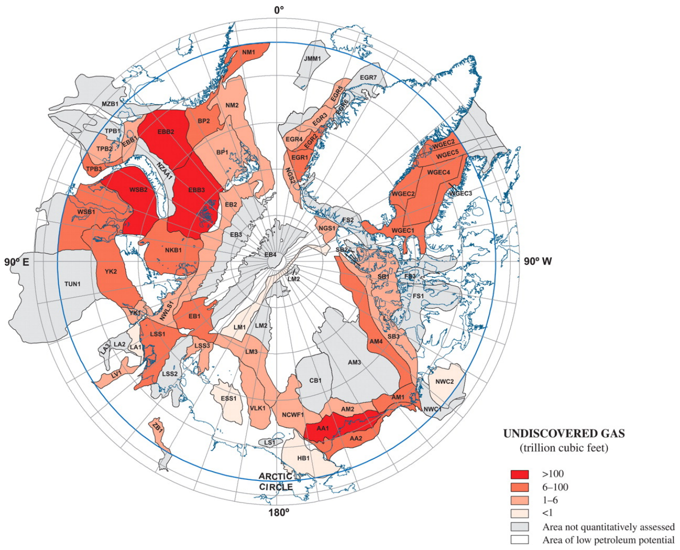


Figure 1.2: Undiscovered gas in the Arctic (Gautier et al., 2009).

the Sakhalin area consists of first year ice only (Jenssen et al., 2009). The multi-year ice inclusion is thicker and thus tougher to break.

In August 2004 Moran et al. (2006) performed drilling in the severely ice-covered Arctic Ocean north of Franz Josef Land. Figure 1.3 shows a map of Franz Josef Land and the Sea of Okhotsk; clearly Moran et al. (2006) were much deeper into the Arctic than Keinonen et al. (2000), and naturally they experienced tougher ice conditions. The Arctic Ocean-drilling succeeded and they managed to maintain the drill-ship on location continuously over many days (Moran et al., 2006). This was done by good *ice management*. Two ice-breakers were employed to break up the surrounding ice and to keep the drill-ship relatively free of ice. One broke the larger ice-floes into medium sized floes, and the other broke those into even smaller ones, which the drill-ship could handle. Another important part of the ice management was to monitor the sea ice and alert the crew at the drill-ship if ice loads too heavy to handle were approaching. In these cases the drilling had to be aborted until the ice conditions became manageable.

1.4 The contribution of this thesis

This thesis will address the task of estimating the ice environment surrounding a ship conducting DP operations in the Arctic. The problems related to

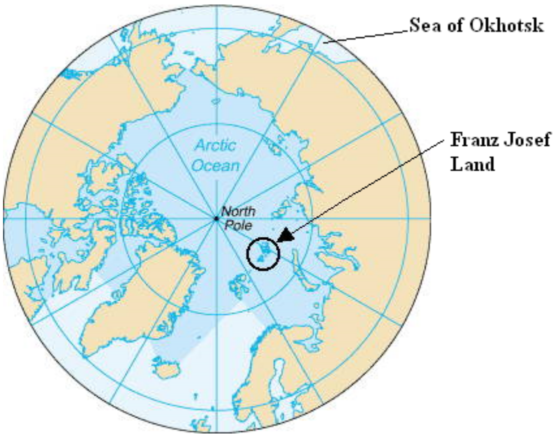


Figure 1.3: Map of Franz Josef Land and Sea of Okhotsk.

DP in ice environments, and why estimating the sea-ice is important is further discussed in Chapter 2, but it is quite intuitive that information about the surrounding ice is an important part of Arctic DP operations, both in terms of safety and carrying out the DP operation.

The scope of this thesis is not to present a complete solution to the problem of proper ice-environment estimation, but to implement a nonlinear sea-ice model and a state estimator, which can be applied as a platform for testing such solutions. A method for deciding which model measurements that leads to the best possible state estimate will also be investigated. This method can, if time permits, be tested on the implemented model and state estimator.

1.5 Report layout

This report is divided into a total of seven chapters. Chapter 2 discusses more in detail the challenges related to dynamic positioning in ice environments. The concept of a state estimator, and the role it might play in a solution to some of these challenges, are also presented here. Chapter 2 lays the foundation for understanding how this report may contribute to a solution to some of the challenges related to DP in the Arctic. The main content of both this chapter as well as Chapter 1 is based on the respective chapters of Ersdal (2010), since the work in this thesis partially builds on this project work.

The sea-ice model used in this thesis is presented in Chapter 3, along with an introduction to approaches for modeling sea ice. The implementation of the model is described and it is also verified through simulations. In Chapter 4 the Kalman filter, the extended Kalman filter and the ensemble Kalman filter are presented as state estimators. Two ensemble Kalman filters, based on two different measurement matrices are also designed and implemented.

Chapter 5 is dedicated to present a suggestion for a new method which can be used to select measurement configurations on-line, so that the best possible state estimate can be attained at each time step.

This concludes the work done in this thesis, and Chapter 6 discusses the results of Chapter 3, 4 and 5, while Chapter 7 contains the conclusion and also suggest further work on the topic.

Chapter 2

Dynamic Positioning in Ice Environments

This chapter considers the challenges related to dynamic positioning in ice environments, and how a state estimator may fit into a solution to these challenges. First, the general concepts of state estimators and observability are presented in Section 2.1.1, before the challenges related to DP in ice-environments and which of those challenges this report will focus, on are presented in Section 2.2. Finally, Section 2.3 describes how a state estimator may fit into a possible solution to these challenges, and why the measurement configuration is of importance.

As mentioned to begin with, this chapter is mainly based on Chapter 2 in Ersdal (2010). However, large parts of Section 2.1.1 and Section 2.3.1 are altered, and some figures are added.

2.1 State estimators

Given a nonlinear system

$$\dot{\boldsymbol{x}} = \boldsymbol{f}\left(\boldsymbol{x}, \boldsymbol{u}\right) \tag{2.1a}$$

$$\boldsymbol{y} = \boldsymbol{h}\left(\boldsymbol{x}\right) \tag{2.1b}$$

where $\boldsymbol{x} \in \mathbb{R}^n$ is a vector of state variables, $\dot{\boldsymbol{x}}$ is the state vector time derivative, $\boldsymbol{y} \in \mathbb{R}^m$ is a vector of measurements and $\boldsymbol{u} \in \mathbb{R}^p$ is the system input, it is not always given that every state is measurable. In fact, it seldom is. The state variables of a system are often not accessible for direct connection, and if they are, it is often expensive and/or time consuming to measure them all (Chen, 1999). In order to gain knowledge about the non-measured states, a state estimator must be designed. A state estimator is an algorithm that, based on the input $\boldsymbol{u}$ and the system measurements $\boldsymbol{y}$, generates an estimate of all the state variables of the system (Chen, 1999).

2.1.1 The concept of observability

It is not given that any arbitrary system's state variables can be estimated using a state estimator. Sometimes $\boldsymbol{y}$ simply does not provide enough information for the state estimator to propagate in a satisfying manner. The concept of *observability* studies the possibility of estimating the system states $\boldsymbol{x}$ given the system input $\boldsymbol{u}$ and output $\boldsymbol{y}$ (Chen, 1999). Chen (1999) defines observability for a linear system as follows, but the definition is also applicable for nonlinear systems.

> The state equation (2.2) is said to be observable if for any unknown initial state $\boldsymbol{x}(0)$, there exists a finite $t_1 > 0$ such that the knowledge of the input $\boldsymbol{u}$ and the output $\boldsymbol{y}$ over $[0, t_1]$ suffices to determine uniquely the initial state $\boldsymbol{x}(0)$. Otherwise, the equation is said to be unobservable.

Thus, if a system is observable a state estimator can be applied to estimate those system state variables which cannot be measured directly.

One way to determine whether a system is observable or not is through the system's *observability gramian*. The observability gramian of a linear system

$$\dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{u} \tag{2.2a}$$

$$\boldsymbol{y} = \boldsymbol{C}\boldsymbol{x} + \boldsymbol{D}\boldsymbol{u} \tag{2.2b}$$

where $\boldsymbol{x}$, $\boldsymbol{y}$ and $\boldsymbol{u}$ are defined as for (2.1), is given as

$$\boldsymbol{W}_{o,linear} = \int_0^\infty e^{\boldsymbol{A}^T\tau}\boldsymbol{C}^T\boldsymbol{C}e^{\boldsymbol{A}\tau}d\tau \tag{2.3}$$

and the system (2.2) is observable if and only if $\boldsymbol{W}_{o,linear}$ is nonsingular (is of full rank) (Chen, 1999).

For nonlinear systems, the conditions for observability are not as clear. During the past two decades, many have tried to derive conditions for the observability of nonlinear systems, resulting in conditions too complex to be useful for all but very simple systems (Singh and Hahn, 2005). One alternative to these complex conditions is to use the *empirical observability gramian* (EOG) (Singh and Hahn, 2005). As the name implies, the empirical observability gramian is calculated from simulated data, and is thus an empirical approximation of the observability gramian.

The EOG is found trough perturbation around (one of) the systems's steady state(s), and it is based on the different output-trajectories generated by each perturbation. The following matrices are defined before the definition of the EOG is presented:

$$\boldsymbol{T} = \left\{\boldsymbol{T}_1, \cdots, \boldsymbol{T}_r;\ \boldsymbol{T}_i \in \mathbb{R}^{n\times n},\ \boldsymbol{T}_i^T\boldsymbol{T}_i = \boldsymbol{I},\ i = 1, \cdots, r\right\} \tag{2.4}$$

$$\boldsymbol{M} = \{c_1, \cdots, c_s;\ c_i \in \mathbb{R},\ c_i > 0,\ i = 1, \cdots, s\} \tag{2.5}$$

$$\boldsymbol{E} = \{e_1, \cdots, e_n;\ \text{standard unit vectors in } \mathbb{R}^n\} \tag{2.6}$$

where r is the number of matrices for the perturbation directions, s is the number of different perturbation sizes for each direction, and n is the number of states in the system (Singh and Hahn, 2005). The largest perturbation size should be chosen so that the system stays within the region of attraction of the steady state (Singh and Hahn, 2005). Given this, the EOG for the stable, nonlinear system (2.1) is defined as (Singh and Hahn, 2005)

$$\boldsymbol{W}_o = \sum_{l=1}^{r}\sum_{m=1}^{s}\frac{1}{rsc_m^2}\int_0^\infty \boldsymbol{T}_l\boldsymbol{\Psi}^{lm}(t)\boldsymbol{T}_l^T dt, \tag{2.7}$$

where $\boldsymbol{\Psi}^{lm}(t) \in \mathbb{R}^{n\times n}$ equals

$$\boldsymbol{\Psi}^{lm}(t) = \left(\boldsymbol{y}^{ilm}(t) - \boldsymbol{y}_{ss}\right)^T \left(\boldsymbol{y}^{ilm}(t) - \boldsymbol{y}_{ss}\right), \tag{2.8}$$

and $\boldsymbol{y}^{ilm}(t)$ is the output trajectory of the system corresponding to the initial condition $\boldsymbol{x}_0 = \boldsymbol{x}_{ss} + c_m\boldsymbol{T}_l e_i$. The vectors $\boldsymbol{x}_{ss}$ and $\boldsymbol{y}_{ss}$ are the steady state and steady-state output of the system, respectively. If the EOG of system (2.1) is nonlinear (has full rank), then the system is observable.

Since it is hardly possible to generate output trajectories lasting into infinity, the simulations should be carried until the system has reached steady state and $\boldsymbol{\Psi}^{lm}(t) \approx 0$.

It is important to notice that the output-trajectories are generated using the nominal value of the system inputs, and the EOG is therefore only applicable for nominal values of the inputs around the steady state $\boldsymbol{x}_{ss}$.

2.1.2 The concept of degree of observability

There are examples in the literature where the observability gramian, both for linear and nonlinear systems, have been used to define measures for *degree of observability*. A measure for degree of observability is a scalar measure beyond simply observable/non-observable, and they are used to rank different measurement configurations who all result in observable systems against each other.

The idea is that the higher the degree of observability for a system with a given measurement configuration is, the more process information lies in those measurements. Hence, a state estimator based on a measurement configuration resulting in a system with higher degree of observability will provide a better estimate than one based on a measurement configurations resulting in a system with lower degree of observability.

Different measures for degree of observability (MDO) based on the observability gramian were discussed in Ersdal (2010). Here, linear systems were discussed, but the MDOs are applicable for nonlinear systems and empirical observability gramians as well.

2.1.3 Determining the output, y

Since the system dynamics $\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u})$ are given and cannot be altered, it is the system measurement configuration $\boldsymbol{y} = \boldsymbol{h}(\boldsymbol{x})$ which can be changed in order to obtain the best (in one respect or another) state estimator. Since $\boldsymbol{h}(\boldsymbol{x})$ states which state variables that are being measured, it can be changed by altering the measurement configurations, and the key to a satisfying state estimator is to choose a measurement configuration which provides the most information possible about the state variables. Obviously, there are many constraints to the choice of $\boldsymbol{h}(\boldsymbol{x})$, ranging from access to state variables to economic questions. These constraints must of course be taken into consideration when determining $\boldsymbol{h}(\boldsymbol{x})$, but they will not be further elaborated in this thesis.

2.2 Challenges related to DP in ice

The environmental conditions in the Arctic are very different from those found in open waters where DP is applied today. This leads to a number

Figure 2.1: Underwater view of the hull before and after it's been redesigned (Jenssen et al., 2009).

of challenges when it comes to performing operations in these waters. They include everything from the effects of low temperature (e.g., formation of ice on deck and various instruments) to atmospheric phenomena as darkness, fog and precipitation (Bonnemaire et al., 2007). However, the single most important challenge when it comes to station keeping in the Arctic is the ice environment.

The ice environment in the Arctic leads to several challenges when performing DP operation. One is due to the varying ice-drift direction. Wind, current and tidal influences will in the Arctic often lead to a circular ice drift direction-pattern (Bonnemaire et al., 2007). One of the schemes for station keeping in the Arctic is to keep the vessel heading directly towards the ice drift. Due to the varying ice drift direction, the vessel will constantly rotate in the same direction and special ice management schemes must be employed in order to avoid wind up of umbilicals and choking line hoses (Jenssen et al., 2009).

As ice hits the vessel, it will often be broken downwards and forced under the it. This causes a second challenge: ice being forced under the vessel could easily damage equipment mounted on the hull. The vessel must therefore be designed with "built-in ice management characteristics" (Jenssen et al., 2009). This could include properties such as proper hull design and the use of azimuth thrusters and riser systems to effectively remove the ice prior to being led under water (Bonnemaire et al., 2007). Jenssen et al. (2009) showed what an improvement that good hull design may lead to, with less ice floes being forced under water, see Figure 2.1.

The presence of ice also adds another dimension to the environmental loads. The average ice forces acting on the vessel can at times be very high, both in surge and sway, as ice builds up around the hull (Jenssen et al., 2009). In addition to this, there are occurrences of force peaks as larger ice floes hit the hull at high speed. The magnitude of the ice load is decided by factors such as ice drift velocity, ice dimensions, ice properties and size-distribution of the ice floes. Knowledge of these forces are essential for being able to carry out the dynamic positioning, both in terms of station keeping and with respect to safety. Sometimes the ice load will become too severe, and the vessel will not be able to keep its position. This must be known in advance, so that the operation can be put on hold and the vessel may drift with the ice until the conditions improve.

2.3 The use of state estimators within DP in the Arctic

In Section 2.2 several challenges related to carrying out DP in an ice environment were mentioned. This report addresses the last challenge listed, namely that an ice environment adds another dimension to the environmental loads acting on the vessel. However, the ice load can be very difficult to estimate. The ice fields have their own dynamics which again is coupled with the vessel dynamics, and in order to determine the forces acting from the ice on the vessel, an ice-vessel interaction model is needed. Deriving such a complex, coupled model is a topic of current research, and will not be within the scope of this thesis. Here, the vessel is disregarded, and the well known dynamic thermodynamic sea-ice model of Hibler III (1979) will be applied. In this model the vessel itself is not included, and the focus will be on the ice dynamics only.

It is crucial, both to safety and to the station keeping that knowledge about the upcoming ice loads exists so that proper ice management may be executed. Eik (2010) defines ice management in the following manner:

> Ice management is the sum of all activities where the objective is to reduce or avoid actions from any kind of ice features. This will include, but is not limited to:
>
> - Detection, tracking and forecasting of sea ice, ice ridges and icebergs.
>
> - Threat evaluation.

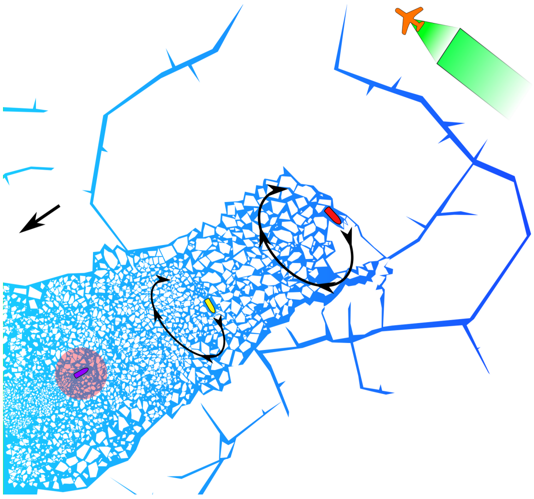


Figure 2.2: Ice management with the drill ship, UAV and two ice breakers (Arctic DP operation, courtesy of Joakim Haugen).

- Physical ice management such as ice breaking and iceberg towing.
- Procedures for disconnection of offshore structures applied in search for or production of hydrocarbons.

In this report, it is the first point listed above which is the main focus, and a state estimator based on a good sea-ice model could be very helpful in this matter. If all the state variables can be estimated accurately, this information can be used both to optimize the ice management and to carry out the station keeping in a satisfying manner. Ice management is crucial for DP to function in ice-covered waters, and with an accurate state estimator of the ice movement in the surrounding waters, one could in a better way plan and carry out the ice management. Figure 2.2 shows a possible ice management scheme. The physical ice management is carried out by two ice breakers. One breaks the larger ice-floes into medium sized floes, and the other one breaks them into even smaller pieces, which the drill-ship can handle. An *unmanned aerial vehicle* (UAV) making ice measurements is also depicted.

Some of the factors determining the ice load from an ice field are given in Section 2.2, and they are typically ice-drift velocity, ice coverage, ice properties and ice thickness. Most sea-ice models will most likely include some of these factors in their state variables, and they will therefore be estimated by a state estimator based on these models. Incorporating this information into the ice

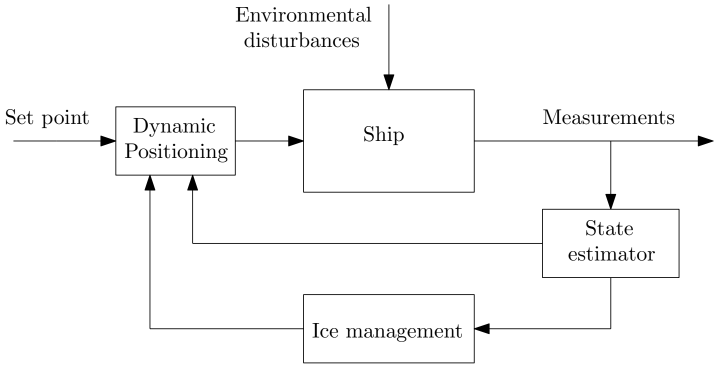


Figure 2.3: Control loop with ship, DP controller, state estimator and ice management.

management will improve the efficiency of the DP operation as a whole. Figure 2.3 shows the DP control loop including the ship, DP controller, state estimator and ice management. The set point is the DP operating point or desired trajectory and the environmental disturbances include wind, waves, currents and ice environment.

2.3.1 Ice system measurements

As mentioned in Section 2.1.3, the state-estimator designers can alter the measurement configurations by determining which state variables they want to measure. If the measurements are made using a *mobile sensor network*, different measurements can be taken at different times, and which of the possible measurement configurations that provides the best information about the system at the current time becomes an interesting question.

In addition to summarizing different observability-gramian based MDOs, Ersdal (2010) discussed methods based on these MDOs for selecting the measurement combination that results in the best state estimator for linear systems. The sea-ice model ant state estimator which is implemented in this thesis can be used to test methods such as these. A new method based partially on Ersdal (2010) is also investigated here.

2.4 Chapter summary

It has been seen in this chapter that the presence of ice may lead to several challenges when it comes to carrying out DP operations in the Arctic. The ones mentioned here are

- Damages to line hoses et cetera due to the vessel's constant rotation following the circular ice-drift direction pattern,
- sea ice being forced under the vessel, possibly causing damage to equipment mounted on the hull,
- and the extra dimension added to the environmental loads which must be compensated for by the DP controller.

It is the last challenge listed which is focused on in this thesis.

The general state estimator was also presented, and the observability gramian was introduced as a tool for determining whether a system is observable or not, and for deciding its degree of observability. How a state estimator may fit into a solution to the challenge of determining the extra environmental load has also been evaluated.

Chapter 3

The Sea-Ice Model

This chapter deals with the sea-ice model which is applied in this thesis. A brief overview of the different features characterizing an ice environment is presented in Section 3.1, before an introduction to different approaches to modeling of sea ice is given in Section 3.2. In Section 3.3 the dynamic thermodynamic sea-ice model of Hibler III (1979) is presented, and the implementation and validation of it is described in Section 3.4 and 3.5, respectively.

The Hibler-model is well established and has become the standard sea-ice dynamics model (Hunke and Dukowicz, 1997). It is widely used for ice simulations, and it was chosen because it is relatively simple while still being complex enough for the scope of this thesis.

3.1 Ice environment characteristics

Roughly, one can say that an ice environment is characterized by *ice features* and *ice properties* (Løset et al., 2006), see Figure 3.1. Ice properties concern the structure of the ice itself, e.g. porosity, salinity and crystallography, while ice features describe the type of ice present. The ice may take on many different forms depending on the physical processes the ice undergoes after formation (Løset et al., 2006). *Level ice* is ice which is unaffected by deformation, *rafted ice* is deformed ice where the floes are overriding one another, *ridges* are lines of broken ice forced up by pressure, *rubble* ice is a random collection of ice fragments or small pieces of ice covering a large area and *icebergs* are massive pieces of land ice floating at sea (Løset et al., 2006). These different ice features are again characterized by factors such

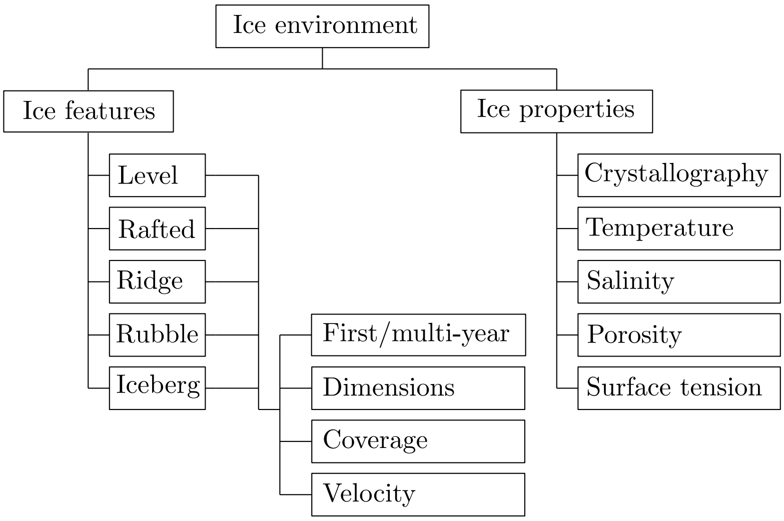


Figure 3.1: Overview of main features of an ice environment, recreated from Figure 5.1.1 in Løset et al. (2006).

as the percentage of ice cover, if it consists of first year or multi-year ice, ice-floe dimensions and ice-floe velocity. Figure 3.2 includes pictures of some ice features.

3.2 Approaches to modeling of ice

In the 1960s, numerical modeling of sea ice dynamics began in the Soviet union and in the USA (Leppäranta, 2005). The sea-ice models started out as quite simple, linear models, but by the end of the 1970s the modern theory of nonlinear sea-ice dynamics was developed. Since then, coupled ice-ocean models have been developed and the spatial resolution has improved. Today, these models are mainly used for short-term ice forecasting, weather forecasting and long-term simulation of climate change (Leppäranta, 2005).

3.2.1 Scales and continuum vs discrete models

When modeling sea ice, there are two dimensions to consider: whether to use a *discrete* or a *continuum* model, and which *scale* this model should consider. In discrete models each individual ice floe is modeled as a “particle”, and the ice environment as a whole is modeled as a granular assembly (Heinonen, 2004). The shape, the size and the speed of each “particle” is approximated,

(a) Ice ridge (b) Rubbled ice (c) Iceberg

Figure 3.2: Some ice features.

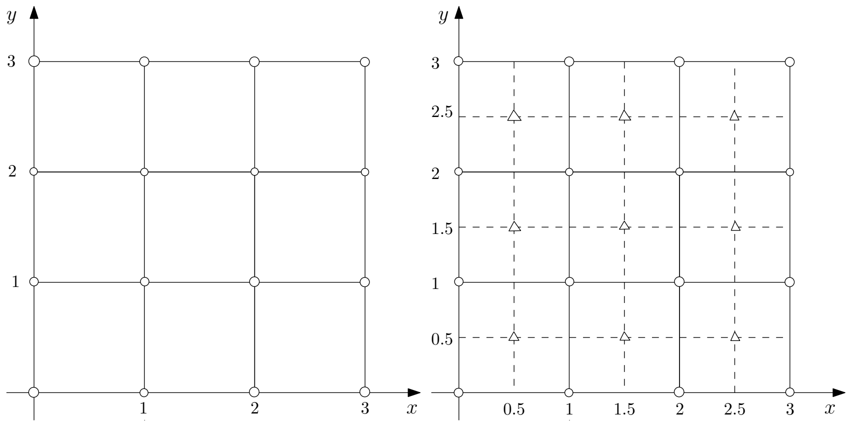
and the interaction between them is given through contact forces and relative displacements (Heinonen, 2004). While discrete models view each ice floe as an individual unit, the continuum models do not share the same level of detail. The latter considers the behavior of a volume of ice in an average sense, where the number of ice floes in this volume and their particular properties are unknown, and where the interaction between the ice is described by continuous stress and strain functions (Heinonen, 2004).

The other dimension to consider is, as mentioned, the scaling of the model. When modeling sea ice there is a number of different scales one can consider, dependent on the desired level of detail (Leppäranta, 2005):

- *Microscale*: When investigating individual ice particles from submillimeters to 0.1 m
- *Local scale*: Extending from $0.1 - 10$ m. When one does not care about the ice particles, and the sea ice is considered to be solid.
- *Ice-floe scale*: Reaching from 10 m to 10 km. Here, individual ice floes and different ice features are included.
- *Mesoscale*: Operating at 100 km, regarding larger areas of ice environments.
- *Large scale*: Operating at 1000 km, regarding even larger areas of ice environments.

When considering scales as mesoscale and large scale, the ice medium is referred to as *drift ice* or *pack ice* (Leppäranta, 2005).

The discrete models are often more physically correct, but for modeling larger ice environments they usually become too complex, and difficulties arise when the geometry of the internal structures and the contact phenomena are to be



(a) Unstaggered grid (b) Staggered grid

Figure 3.3: Unstaggered and staggered grid. In the unstaggered grid every property is calculated at the circles, in the staggered grid some properties are calculated at the circles and some at the triangles.

modeled accurately (Heinonen, 2004). So for larger scales, such as mesoscale and large scale, all workable dynamic sea-ice models are continuum models (Leppäranta, 2005). Ice management is often carried out on a mesoscale or large-scale level, and it is therefore the continuum models which are discussed in the following.

3.2.2 Grids and Eulerian/Lagrangian frames

When implementing a continuum model, the area of interest is covered by an imaginary grid, containing multiple grid cells. The idea is that every attribute is constant within one grid cell, and that the adjacent cells affect each others properties. In practice, most sea-ice models use a rectangular grid, but in theory it may have any geometry (Leppäranta, 2005). Grids may also differ with regards to which grid points the different ice properties are calculated at. The very first models calculated every attribute in the same grid points (Leppäranta, 2005), which is known as an *unstaggered* or *Arakawa A*-grid. The most common grid today is the *staggered* or *Arakawa B*-grid, which evaluates different properties at two different grid points (Leppäranta, 2005). Figure 3.3 illustrates these two types of grid.

Another choice to be made when developing a continuum model is whether to use an Eulerian or a Lagrangian frame. With an Eulerian frame, the grid is fixed in space and with a Lagrangian frame the grid is fixed in the medium, i.e., it moves with the sea ice. It is the Eulerian frame that has gained most popularity within sea ice dynamic models, but for some areas of application, such as when modeling ice drift, the Lagrangian frame is more applicable.

3.2.3 System of equations

Leppäranta (2005) states that a full sea ice model consists of four basic elements: The sea-ice state(s), the sea-ice rheology, conservation of momentum and conservation of ice. The sea ice states

$$\boldsymbol{J} = \begin{bmatrix} J_1 & J_2 & J_3 & \cdots \end{bmatrix}^T \tag{3.1}$$

are model properties such as ice floe diameter, ice thickness, et cetera. Rheology is the study of the flow of matter, and it examines how the stress in a medium depends on its material properties and strain (Leppäranta, 2005). When one speaks of the sea-ice rheology in a model, it is thus a function describing the sea-ice stress $\boldsymbol{\sigma}$ given by the sea-ice state $\boldsymbol{J}$, the sea-ice strain $\boldsymbol{\epsilon}$ and the sea-ice strain rate $\dot{\boldsymbol{\epsilon}}$.

$$\boldsymbol{\sigma} = \boldsymbol{\sigma}(\boldsymbol{J}, \boldsymbol{\epsilon}, \dot{\boldsymbol{\epsilon}}) \tag{3.2}$$

The basic models for rheology are linear elastic (assumes that stress is proportional to strain), linear viscous (assumes that stress in proportional to strain rate) and ideal plastic (collapses once stress stress achieves yield strength) (Leppäranta, 2005).

The conservation of momentum is an equation which contains the derivative of the ice velocity $\boldsymbol{U}$

$$\frac{D\boldsymbol{U}}{Dt} = f(\boldsymbol{U}, \boldsymbol{\sigma}, \boldsymbol{\tau}) \tag{3.3}$$

where $\boldsymbol{\tau}$ are external forces acting on the ice, and the equation for conservation of ice contains the derivatives of the ice states

$$\frac{J\boldsymbol{D}}{Dt} = \boldsymbol{\psi} + \boldsymbol{\beta} \tag{3.4}$$

where $\boldsymbol{\psi}$ is the mechanical deformation and $\boldsymbol{\beta}$ the thermodynamic changes (Leppäranta, 2005). These equations therefore work as update equations for the ice velocity and ice states.

3.2.4 The finite element and finite difference approaches

When solving these partial differential equations (PDE), one must also decide which numerical technique to apply when approximating the spatial derivatives: *the finite element* or *the finite difference* approach. While the finite difference method approximates the solutions to differential equations by replacing the derivatives with approximately equivalent difference expressions (Süli and Mayers, 2004), the finite element methods either eliminates the differential equation from the problem, or converts the PDE into an approximating system of ordinary differential equations (Strikwerda, 2003).

3.3 The Hibler-model

The model chosen to be implemented in this thesis is the model described by Hibler III (1979). It is a continuum model with a square staggered grid and Eulerian frame, which uses the finite difference approach to approximate the spatial derivatives. It is a widely used model, and many present sea-ice models are based on it (Leppäranta, 2005). The Hibler-model mainly consists of four different components:

The momentum balance
Describes the derivative of the ice velocity. It includes air and water stresses, Coriolis force, internal forces and ocean tilt.

The ice strength
Describes the ice strength as a function of the ice thickness and fraction of open water.

The constitutive law
Describes the nature of the ice interaction and the ice rheology. It relates the ice stress to the strain rate and the ice strength, which again can be related to the internal forces in the momentum balance.

The ice-thickness distribution
Describes the derivatives of the thickness and compactness of ice. It includes growth or ablation of ice as well as advection and deformation.

These components are related to (3.1) - (3.4) in the following manner: The sea ice states $\boldsymbol{J}$ of (3.1) are the ice thickness h and the sea ice compactness A, i.e., $\boldsymbol{J} = \begin{bmatrix} h & A \end{bmatrix}^T$. The constitutive law and the ice strength equation are related to the ice rheology and fits in under equation (3.2), while the ice thickness distribution is the equations for conservation of mass (3.4) and the momentum balance is the conservation of momentum (3.3).

The equations related to each of the components listed above will now be described in detail, and then a summary of the cell properties this system of equations requires will be given.

3.3.1 The momentum balance

The momentum balance is a two-dimensional equation given by

$$m\frac{D\boldsymbol{U}}{Dt} = -mf\boldsymbol{k}\times\boldsymbol{U} + \boldsymbol{\tau}_a + \boldsymbol{\tau}_w - mg\nabla H + \boldsymbol{F} \tag{3.5a}$$

$$\frac{D\boldsymbol{U}}{Dt} = \frac{\partial\boldsymbol{U}}{\partial t} + \boldsymbol{U}\nabla\boldsymbol{U} \tag{3.5b}$$

where $\boldsymbol{U} = \begin{bmatrix} u & v \end{bmatrix}^T$ is the planar ice velocity [$^{\mathrm{m}}/_{\mathrm{s}}$] and equation (3.5b) the material derivative [$^{\mathrm{m}}/_{\mathrm{s}^2}$]. m the ice mass per unit area [$^{\mathrm{kg}}/_{\mathrm{m}^2}$], f the Coreolis parameter [s^{-1}], and $\boldsymbol{k}$ a unit vector normal to the sea surface [$-$]. $\boldsymbol{\tau}_a$ and $\boldsymbol{\tau}_w$ are forces per unit area due to air and water stresses [$^{\mathrm{N}}/_{\mathrm{m}^2}$], H the sea surface dynamic height [m], g acceleration due to gravity [$^{\mathrm{m}}/_{\mathrm{s}^2}$], and $\boldsymbol{F}$ the internal ice forces per unit area [$^{\mathrm{N}}/_{\mathrm{m}^2}$]. In this thesis it is assumed that H is constant over the entire simulation grid, and $mg\nabla H$ therefore equals zero.

The forces due to air and water stresses are given as

$$\boldsymbol{\tau}_a = \rho_a C_a |\boldsymbol{U}_g| \Big(\boldsymbol{U}_g \cos(\phi) + \boldsymbol{k}\times\boldsymbol{U}_g \sin(\phi)\Big) \tag{3.6}$$

$$\boldsymbol{\tau}_w = \rho_w C_w |\boldsymbol{U}_w - \boldsymbol{U}| \Big((\boldsymbol{U}_w - \boldsymbol{U})\cos(\theta) + \boldsymbol{k}\times(\boldsymbol{U}_w - \boldsymbol{U})\sin(\theta)\Big) \tag{3.7}$$

$\boldsymbol{U}_g$ is the geostrophic wind [$^{\mathrm{m}}/_{\mathrm{s}}$], $\boldsymbol{U}_w$ the geostrophic ocean current[1] [$^{\mathrm{m}}/_{\mathrm{s}}$], C_a and C_w air and water drag coefficients [$-$], ρ_a and ρ_w air and water densities [$^{\mathrm{kg}}/_{\mathrm{m}^3}$], and ϕ and θ air and water turning angles [$-$], respectively. Both ϕ and θ are assumed to be constant. Because $\boldsymbol{U}_g \gg \boldsymbol{U}$, the ice motion is neglected in (3.6).

---

[1] The geostrophic wind and current are theoretical values which would form if there was an exact balance between the Coriolis force and the pressure gradient force (Holton, 2004).

3.3.2 The ice strength

The ice strength P [N] is coupled to the ice thickness through the following equation

$$P = P^{\star} h e^{-C(1-A)} \tag{3.8}$$

where h is the mean ice thickness of the cell [m], $0 \leq A \leq 1$ the ice compactness [$-$], and $P^{\star}$ [N/m] and C [$-$] are fixed empirical constants. This formulation makes the ice strength strongly dependent on the amount of thick ice present (through A), as well as allowing it to increase as this thick ice becomes thicker.

3.3.3 The constitutive law

The constitutive law describes the nature of the ice interaction. This is modeled by means of the relationship between the ice-stress tensor and the ice strain-rate tensor. The ice-stress tensor of rank two $\boldsymbol{\sigma}$ [N] is a matrix containing the intensity of the internal forces acting within the ice, and the ice strain-rate tensor of rank two $\dot{\boldsymbol{\epsilon}}$ [s^{-1}] is a matrix containing the ice deformation rate (Riley et al., 2007). For the two-dimensional case, $\boldsymbol{\sigma}$ and $\dot{\boldsymbol{\epsilon}}$ are defined as

$$\boldsymbol{\sigma} = \begin{bmatrix} \sigma_{xx} & \sigma_{xy} \\ \sigma_{yx} & \sigma_{yy} \end{bmatrix} \quad \text{and} \quad \dot{\boldsymbol{\epsilon}} = \begin{bmatrix} \dot{\epsilon}_{xx} & \dot{\epsilon}_{xy} \\ \dot{\epsilon}_{yx} & \dot{\epsilon}_{yy} \end{bmatrix} \tag{3.9}$$

where

$$\dot{\epsilon}_{xx} = \tfrac{\partial u}{\partial x}, \qquad \dot{\epsilon}_{yy} = \tfrac{\partial v}{\partial y} \quad \text{and} \quad \dot{\epsilon}_{yx} = \dot{\epsilon}_{xy} = \tfrac{1}{2}\left(\tfrac{\partial u}{\partial y} + \tfrac{\partial v}{\partial x}\right). \tag{3.10}$$

Hibler III (1979) considers the ice to be a nonlinear, viscous, and compressible fluid, which leads to the following relationship between σ_{kl} and $\dot{\epsilon}_{kl}$ ($k \in \{x, y\}$, $l \in \{x, y\}$)

$$\sigma_{kl} = 2\eta\dot{\epsilon}_{kl} + (\zeta - \eta)\left(\dot{\epsilon}_{xx} + \dot{\epsilon}_{yy}\right)\delta_{kl} - \frac{P}{2}\delta_{kl} \tag{3.11}$$

where δ_{kl} is the *Kronecker delta*

$$\delta_{kl} = \begin{cases} 1 & \text{if } k = l \\ 0 & \text{if } k \neq l \end{cases} \tag{3.12}$$

P is the ice strength, ζ the nonlinear bulk viscosity, and η the nonlinear shear viscosity. The nonlinear viscosities are given as functions of P and $\dot{\boldsymbol{\epsilon}}$.

These functions are set so that the stress state lies on an elliptical yield curve passing through the origin:

$$\zeta = \frac{P}{2\Delta} \tag{3.13}$$

$$\eta = \frac{\zeta}{r^2} \tag{3.14}$$

where

$$\Delta = \sqrt{\left(\dot{\epsilon}_{xx}^2 + \dot{\epsilon}_{yy}^2\right)\left(1 + \frac{1}{r^2}\right) + \frac{4}{r^2}\dot{\epsilon}_{yx}^2 + 2\dot{\epsilon}_{x}^2\dot{\epsilon}_{yy}^2\left(1 - \frac{1}{r^2}\right)} \tag{3.15}$$

and r is the ratio of principal axes of the ellipse. For more details on this ellipse, see Hibler III (1979).

The internal forces $\boldsymbol{F}$ of the momentum balance (3.5) are given by (Leppäranta, 2005)

$$\boldsymbol{F} = \nabla \boldsymbol{\sigma} \tag{3.16}$$

which are equal to

$$\begin{aligned} F_x &= \frac{\partial \sigma_{xx}}{\partial x} + \frac{\partial \sigma_{xy}}{\partial y} \\ &= \frac{\partial}{\partial x}\left[(\eta + \zeta)\frac{\partial u}{\partial x} + (\zeta - \eta)\frac{\partial v}{\partial y} - \frac{1}{2}P\right] + \frac{\partial}{\partial y}\left[\eta\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right)\right] && (3.17) \\ F_y &= \frac{\partial \sigma_{yy}}{\partial y} + \frac{\partial \sigma_{yx}}{\partial x} \\ &= \frac{\partial}{\partial y}\left[(\eta + \zeta)\frac{\partial v}{\partial y} + (\zeta - \eta)\frac{\partial u}{\partial x} - \frac{1}{2}P\right] + \frac{\partial}{\partial x}\left[\eta\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right)\right] && (3.18) \end{aligned}$$

Hence, the internal forces acting within the ice are given as functions of the ice planar velocity $\boldsymbol{U}$, the ice strength P as well as the ice strain-rate tensor $\boldsymbol{\epsilon}$.

3.3.4 The ice thickness distribution

The sea-ice cover will typically consist of several ice thicknesses, and in a given region, fractions of both thin and thick ice may be present at the same time. In order to approximately parameterize this variable thickness ice cover, Hibler III (1979) introduces two idealized thickness levels: thick and thin. As long as the ice thickness is less than some cutoff thickness h_0

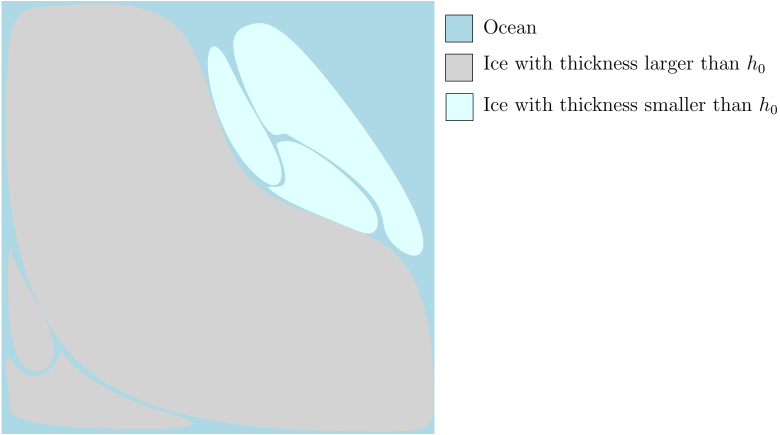


Figure 3.4: An example of ice-thickness distribution in a random cell. In this case $A \approx 0.75$ and h is the mean thickness of the gray ice if it was distributed over the entire area.

(and falls under the category thin ice), it is considered as being open water. The amount of open water in a region is given by the compactness A, which is the percentage of an area that is covered by thick ice. The thick ice is ice with thickness larger than h_0, and it is considered to be of equal thickness h. This is the mean thickness that the thick ice would have if it covered the entire area. Figure 3.4 illustrates the relationship between the two idealized thickness levels and the compactness A. The light blue area, representing ice with thickness smaller than h_0, is not thick enough to be considered when calculating A. A only represents the gray ice with thickness larger than h_0, and therefore becomes approximately 0.75.

Thus there are two variables which describe the ice-thickness distribution: the compactness A and the mean thickness h. The following continuity equations are used in updating them:

$$\frac{\partial h}{\partial t} = -\frac{\partial (uh)}{\partial x} - \frac{\partial (vh)}{\partial y} + S_h + \xi_h \tag{3.19}$$

$$\frac{\partial A}{\partial t} = -\frac{\partial (uA)}{\partial x} - \frac{\partial (vA)}{\partial y} + S_A + \xi_A \tag{3.20}$$

Here $A \leq 1$ and S_h and S_A are thermodynamic terms given as

$$S_h = g\left(h/A\right) A + (1-A)\, g(0) \tag{3.21}$$

$$S_A = \begin{cases} \frac{g(0)}{h}(1-A) & \text{if} \quad g(0) > 0 \\ 0 & \text{if} \quad g(0) < 0 \end{cases} + \begin{cases} 0 & \text{if} \quad S_h > 0 \\ \frac{A}{2h} S_h & \text{if} \quad S_h < 0 \end{cases} \tag{3.22}$$

where $g(h)$ is the growth rate of ice with thickness h. Even though the growth rate is represented as a function of h here, it is also dependent on the air temperature. Hibler III (1979) uses different growth-rate functions for each month, thereby including the air temperature as a monthly varying parameter. This is accurate enough when simulating for several years, as was done in Hibler III (1979), but when simulating for about one or two weeks it will probably be beneficial to include a more accurate ice growth-rate function.

If one includes the ice-atmosphere coupling and the heat flux from the water, but excludes the effect of snow, Leppäranta (1993) presents the following expression for ice growth rate. It is a function of both the ice thickness h and the air temperature T_a:

$$g(h, T_a) = \frac{1}{\rho_i L}\left[\frac{\kappa_i}{h + \kappa_i/k_a}(T_f - T_a) - Q_w\right] \tag{3.23}$$

where ρ_i is the ice density [$\mathrm{kg/m^3}$], L the latent heat of freezing [$\mathrm{J/kg}$], κ_i the thermal conductivity of ice [$\mathrm{W/Km}$], k_a a heat exchange coefficient [$\mathrm{W/Km^2}$], T_f the freezing point of sea water [°C] and Q_w the heat flux from the water [$\mathrm{W/m^2}$].

The terms ξ_h and ξ_A are diffusion terms, and consist of both harmonic and biharmonic diffusion. They are small, and have only been added for numerical stability:

$$\xi_h = D_1\nabla^2 h - D_2\nabla^4 h \tag{3.24}$$

$$\xi_A = D_1\nabla^2 A - D_2\nabla^4 A \tag{3.25}$$

where

$$\nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \tag{3.26}$$

$$\nabla^4 = \frac{\partial^4}{\partial x^4} + \frac{\partial^4}{\partial y^4} + 2\frac{\partial^4}{\partial x^2 y^2} \tag{3.27}$$

and D_1 and D_2 are constant diffusion coefficients.

In Hibler III (1979) it seems that both the harmonic and the biharmonic diffusion terms are positive. The reason that the latter is set to be negative in this thesis is that during simulation it seemed that a positive biharmonic term would not stabilize the ice states, but that a negative one would. The use of a negative biharmonic diffusion term in general is supported by Delhez and Deleersnijder (2006) and Samuelson and Vallis (1997). Delhez and

Deleersnijder (2006) define the diffusion flux of a general scalar property w as

$$\boldsymbol{G} = -k_1 \nabla w + k_2 \nabla^3 w \tag{3.28}$$

where k_1 and k_2 are diffusion coefficients. It is given from Fick's law of diffusion that

$$\frac{\partial w}{\partial t} = -\nabla \boldsymbol{G} = k_1 \nabla^2 w - k_2 \nabla^4 w \tag{3.29}$$

and the biharmonic diffusion term is clearly negative. Also in Equation (2.5) of Samuelson and Vallis (1997), the harmonic diffusion for the ocean temperature is positive and the biharmonic diffusion for the ocean temperature is negative. Based on this, the biharmonic diffusion is chosen to be negative in this thesis.

3.3.5 Summary of the model equations

In order to give a general overview of the equations that must be implemented, a summary of the model equations is given in Appendix A.

3.3.6 The grid and its cell properties

Some variables in the equations of Sections 3.3.1 - 3.3.4 are considered to be cell properties. It is within these properties that the information about the ice state of each cell lies. The main ones are of course the model state (the mean ice thickness h and cover A), the two-dimensional ice velocity $\boldsymbol{U}$, and the ice mass per unit area m. But the ice strength P, the bulk and shear viscosities ζ and η, and the strain rate tensor $\dot{\boldsymbol{\epsilon}}$ are equally important properties which contain a lot of information about the ice state of the cell. A summary of these properties is given in Table 3.1.

As mentioned earlier, Hibler III (1979) uses a square, staggered grid equal to that in Figure 3.3b, where each grid cell has sides of length d. For each of these cells, the cell properties in Table 3.1 must be calculated. Figure 3.5 shows how Hibler III (1979) separates these properties between the two points of the staggered grid. It shows that u, v and m are calculated at integers of x and y and are constant within each "dashed" grid cell, while h, A, $\dot{\boldsymbol{\epsilon}}$, ζ, η and P are calculated at half-integers of x and y and are constant within each "full" grid cell.

In Hibler III (1979) D_1 and D_2 are also defined at grid points. This is because these diffusion coefficients are zero at grid boundaries bordering land, and

Property		Connected to
Velocity in x-direction	u	The Momentum balance
Velocity in y-direction	v	
Ice mass per unit area	m	
The ice strength	P	The constitutive law
The strain-rate tensor	$\dot{\boldsymbol{\epsilon}}$	
The bulk viscosity	ζ	
The shear viscosity	η	
The ice mean thickness	h	The ice conservation equations
The ice cover	A	

Table 3.1: Cell properties.

constant in the grid interior. For simplicity, sea ice will only be simulated for open water areas in this thesis, and the diffusion coefficients will therefore be constant for the entire grid, and are not calculated at certain grid points.

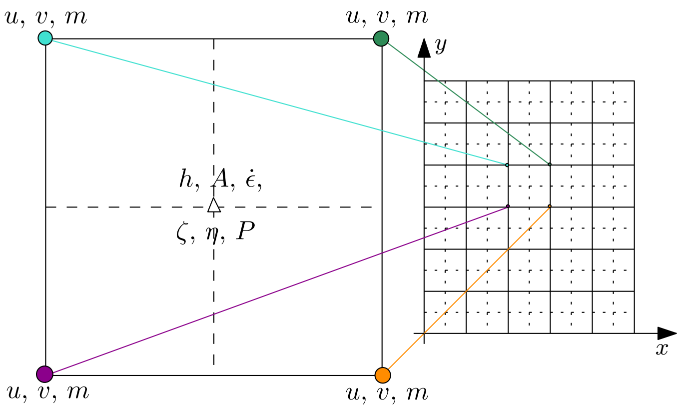


Figure 3.5: Placement of properties in cells of the staggered grid used by Hibler III (1979).

3.4 Implementing the model

When implementing and simulating the Hibler-model, it is treated as a general nonlinear dynamic system with state $\boldsymbol{x}$ and input $\boldsymbol{\Upsilon}$:

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\Upsilon}) \tag{3.30}$$

The system states and inputs are defined as

$$\boldsymbol{x} = \begin{bmatrix} u_1 & v_1 & h_1 & A_1 & \cdots & u_p & v_p & h_p & A_p \end{bmatrix}^T \tag{3.31}$$

$$\boldsymbol{\Upsilon} = \begin{bmatrix} \boldsymbol{\chi} & T_a & \boldsymbol{U}_g & \boldsymbol{U}_w & \phi & \theta \end{bmatrix}^T \tag{3.32}$$

where p is the number of cells in the grid, and $\boldsymbol{\chi}$ is the boundary conditions, which will be discussed later.

It is worth noticing that the air temperature T_a, though varying with time, is not set to vary in space. The air temperature is therefore equal for all grid cells.

3.4.1 Classes, objects and class attributes

The Hibler-model was implemented in object-oriented MATLAB through two classes; the `grid` class and the `cell` class. Each cell of the grid is represented as an object of the `cell` class which contains information about the properties of this particular cell. The entire grid is represented as an object of the `grid` class which contains properties common for the entire grid as well as a table which keeps track of each cell object in this grid. When a property is connected to a class, and hence an object of this class, it is referred to as a class *attribute*. Table 3.2 gives an overview of the most important class attributes.

	Attributes of the `cell` class	Attributes of the `grid` class
Variables	Ice velocity	Air temperature
	Ice mass per unit area	Ocean current velocity
	Mean ice thickness	Wind velocity
	Ice coverage	
	Ice strength	
	Strain rate tensor	
	Bulk and shear viscosities	
Constants	Cell number	Size of grid area
	x and y coordinates in grid	Grid resolution (cell length)
		Table containing all grid cells

Table 3.2: Important class attributes.

Even though these attributes in reality are calculated at different grid points (as described in section 3.3.6), they are, for simplicity, all attached to the same *cell object* when the model is implemented. I.e., both $\{u, v, m\}_{ij}$ and $\{h, A, \dot{\boldsymbol{\epsilon}}, \zeta, \eta, P\}_{i+\frac{1}{2},j+\frac{1}{2}}$ are attached to, and constant for, cell object (i, j).

All of the cell properties listed in Table 3.2 are attributes of the `cell` class, but there are also other attributes in addition to these, i.e.,

$$\text{cell properties} \;\subset\; \text{cell class attributes.}$$

3.4.2 Numerical details

Spatial finite differences

The model equations of the Hibler-model contains derivatives with regards to both time and space. In order to be able to integrate over time only, Hibler III (1979) uses *spatial finite differences* (SFD) to approximate the spatial derivatives. Finite differences are approximations of derivatives, and are often used to find numerical solutions of partial differential equations. There are several finite difference operators (Ames, 1992), and the following *central difference* is what will be used in this thesis:

$$\frac{dz}{dn} \approx \frac{1}{\Delta n}\left(z_{n+\frac{1}{2}} - z_{n-\frac{1}{2}}\right) \tag{3.33}$$

The SFDs needed in this thesis, and the expressions for them are given in Appendix B. Two of the SFDs differ from the expression given in Hibler III (1979), but it is assumed that there is a misprint in Hibler III (1979). See Appendix B for further details.

Limits on the viscosities

It is seen by (3.13) - (3.15) that when the strain rate $\dot{\boldsymbol{\epsilon}}$ becomes very small, the bulk and shear viscosities ζ and η become arbitrarily large. Hibler III (1979) avoids this by placing an upper limit on ζ and η which is dependent on the ice strength P.

$$\zeta_{max} = 2.5 \cdot 10^8 P \tag{3.34}$$

$$\eta_{max} = \frac{\zeta_{max}}{e^2} \tag{3.35}$$

When implemented here, this maximum limits for ζ and η are converted into a minimum limit for Δ:

$$\frac{P}{2\Delta} \leq 2.5 \cdot 10^8 P \tag{3.36}$$

$$\Downarrow$$

$$\Delta \geq 2 \cdot 10^{-9} \tag{3.37}$$

and

$$\Delta_{min} = 2 \cdot 10^{-9} \tag{3.38}$$

To avoid nonlinear instability Hibler III (1979) also places a lower boundary on ζ and η, independent of P.

$$\zeta_{min} = 4 \cdot 10^8 \tag{3.39}$$

$$\eta_{min} = \frac{\zeta_{min}}{e^2} \tag{3.40}$$

This leads to the following upper boundary for Δ:

$$\Delta_{max} = 1.25 \cdot 10^{-9} P \tag{3.41}$$

The viscous-plastic rheology used by Hibler III (1979) is known to suffer from numerical difficulties related to the model's large range of viscosities, which leads to a demand for large computational resources (Hunke and Dukowicz, 1997). With the limits given above this is exactly what happens; the time step of the simulation becomes very small and the simulation time exceeds the limit for practical use. Through trial and error the following limits were found to give an acceptable simulation time.

$$\Delta_{min} = 2.5 \cdot 10^{-7} \tag{3.42}$$

$$\Delta_{max} = 1.25 \cdot 10^{-12} P \tag{3.43}$$

Clearly there are values of P where $\Delta_{max} < \Delta_{min}$, in these cases Δ_{min} is ignored and $\Delta = \Delta_{max}$ (Hunke and Dukowicz, 1997).

3.4.3 Boundary conditions

Earlier the boundary conditions for ice environment simulations have been either solid land or open water (free of sea ice). In such cases the boundary

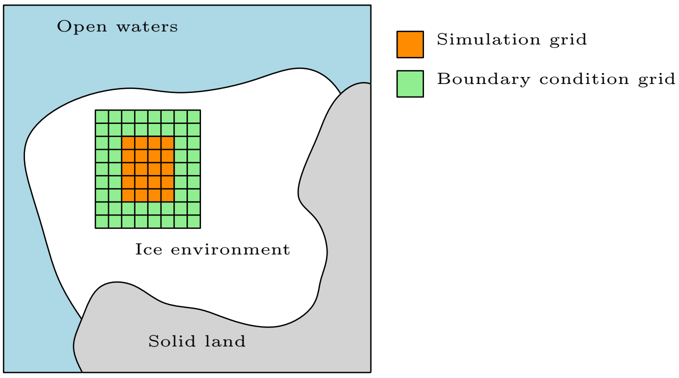


Figure 3.6: Simulation area with surrounding boundary-condition cells.

conditions can, in general, be implemented as (Leppäranta, 2005)

$$\begin{aligned} \boldsymbol{\sigma} \cdot \boldsymbol{n} &\leq 0 \quad \text{solid land boundary} \\ \boldsymbol{\sigma} \cdot \boldsymbol{n} &= 0 \quad \text{open water boundary} \end{aligned} \tag{3.44}$$

where $\boldsymbol{n}$ is a vector normal to the ice. These boundary conditions are often replaced by a simplified form (Leppäranta, 2005)

$$\begin{aligned} \boldsymbol{U} &= 0 \quad \text{solid land boundary} \\ h &= 0 \quad \text{open water boundary} \end{aligned} \tag{3.45}$$

In this thesis a smaller area of the entire ice environment will be simulated, and consequently the grid will be surrounded by neither open waters nor solid land, but waters containing sea ice. None of these boundary conditions can therefore be applied here, and a new set of boundary conditions containing ice properties must be developed.

When calculating the SFDs of cell (i, j) one must have knowledge of the properties of cells $(i \pm 1, j \pm 1)$ as well as $(i \pm 2, j \pm 2)$; i.e. two "rounds" of boundary-condition cells are needed. Figure 3.6 portrays the simulation grid and the boundary-condition grid in a larger ice environment. The "dashed" grid is left out for simplicity.

The ice properties which must be set for each of the boundary-condition cells are u, v, h, A, ζ and η. For simplicity, u and v are chosen to be constant and equal for all the boundary-condition cells, while h, A, ζ and η are set to vary with the air temperature, however also equal for all boundary-condition cells:

$$\boldsymbol{\chi}(T_a) = \{u_{bc}, v_{bc}, h_{bc}(T_a), A_{bc}(T_a), \zeta_{bc}(T_a), \eta_{bc}(T_a)\}$$

$h_{bc}\left(T_a\right)$ and $A_{bc}\left(T_a\right)$ are design choices, and must be chosen in accordance with the initial values for h and A of the simulation grid.

Since $\boldsymbol{U}_{bc} = \begin{bmatrix} u_{bc} & v_{bc} \end{bmatrix}^T$ is constant for the boundary-condition grid, it can be found through the stationary solution of (3.5a) (remember that $\nabla H = 0$):

$$0 = -mf\boldsymbol{k} \times \boldsymbol{U}_{bc} + \boldsymbol{\tau}_a + \boldsymbol{\tau}_w + \boldsymbol{F} \tag{3.46}$$

Since $\boldsymbol{U}_{bc}$ are equal for all boundary-condition cells, $\boldsymbol{F} \approx 0$. In addition, the forces due to water stress $\boldsymbol{\tau}_w$ may be ignored as the forces due to air stress $\boldsymbol{\tau}_a$ are the prevailing ice driving forces (Leppäranta, 2005). This results in the equation

$$0 = -mf\boldsymbol{k} \times \boldsymbol{U}_{bc} + \boldsymbol{\tau}_a \tag{3.47}$$

which easily solves for $\boldsymbol{U}_{bc}$, given U_g and ϕ:

$$\boldsymbol{U}_{bc} = \begin{bmatrix} u_{bc} \\ v_{bc} \end{bmatrix} = \frac{\rho_a C_a}{mf} \begin{bmatrix} U_{gy} \\ -U_{gx} \end{bmatrix} \left(\begin{bmatrix} U_{gy} \\ U_{gx} \end{bmatrix} \cos(\phi) + \begin{bmatrix} U_{gx} \\ -U_{gy} \end{bmatrix} \sin(\phi) \right) \tag{3.48}$$

For the same reason that $\boldsymbol{F} \approx 0$, $\Delta = \Delta_{min}$. This leads to $\zeta_{bc} = \zeta_{max}\left(P_{bc}\right)$ and $\eta_{bc} = \eta_{max}\left(P_{bc}\right)$. Thus, given $h_{bc}\left(T_a\right)$ and $A_{bc}\left(T_a\right)$ (and hence $P_{bc}\left(T_a\right)$), u_{bc}, v_{bc}, $\zeta_{bc}\left(P_{bc}\right)$ and $\eta_{bc}\left(P_{bc}\right)$ may be calculated.

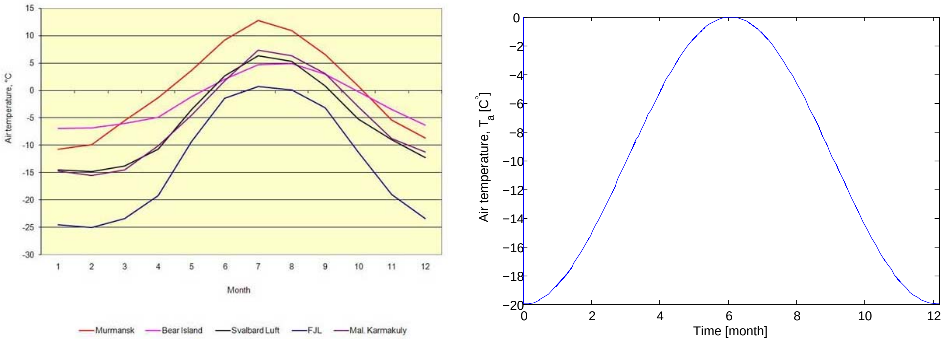
3.5 Simulation and validation of model

In order to validate the implemented model, it was simulated over one year with varying air temperature. Even though simulation times of this length are not required for the application discussed in this thesis, it allows one to check whether or not the model responds reasonably to annual temperature changes, and hence validate it.

3.5.1 Air temperature, wind, current and other constant parameters

The air temperature used for simulation was decided based on arctic temperature data from *Meteorological conditions* (2011), shown in Figure 3.7a. If the air temperature at Franz-Josef Land is used as a sketch, the seasonal air temperature can be modeled as a shifted cosine wave with amplitude of 10 °C, period of one year and mean equal to $-10\ ^{\circ}C$:

$$T_a(t) = -10\cos\left(\frac{\pi t}{24 \cdot 182}\right) - 10 \tag{3.49}$$



(a) Measured seasonal air temperatures (*Meteorological conditions*, 2011).

(b) Generated seasonal air temperatures.

Figure 3.7: Measured and generated seasonal air temperatures in the Arctic.

where t is given in hours, see Figure 3.7 for comparison. For simplicity, the same air temperature is used for the entire grid.

When it comes to the geostrophic wind and current, inspiration was taken from Wang and Ikeda (2004), where the following geostrophic wind and current were used

$$(\boldsymbol{U}_g)_{i,j} = \begin{bmatrix} 3.2 + 0.1(i+j)/d \\ -3.6 + 0.2(i+j)/d \end{bmatrix} \tag{3.50}$$

$$(\boldsymbol{U}_w)_{i,j} = \begin{bmatrix} 0.010 - 0.001(i-j)/d \\ 0.052 + 0.001(i+j)/d \end{bmatrix} \tag{3.51}$$

where d is the cell length and (i, j) is the cells spatial location in the grid. In reality, both the geostrophic wind and current are time dependent. The expressions given above are however not, and they must be seen as a simplification of real-life conditions. When calculating $\boldsymbol{U}_{bc}$, only the part of $\boldsymbol{U}_g$ which is constant for the entire grid is used.

Table 3.3 lists the values which were used for the constant model parameters during simulation. Most of the values are equal to those used by Hibler III (1979), except Δ_{min} and Δ_{max} (as described above above), and $P^{\star}$ which, according to Leppäranta (1993), has been found to be higher than first assumed by Hibler III (1979). Hibler III (1979) used $P^{\star} = 5 \cdot 10^3$ ${}^{\mathrm{N}}\!/\!{}_{\mathrm{m}}$, while Leppäranta (1993) states that normal levels for $P^{\star}$ is around $20 - 30 \cdot 10^3$ ${}^{\mathrm{N}}\!/\!{}_{\mathrm{m}}$. The values of the properties connected to the growth-rate function are from Leppäranta (1993).

C_a	$= 0.0012$	ρ_w	$= 1027\ {}^{\mathrm{kg}}/_{\mathrm{m}^3}$
C_w	$= 0.0055$	ρ_a	$= 1.3\ {}^{\mathrm{kg}}/_{\mathrm{m}^3}$
C	$= 20$	ρ_{ice}	$= 920\ {}^{\mathrm{kg}}/_{\mathrm{m}^3}$
$P^{\star}$	$= 25 \cdot 10^3\ {}^{\mathrm{N}}/_{\mathrm{m}}$	Δ_{min}	$= 2.5 \cdot 10^{-7}\ \mathrm{s}^{-1}$
f	$= 1.46 \cdot 10^{-4}\ \mathrm{s}^{-1}$	Δ_{max}	$= 1.25 \cdot 10^{-12} P\ (\mathrm{Ns})^{-1}$
h_0	$= 0.5\ \mathrm{m}$	D_1	$= 0.004d$
r	$= 2$	D_2	$= 0.004d^3$
d	$= 125\ \mathrm{km}$	ϕ	$= \theta = 25°$
L	$= 335 \cdot 10^3\ {}^{\mathrm{J}}/_{\mathrm{kg}}$	κ_i	$= 2.1\ {}^{\mathrm{W}}/_{\mathrm{Km}}$
k_a	$= 21$	T_f	$= -1.9\ °\mathrm{C}$
Q_w	$= 2\ {}^{\mathrm{W}}/_{\mathrm{m}^2}$		

Table 3.3: Parameters used during simulation.

3.5.2 Simulation flow chart

When simulating the Hibler-model, the execution sequence described by the flow chart in Figure 3.8 was applied. It shows the high-level order of the different updates made for each cell.

3.5.3 Initial values

Since the area simulated here is fictional, there are no data to lean on when choosing realistic initial values for h and A. But since other numerical parameters are, to a large extent, set equal to those used by Hibler III (1979), and the fictive area is assumed to be in the Arctic, the initial values for h and A are inspired by those in Hibler III (1979). There, $h_{init} = 3.2967\ m$ and $A_{init} = 1.0$ for every cell on Julian Day 1[2] (Hibler III, 1979), and so h_{init} and A_{init} were chosen to be 3 m and 1.0, respectively. The initial values for u and v are simply set equal to zero.

In order to get an even more realistic initial state, the system is simulated for 365 "Julian Day 1-days", so that the state stabilizes at equilibrium values. Ignoring the year, Julian Day 1 is January 1st, at which the air temperature is about −20 °C. I.e., the system is first simulated for one year with constant air temperature of −20 °C, before the actual simulation with varying air temperature begins. The simulation results from cell number 11-15, with the initial values given above, $T_a = -20$ °C, U_g and U_w as in (3.50)-(3.51), and

---

[2] According to *Julian Day* (2011), Julian Day 1 is the 1st of January 4713 BC.

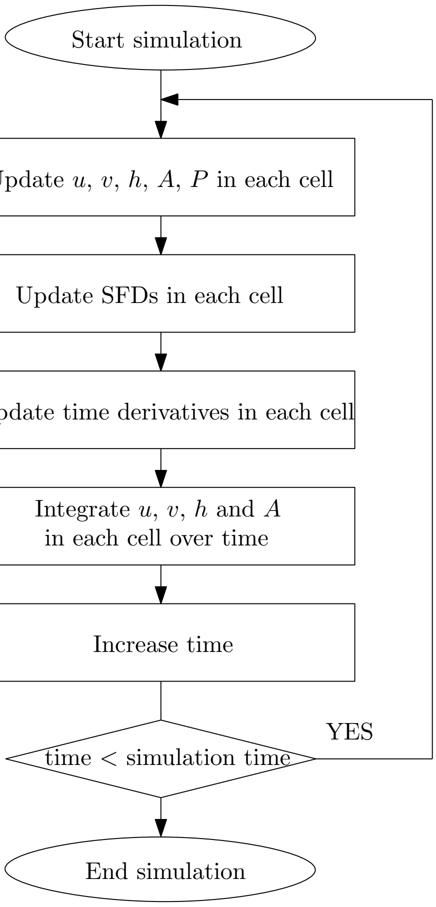


Figure 3.8: Flow chart for each time step.

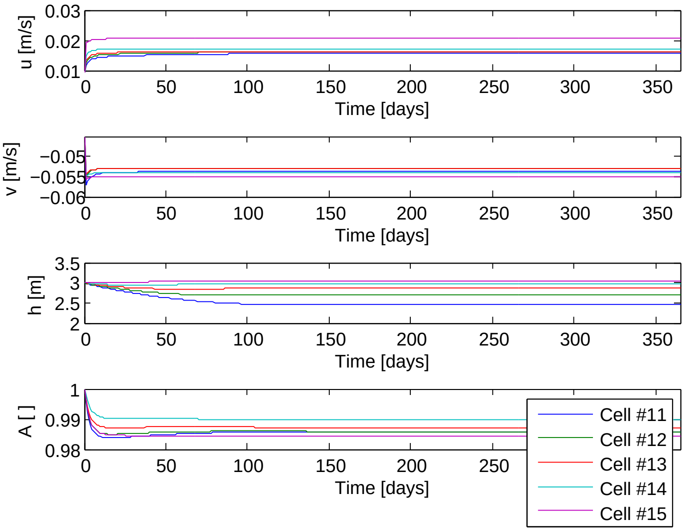


Figure 3.9: Simulation results for constant air temperature at −20 °C.

the boundary conditions given below, can be seen in Figure 3.9. It clearly shows that u stabilizes around $0.01 - 0.02$ $^{\mathrm{m}}/_{\mathrm{s}}$, v at around -0.055 $^{\mathrm{m}}/_{\mathrm{s}}$, h between 2.5 and 3 m and A at about $0.98 - 0.99$. The remaining cells are left out as their results are similar to those in cell 11-15. The results from this simulation are the actual initial values for the system given these conditions.

3.5.4 Choosing the boundary condition parameters

The mean thickness and the compactness of the boundary-condition cells $h_{bc}\left(T_a\right)$ and $A_{bc}\left(T_a\right)$ influence the simulation results, especially the mean ice thickness of the cells. This is because, in addition to the air temperature and the geostrophic wind and current, it is the boundary conditions which drive the model.

As long as the ice velocity is non-zero, the ice properties of each cell will be highly influenced by the properties of its surrounding cells. The boundary-condition cells will therefore affect the cells around the edges of the grid, which again will affect the cells one step further in, and so on. Since the geostrophic wind and current in each cell are dominated by terms which are constant for the entire grid, the cells positioned at grid edges directly exposed to this constant wind and current are especially influenced, and it is mainly from here that the boundary-condition effect propagates through the grid.

When choosing $h_{bc}(T_a)$ and $A_{bc}(T_a)$, emphasis was placed on making them

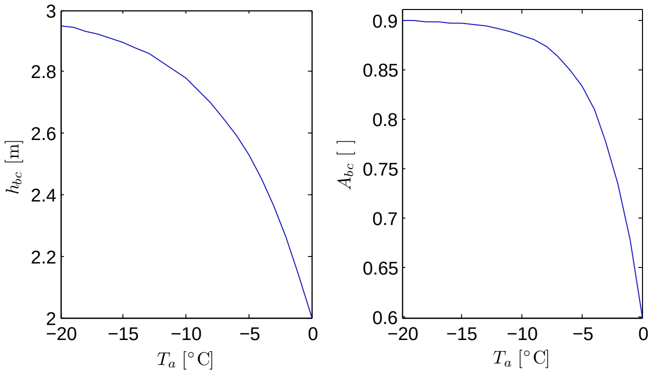


Figure 3.10: Temperature-dependent mean ice thickness and ice compactness of boundary-conditions cells.

somewhat realistic functions of the air temperature T_a. It is assumed that the thickness and compactness of the ice in the boundary-condition cells "follows" the air temperature by increasing as the air temperature decreases, and from (3.23) it can be seen that, given a certain air temperature, the growth rate of the ice will be smaller for thicker ice. Hence, it is natural that both $h_{bc}(T_a)$ and $A_{bc}(T_a)$ are functions which are steep for small negative air temperatures, and which flatten out as the air temperature becomes increasingly negative. Since $T_a \leq 0$, it is sufficient that $h_{bc}(T_a)$ and $A_{bc}(T_a)$ are defined for negative air temperatures.

Two functions which meet these criteria, and keep the mean thickness of the grid cells around $2 - 3$ m are

$$h_{bc}(T_a) = 2 + \left(1 - e^{0.15 T_a}\right) \tag{3.52}$$

$$A_{bc}(T_a) = 0.6 + 0.3\left(1 - e^{0.3 T_a}\right) \tag{3.53}$$

where $T_a \leq 0$. Plots of these functions can be seen in Figure 3.10.

3.5.5 Simulation and validation

The system is simulated with the air temperature, constant parameters, initial values and boundary conditions given above. First it is simulated with no geostrophic wind or current in order to show that the states behaves reasonably with approximately stagnant ice, before $\boldsymbol{U}_g$ and $\boldsymbol{U}_w$ are set as in (3.50)-(3.51).

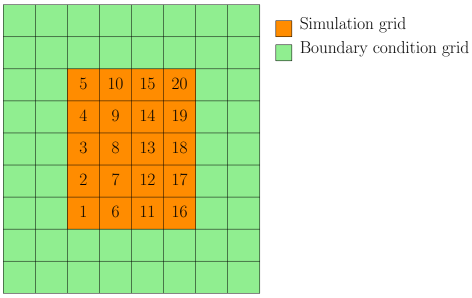


Figure 3.11: The simulation grid with numbered cells.

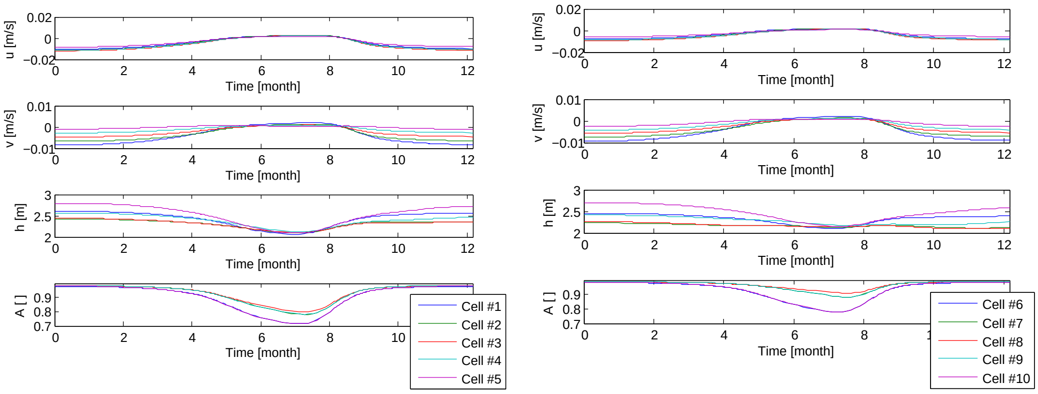
The fictive area which is simulated, is part of a larger ice-environment, and consists of a 4 × 5-grid containing cells of length 125 km. The entire simulation area is thus 500 km i x-direction and 625 km in y-direction. Figure 3.11 shows the simulation grid displaying the number of each cell, as well as the boundary-condition grid. Because it a fictive area that is being simulated, it is difficult to find any results from the literature to validate against; the validations rather consists of evaluating the model's response to a realistically varying seasonal air temperature up against intuitive knowledge.

No geostrophic wind or current

Figure 3.12 displays how u, v, h and A respond to the varying air temperature with zero geostrophic wind and current. It clearly shows that the ice melts in the summer months and freezes up again as the air temperature drops.

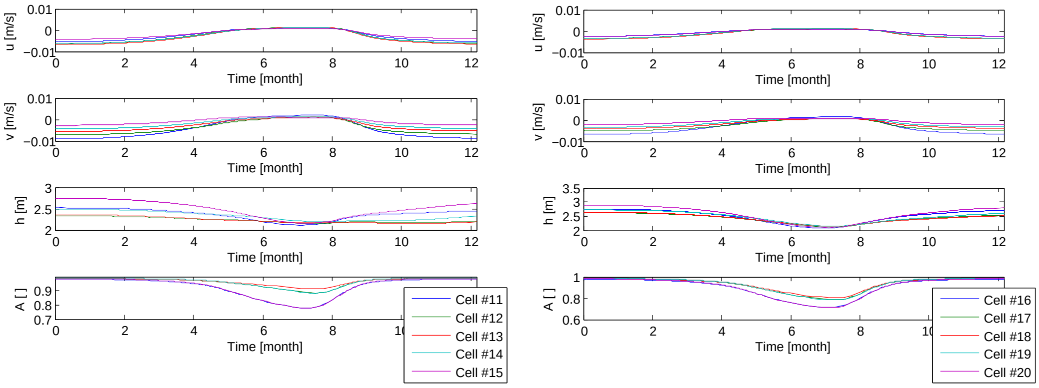
It is also seen from Figure 3.12b and 3.12c that the cells in the inner part of the grid (cell number $7-9$ and $12-14$) are less affected by the temperature increase than those around the edges, which is consistent with the propagation of the boundary-condition effect mentioned in the previous section. Since both the geostrophic wind and current are set equal to zero, one might think that the ice velocity is zero and that the boundary-conditions has no effect. In that case, it would be the air temperature alone that influences h and A. But the fact is that even with zero geostrophic wind and current, the velocity of the ice will not remain zero, and the boundary conditions will influence the grid cells. However, as will be seen next, the effect becomes greater when the ice velocity increases due to non-zero geostrophic wind and current.

All in all, the system responds as expected: Both the ice compactness and the mean ice thickness decrease as the air temperature rises, and increase



(a) Cell number $1 - 5$.

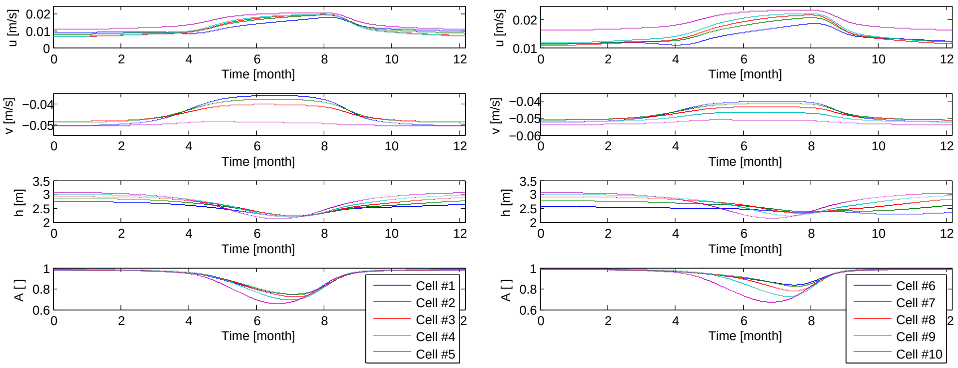
(b) Cell number $6 - 10$.



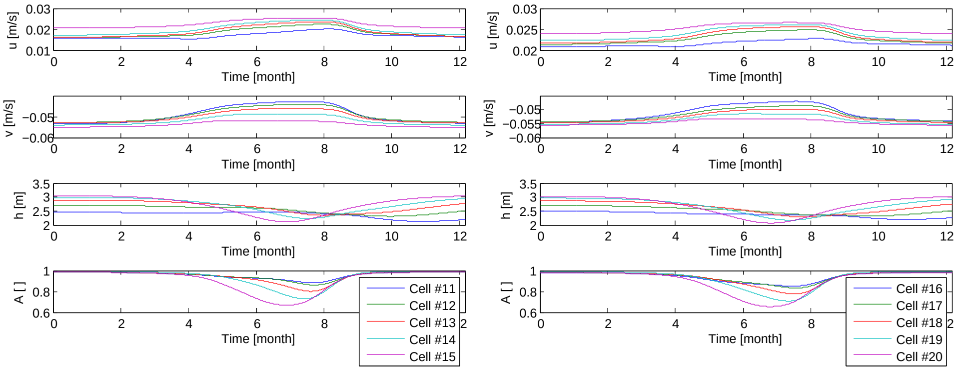
(c) Cell number $10 - 15$

(d) Cell number $16 - 20$.

Figure 3.12: u, v, h and A from simulating with no geostrophic wind or current.



(a) Cell number $1-5$. (b) Cell number $6-10$.



(c) Cell number $10-15$ (d) Cell number $16-20$.

Figure 3.13: u, v, h and A from simulating with geostrophic wind and current.

again as the air temperature drops; due to the resulting change in ice mass, the ice velocity is also affected by the varying air temperature.

Non-zero geostrophic wind and current

The system is now simulated with non-zero geostrophic wind and current, the results are displayed in Figure 3.13. The results are quite similar to those in Figure 3.12, except that the absolute value of the ice velocity is larger, and that the effect from the boundary conditions are more prominent, both as expected. It is seen from Figure 3.13b, 3.13c and 3.13d that cells which are not directly exposed to the dominating geostrophic wind and current, are a bit delayed in their *mean thickness*-response and *compactness*-response to the temperature rise. And the further away the cells are from the "exposure"-edges, the more delayed these responses are, and the less is the effect of the temperature rise.

Cell number $1-5$ are all directly exposed to the dominating geostrophic wind and current, and the boundary conditions affect them all equally. Figure 3.13a clearly shows this, as both h and A responds in a somewhat similar manner for all cells. From Figure 3.13b, 3.13c and 3.13d it is seen that cell number 10, 15 and 20, which are the remaining cells at the "exposure"-edges, responds to the temperature rise both quicker and heavier than the rest. The *mean thickness*-response and *compactness*-response of cell number $8-9$, $13-14$ and $18-19$, and the *compactness*-response of cell number $6-7$, $11-12$ and $16-17$, are all delayed and down-scaled versions of 10, 15 and 20, respectively, while the mean thickness of cell number $6-7$, $11-12$ and $16-17$ is less affected. This is a natural effect of the chosen boundary conditions, and shows how it spreads through the grid as mentioned in Section 3.5.4.

3.6 Chapter summary

In this chapter, the dynamic thermodynamic sea-ice model of Hibler III (1979) was presented, its implementation was described and it was validated through simulation.

When modeling sea ice there are roughly four topics to consider:

- Deciding which scale(s) the model should include, and whether to use a continuum or discrete model.
- Deciding what sort of grid to use, and whether to use an Eulerian or Lagrangian frame.
- Defining the system states, and
- whether to use finite element or finite difference approach to solve the PDE.

The Hibler-model is a continuum model with a square staggered grid and Eulerian frame, which uses finite difference approach to approximate the spatial derivatives. The boundary conditions were set to be temperature-varying and spatially independent. Hence, they are equal for the entire boundary-condition grid. For simplicity, some assumptions were made during the implementation of the Hibler-model. They are as follows:

- It was assumed that the sea-surface dynamic height H is constant for the entire grid,
- that the time varying air temperature $T_a(t)$ is common for the entire grid,
- and that the geostrophic wind and current $\boldsymbol{U}_g$ and $\boldsymbol{U}_{\boldsymbol{w}}$, are time invariant.

Even though these simplifications of real-life conditions lead to some inaccuracies, they do not affect the model's realistic behavior, as seen by the simulations in Section 3.5.

Chapter 4

The State Estimator

In this chapter the ensemble Kalman filter will be presented as a state estimator for the sea-ice model presented in Chapter 3.

In Section 4.1, both the original Kalman filter for linear systems and the extended Kalman filter for nonlinear system are described, before the ensemble Kalman filter is presented as an alternative to the extended Kalman filter.

Possible measurements of the Hibler-model are discussed in Section 4.2, and suggestions for measurement-matrix candidates are given. Process and measurement noise is added in Section 4.3, and the performance of different ensemble Kalman filters based on different measurement matrices, are shown in Section 4.4.

4.1 The Kalman Filter

In 1960, Rudolf E. Kalman published a paper describing an new method for estimating the states of a linear process, known as *Kalman filtering* (Simon, 2006). The Kalman filter is a recursive algorithm which exploits knowledge of both the process as well as the noise it is subjected to, in order to estimate the state vector so that the mean of the squared estimate error is minimized. The Kalman filter has become popular within many areas of application, from navigation to weather forecasting, especially since the the extended Kalman filter was introduced in the late 1960s (Simon, 2006), which is a Kalman Filter for non-linear processes.

The Kalman filter can be given both as discrete or continuous in time. The discrete-time Kalman filter is the most common one, and it is also applied in this thesis. It is therefore only the discrete-time Kalman filter which is discussed further, and it will from now on be referred to merely as the Kalman filter.

The Kalman filter is a sequential estimator. Assuming that there is a measurement available at each time step, each cycle of the Kalman filter can be divided into four main parts: the integration, the computation of the Kalman gain, the measurement update and the incrementation of time, see Figure 4.1. After the time is incremented, the estimate is integrated forward in time, this estimate is then refined during the measurement update, using the current measurement and the Kalman gain.

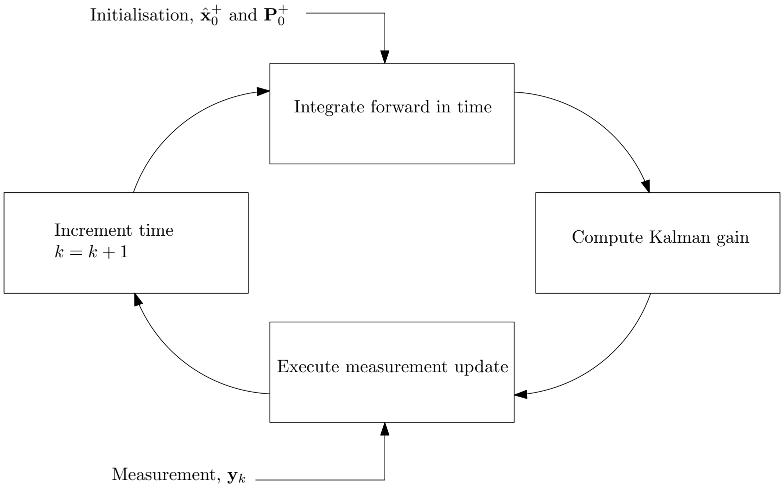


Figure 4.1: The Kalman filter cycle.

There are many different versions of the Kalman filter, such as linear (the original Kalman filter for linear systems), extended (for nonlinear systems) and ensemble (for large, nonlinear systems). They all share the structure of Figure 4.1, but the equations within each part of the cycle vary for each of them.

4.1.1 The linear Kalman filter

The *linear Kalman filter* (KF) was the one first presented by Rudolf E. Kalman, and it has later proven to be the optimal linear state estimator (Simon, 2006). The KF is not applied in this thesis, but it is presented here as a foundation for understanding the later presented extended and ensemble Kalman Filter.

Given a linear discrete-time system

$$\boldsymbol{x}_k = \boldsymbol{A}\boldsymbol{x}_{k-1} + \boldsymbol{B}\boldsymbol{u}_{k-1} + \boldsymbol{w}_{k-1} \tag{4.1a}$$

$$\boldsymbol{y}_k = \boldsymbol{C}\boldsymbol{x}_k + \boldsymbol{v}_k \tag{4.1b}$$

where $\boldsymbol{w}_k$ and $\boldsymbol{v}_k$ are the process and measurement noise, respectively. They are white, zero-mean and uncorrelated with known covariance matrices $\boldsymbol{Q}$ and $\boldsymbol{R}$.

$$\boldsymbol{w}_k \sim (0, \boldsymbol{Q}) \tag{4.2}$$

$$\boldsymbol{v}_k \sim (0, \boldsymbol{R}) \tag{4.3}$$

During the execution, the KF keeps track of, and make use of, the *error covariance matrix* $\boldsymbol{P}_k$, which denotes the covariance of the error between the estimated and actual state vector (Simon, 2006).

$$\boldsymbol{P}_k = E\left[(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k)(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k)^T\right] \tag{4.4}$$

The error covariance matrix is an important part of the KF, as it is where the information about the correctness of the estimate is kept. Along with the covariance matrix of the process noise, it plays an important role in determining the Kalman gain, which again controls how much the measurements influence the estimate.

With measurements being made at every time step $\boldsymbol{y}_k$, the linear Kalman filter follows the pattern of Figure 4.1, with the following equations implemented at each step:

Initialization

The initial estimate $\hat{\boldsymbol{x}}_0^+$ is set to be the expected value of the initial state, and the initial error covariance matrix $\boldsymbol{P}_0^+$ is set so that it represents the uncertainty in the initial estimate of $\boldsymbol{x}_0$ (Simon, 2006).

$$\hat{\boldsymbol{x}}_0^+ = E(\boldsymbol{x}_0) \tag{4.5}$$

$$\boldsymbol{P}_0^+ = E\left[(\boldsymbol{x}_0 - \hat{\boldsymbol{x}}_0^+)(\boldsymbol{x}_0 - \hat{\boldsymbol{x}}_0^+)^T\right] \tag{4.6}$$

Integration

The estimate and the error covariance matrix are integrated forward in time.

$$\hat{\boldsymbol{x}}_k^- = \boldsymbol{A}\hat{\boldsymbol{x}}_{k-1}^+ + \boldsymbol{B}\boldsymbol{u}_{k-1} \tag{4.7}$$

$$\boldsymbol{P}_k^- = \boldsymbol{A}\boldsymbol{P}_{k-1}^+\boldsymbol{A}^T + \boldsymbol{Q} \tag{4.8}$$

Computing the Kalman gain

$$\boldsymbol{K}_k = \boldsymbol{P}_k^- \boldsymbol{C}^T (\boldsymbol{C}\boldsymbol{P}_k^- \boldsymbol{C}^T + \boldsymbol{R})^{-1} \tag{4.9}$$

Measurement update

The estimate and error covariance matrix are updated using the measurement $\boldsymbol{y}_k$ and the Kalman gain $\boldsymbol{K}_k$.

$$\hat{\boldsymbol{x}}_k^+ = \hat{\boldsymbol{x}}_k^- + \boldsymbol{K}_k \left(\boldsymbol{y}_k - \boldsymbol{C}\hat{\boldsymbol{x}}_k^- \right) \tag{4.10}$$

$$\boldsymbol{P}_k^+ = (\boldsymbol{I} - \boldsymbol{K}_k \boldsymbol{C}) \boldsymbol{P}_k^- (\boldsymbol{I} - \boldsymbol{K}_k \boldsymbol{C})^T + \boldsymbol{K}_k \boldsymbol{R} \boldsymbol{K}_k^T \tag{4.11}$$

The estimates $\hat{\boldsymbol{x}}_k^-$ and $\hat{\boldsymbol{x}}_k^+$ are called the *a priori* and *a posteriori* estimate, respectively. The a priori estimate is calculated directly from the model equations. It represents the time update of the previous a posteriori estimate, and is the best estimate at time k *before* the measurements $\boldsymbol{y}_k$ has been processed. The error covariance matrix of the a priori estimate $\boldsymbol{P}_k^-$ is also a time update based on the previous error covariance matrix and the covariance matrix of the process noise.

The a posteriori estimate is a refinement of the a priori estimate, based on the current measurements $\boldsymbol{y}_k$, and so is the a posteriori error covariance matrix $\boldsymbol{P}_k^+$. The a posteriori estimate is the best estimate at time k *after* $\boldsymbol{y}_k$ has been processed, and it is also the basis for the next cycle of the Kalman filter. The relationship between these estimates and error covariance matrices are depicted in Figure 4.2.

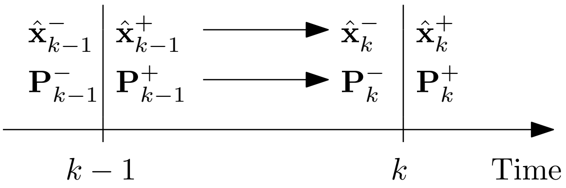


Figure 4.2: A priori and a posteriori estimates and error covariances (Simon, 2006).

4.1.2 The extended Kalman filter

For nonlinear systems, the linear Kalman filter is not applicable, and nonlinear extensions of the Kalman filter must be employed. The most common nonlinear extension of the Kalman filter is the *extended Kalman filter* (EKF).

The EKF was first proposed by Stanley Schmidt, and the idea is to linearize the nonlinear system around the Kalman filter estimate $\hat{\boldsymbol{x}}$, and then use linear Kalman-filter theory to estimate the state value (Simon, 2006).

Given the discrete nonlinear system

$$\boldsymbol{x}_k = \boldsymbol{f}_k(\boldsymbol{x}_{k-1}, \boldsymbol{u}_{k-1}) + \boldsymbol{w}_{k-1} \tag{4.12a}$$

$$\boldsymbol{y}_k = \boldsymbol{C}\boldsymbol{x}_k + \boldsymbol{v}_k \tag{4.12b}$$

$$\boldsymbol{w}_k \sim (0, \boldsymbol{Q}) \tag{4.12c}$$

$$\boldsymbol{v}_k \sim (0, \boldsymbol{R}) \tag{4.12d}$$

where $\boldsymbol{x}$ is a state vector of size n. The equations for the extended Kalman filter, which can be placed directly into Figure 4.1, are as follows.

Initialization

$$\hat{\boldsymbol{x}}_0^+ = E(\boldsymbol{x}_0) \tag{4.13}$$

$$\boldsymbol{P}_0^+ = E\left[(\boldsymbol{x}_0 - \hat{\boldsymbol{x}}_0^+)(\boldsymbol{x}_0 - \hat{\boldsymbol{x}}_0^+)^T\right]. \tag{4.14}$$

Integration

$$\hat{\boldsymbol{x}}_k^- = \boldsymbol{f}_k(\hat{\boldsymbol{x}}_{k-1}^+, \boldsymbol{u}_{k-1}) \tag{4.15}$$

$$\boldsymbol{P}_k^- = \boldsymbol{F}_{k-1}\boldsymbol{P}_{k-1}^+\boldsymbol{F}_{k-1}^T + \boldsymbol{Q} \tag{4.16}$$

where

$$\boldsymbol{F}_{k-1} = \left.\frac{\partial \boldsymbol{f}_k}{\partial \boldsymbol{x}}\right|_{\hat{\boldsymbol{x}}_{k-1}^+} \tag{4.17}$$

Kalman gain

$$\boldsymbol{K}_k = \boldsymbol{P}_k^-\boldsymbol{C}_k^T(\boldsymbol{C}_k\boldsymbol{P}_k^-\boldsymbol{C}_k^T + \boldsymbol{R})^{-1} \tag{4.18}$$

Measurement update

$$\hat{\boldsymbol{x}}_k^+ = \hat{\boldsymbol{x}}_k^- + \boldsymbol{K}_k[\boldsymbol{y}_k - \boldsymbol{C}\hat{\boldsymbol{x}}_k^-] \tag{4.19}$$

$$\boldsymbol{P}_k^+ = (\boldsymbol{I} - \boldsymbol{K}_k\boldsymbol{C}_k)\boldsymbol{P}_k^-(\boldsymbol{I} - \boldsymbol{K}_k\boldsymbol{C}_k)^T + \boldsymbol{K}_k\boldsymbol{R}\boldsymbol{K}_k^T \tag{4.20}$$

The EKF and complex nonlinear system

There are many nonlinear systems which are so complex that the explicit term for equation (4.17) is not easily found. For systems such as these, (4.17) must be approximated through *forward differencing.*

The method of forward-difference approximation is described in Chapter 8 of Nocedal and Wright (2006). Given a vector function $\boldsymbol{g}(\boldsymbol{x}) : \mathbb{R}^n \to \mathbb{R}^m$, its Jacobian can be approximated as

$$\boldsymbol{J}(\boldsymbol{x}) = \left[\frac{\partial g_j}{\partial x_i}\right] = \left[\frac{\partial \boldsymbol{g}(\boldsymbol{x})}{\partial x_1} \cdots \frac{\partial \boldsymbol{g}(\boldsymbol{x})}{\partial x_n}\right] \tag{4.21}$$

where

$$\frac{\partial \boldsymbol{g}(\boldsymbol{x})}{\partial \boldsymbol{x}_i} \approx \frac{\boldsymbol{g}(\boldsymbol{x} + \epsilon \boldsymbol{e}_i) - \boldsymbol{g}(\boldsymbol{x})}{\epsilon} \tag{4.22}$$

$i = 1, \cdots, n$, $j = 1, \cdots, m$, ϵ is the perturbation parameter and $\boldsymbol{e}_i$ is the ith unit vector. Thus, $n + 1$ perturbations are needed to approximate the Jacobian of $\boldsymbol{g}(\boldsymbol{x})$, and for complex functions with large n this can be very time consuming.

4.1.3 The ensemble Kalman filter

The EKF works very well for most nonlinear systems, and Simon (2006) states that it undoubtedly is the most widely used nonlinear state-estimation technique, in large part due to the extensive increase of computational power that the world has witnessed the past few decades.

There are, however, two major drawbacks of the EKF when estimating the states of highly dimensional and complex nonlinear systems (Evensen, 2009). The first one is related to computational issues. It is seen by the previous section that a system with n states requires $n + 1$ perturbations at each time step to calculate $\boldsymbol{F}_{k-1}$, and hence execute the time update of the error covariance matrix. For large, complex systems the time update of the error covariance matrix can therefore be very time consuming (Evensen, 2009), often too time consuming for the state estimator to function properly in connection with a controller.

The second drawback of the EKF is that the linearization of nonlinear systems can lead to poor error covariance evolution, and for some systems unstable error-covariance growth (Evensen, 2009). This may be overcome by using higher order closure schemes, but this requires even greater computational effort and is therefore not applicable for large systems.

The *ensemble Kalman filter* (EnKF) was presented by Evensen (1994) as a less time consuming alternative to the EKF, and it was designed to resolve the issues described above (Evensen, 2009). The key differences between the EKF and the EnKF are depicted in Figure 4.3.

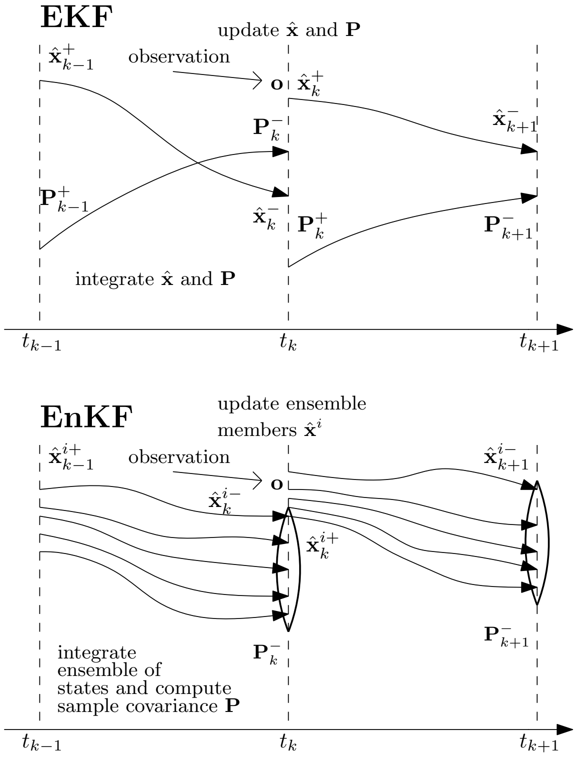


Figure 4.3: Time developement of extended Kalman filter (EKF) and ensemble Kalman filter (EnKF) (Reichle et al., 2002).

The idea behind the EnKF is to replace the deterministic representation of the error covariance matrix in the EKF with a statistical one. As the KF and the EKF, also the EnKF works sequentially from one measurement time to the next, executing first a time update and secondly a measurement update (see Figure 4.1). The EnKF is, however, not dependent on the Jacobian of $\boldsymbol{f}_k(\cdot)$ to calculate $\boldsymbol{P}^-_k$, and with the EnKF there is no need for a measurement update of $\boldsymbol{P}$.

The reason for this is that the EnKF consists of an ensemble of state-vector estimates which it propagates in parallel, each a representation of a particular realization of the possible model trajectory (Reichle et al., 2002). The EnKF does not need to explicitly integrate the error covariance matrix $\boldsymbol{P}$, since it computes it diagnostically from the distribution of the state estimates across the ensemble (Reichle et al., 2002). And since $\boldsymbol{P}_k^-$ is not calculated based on $\boldsymbol{P}_{k-1}^+$, there is no need to calculate $\boldsymbol{P}_k^+$.

The actual state estimate of the EnKF $\hat{\boldsymbol{x}}$ is given by the mean of the ensemble members, and the uncertainty of this estimate is thus represented through the spreading of the ensemble members; the larger the spread the more uncertain is the estimate and vice versa.

Given again the nonlinear system (4.12), and defining the ensemble size as N, the equations of the ensemble Kalman filter can be inserted into Figure 4.1, and are as follows.

Initialization

The EnKF is initialized by generating an ensemble of initial conditions $\hat{\boldsymbol{X}}_0^+ = \begin{bmatrix}\hat{\boldsymbol{x}}_0^{1+} & \cdots & \hat{\boldsymbol{x}}_0^{N+}\end{bmatrix}$. The spread of these initial estimates represents the initial error covariance.

Integration

$$\hat{\boldsymbol{x}}_k^{i-} = \boldsymbol{f}(\hat{\boldsymbol{x}}_{k-1}^{i+}, \boldsymbol{u}_{k-1}) + \bar{\boldsymbol{w}}_k^i \ , \ i = 1, \cdots, N \tag{4.23}$$

$$\boldsymbol{P}_k^- = \frac{1}{N-1}\boldsymbol{D}_k\boldsymbol{D}_k^T \tag{4.24}$$

where

$$\boldsymbol{D}_k = \begin{bmatrix}\hat{\boldsymbol{x}}_k^{1-} - \hat{\boldsymbol{x}}_k^- & \cdots & \hat{\boldsymbol{x}}_k^{N-} - \hat{\boldsymbol{x}}_k^-\end{bmatrix} \tag{4.25}$$

$$\hat{\boldsymbol{x}}_k^- = \frac{1}{n}\sum_{i=1}^{N}\hat{\boldsymbol{x}}_k^{i-} \tag{4.26}$$

Kalman gain

$$\boldsymbol{K}_k = \boldsymbol{P}_k^-\boldsymbol{C}_k^T(\boldsymbol{C}_k\boldsymbol{P}_k^-\boldsymbol{C}_k^T + \boldsymbol{R})^{-1} \tag{4.27}$$

Measurement update

$$\hat{\boldsymbol{x}}_k^{i+} = \hat{\boldsymbol{x}}_k^{i-} + \boldsymbol{K}_k[\boldsymbol{y}_k - \boldsymbol{h}(\hat{\boldsymbol{x}}_k^{i-}) + \bar{\boldsymbol{v}}_k^i] \tag{4.28}$$

It is clearly seen from (4.23) and (4.24) that the number of times $\boldsymbol{f}_k(\cdot)$ must be calculated at each time step is reduced relative to the EKF from $n+1$ to N. The accuracy of the stochastic error covariance matrix depends on the ensemble size N, and for linear dynamics, the EnKF estimate actually converges exactly to the KF solution with increasing ensemble size (Evensen, 2009). For nonlinear dynamics, the EnKF estimate becomes closer to the EKF-estimate with increasing ensemble size. Hence, there is a trade off between having an accurate covariance matrix (large N) and a rapid state estimator (small N).

4.2 Model measurements

When estimating the states of a sea-ice model in real life, one needs to measure not only a selection of the model's state but also the different inputs that drives the model. In the case of the Hibler-model this would be the boundary conditions $\boldsymbol{\chi}$, the air temperature T_a, the geostrophic wind and current $\boldsymbol{U}_g$ and $\boldsymbol{U}_w$ as well as their respective turning angles ϕ and θ. As presented in Chapter 3, all of these inputs are assumed to be known, and measuring them is therefore not a topic in this thesis.

Since measurements are made at time instances, the system measurements $\boldsymbol{y}_k$ are discrete, and are in this thesis represented through the following linear equation

$$\boldsymbol{y}_k = \boldsymbol{C}\boldsymbol{x}_k \tag{4.29}$$

where $\boldsymbol{C}$ is referred to as the measurement matrix. The measurement matrix normally consists of both positive and negative numbers of all sizes, representing whether or not a state is measured, as well as the measurement gain. In this thesis, however, the measurement matrix will only consist of the numbers 0 and 1. The state is either measured or it is not. It is assumed that new measurements are made every hour.

4.2.1 Sensor platform

When measuring the states of the sea-ice model several sensor platforms may be applied, for example a satellite platform, an underwater platform or a UAV platform (Haugen et al., 2011). Even though all the platforms have their pros and cons, the UAV is emphasized as a strong candidate by Haugen et al. (2011), and it is also the only platform which is considered here.

The system states, which consist of u, v, h and A for every grid-cell, are all measurement candidates. With UAVs as sensor platform, each of these states can in theory be measured all at once, but in practice this will be both expensive and time consuming. It is the following therefore discussed which of the system states that easily can be measured together, and hence constitute a measurement-matrix candidate.

A UAV is an unmanned aircraft that flies without any human crew or pilot on board. It can either be remotely piloted (often referred to as a *drone*) or autonomous. Small UAVs are suitable for carrying cameras, sensors and communication equipment, and they can be used to scan the ice floes to obtain information about the ice properties and dimensions. The different sensor types suitable for measuring u, v, h and A from a UAV are presented in Haugen et al. (2011), and in general one can say that u, v and A can be measured by the same, fairly light sensor, while h needs another sensor which is somewhat heavier.

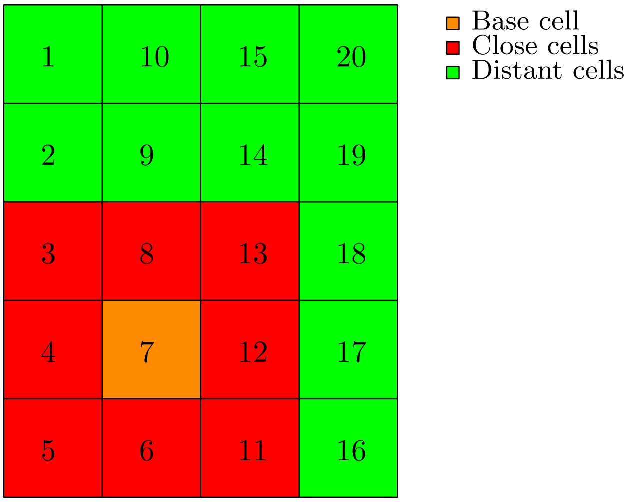
Some key properties of a UAV are weight, maximum payload, altitude, endurance, range and speed (Haugen et al., 2011). UAVs dimensioned for carrying more instruments (both in terms of space and weight) are often not able to match the endurance, altitude and speed of those dimensioned for carrying less instruments. The range of the UAVs dimensioned for carrying more instruments is therefore often shorter since they can fly for a shorter period of time before they must land again.

Given this, and assuming that the platform base is located in one particular grid cell (along with the drillship), one can construct an imaginary UAV sensor platform consisting three different UAVs:

- One that carries a fairly light instrument which measures u, v and A. This UAV can fly to all cells of the grid.
- One that carries a somewhat heavier instrument which measures h. This UAV can also fly to all cells of the grid.
- One that carries a fairly light instrument measuring u, v and A and another, somewhat heavier instrument, measuring h. This UAV can only fly to cells adjacent to the base cell.

4.2.2 Measurement-matrix candidates

Given the implemented Hibler-model, let cell number 7 be the base cell. The first step in determining some measurement-matrix candidates is to divide the grid cells into *close cells* and *distant cells*. The close cells are cells adjacent to the base cell, defining the remaining cells as distant cells, see Figure 4.4



🟩 1	🟩 10	🟩 15	🟩 20
🟩 2	🟩 9	🟩 14	🟩 19
🟥 3	🟥 8	🟥 13	🟩 18
🟥 4	🟧 7	🟥 12	🟩 17
🟥 5	🟥 6	🟥 11	🟩 16

🟧 Base cell
🟥 Close cells
🟩 Distant cells

Figure 4.4: Grid sections.

For simplicity it is decided that when the UAV is sent out to make measurements it only measures one cell, and then it returns back to base. In practice this will most likely not be the case, but in order to avoid factors such as current location of UAV, remaining flight time and so on, this choice is made. Since the UAV measures either $\{u, v, A\}$ or $\{h\}$ for distant cells and $\{u, v, h, A\}$ for close cells, there are 31 different measurement matrix candidates. These are all listed in Table 4.1.

4.2.3 Spatial location and the importance of an accurate estimate

Not every state of the state vector $\boldsymbol{x}$ is of equal importance when estimating the states of the sea-ice model. It is quite intuitive that it is more important to have accurate information about the ice conditions near the drillship, since the drillship is directly exposed to these conditions. It is, of course, also important to have accurate information about the ice conditions further away from the drillship as well, especially when it comes to the part of ice management which concern the removal of larger icebergs. But in general one can say that the closer the grid cell is to the drillship, the more important it is to have accurate knowledge of its cell states.

The main ice-drift direction also plays an important role when it comes to which areas that are more important to have proper estimates in. The main ice-drift direction is a result of the ocean current direction and wind direction,

Measurement matrix	Cell	Measurements	Measurement matrix	Cell	Measurements
$\boldsymbol{C}_1$	1	u, v, h, A	$\boldsymbol{C}_{16}$	10	u, v, A
$\boldsymbol{C}_2$	2	u, v, h, A	$\boldsymbol{C}_{17}$	10	h
$\boldsymbol{C}_3$	3	u, v, h, A	$\boldsymbol{C}_{18}$	14	u, v, A
$\boldsymbol{C}_4$	6	u, v, h, A	$\boldsymbol{C}_{19}$	14	h
$\boldsymbol{C}_5$	7	u, v, h, A	$\boldsymbol{C}_{20}$	15	u, v, A
$\boldsymbol{C}_6$	8	u, v, h, A	$\boldsymbol{C}_{21}$	15	h
$\boldsymbol{C}_7$	11	u, v, h, A	$\boldsymbol{C}_{22}$	16	u, v, A
$\boldsymbol{C}_8$	12	u, v, h, A	$\boldsymbol{C}_{23}$	16	h
$\boldsymbol{C}_9$	13	u, v, h, A	$\boldsymbol{C}_{24}$	17	u, v, A
$\boldsymbol{C}_{10}$	4	u, v, A	$\boldsymbol{C}_{25}$	17	h
$\boldsymbol{C}_{11}$	4	h	$\boldsymbol{C}_{26}$	18	u, v, A
$\boldsymbol{C}_{12}$	5	u, v, A	$\boldsymbol{C}_{27}$	18	h
$\boldsymbol{C}_{13}$	5	h	$\boldsymbol{C}_{28}$	19	u, v, A
$\boldsymbol{C}_{14}$	9	u, v, A	$\boldsymbol{C}_{29}$	19	h
$\boldsymbol{C}_{15}$	9	h	$\boldsymbol{C}_{30}$	20	u, v, A
			$\boldsymbol{C}_{31}$	20	h

Table 4.1: Measurement-matrix candidates.

and knowledge of the ice conditions upstream from the drillship is far more important than those downstream, naturally.

Figure 4.5 shows the drill ship located in cell number 7, with ice surrounding it. It is assumed that ice-management is being carried out, so that the ice-floes near the ship are smaller than those far away from it. However, for simplicity, the ice breakers included in this ice management are left out. The figure shows three circles of different radius, representing zones of different importance, with the area within the red circle as the most important one. In addition to this, the upstream zone of each circle is highlighted, indicating that these three areas are more important than the remaining part of the circles. Comparing Figure 4.4 and Figure 4.5, it can be seen that, in addition to cell number 7, cell number 2, 3, 4, 8 and 9 are somewhat more important to estimate that the rest.

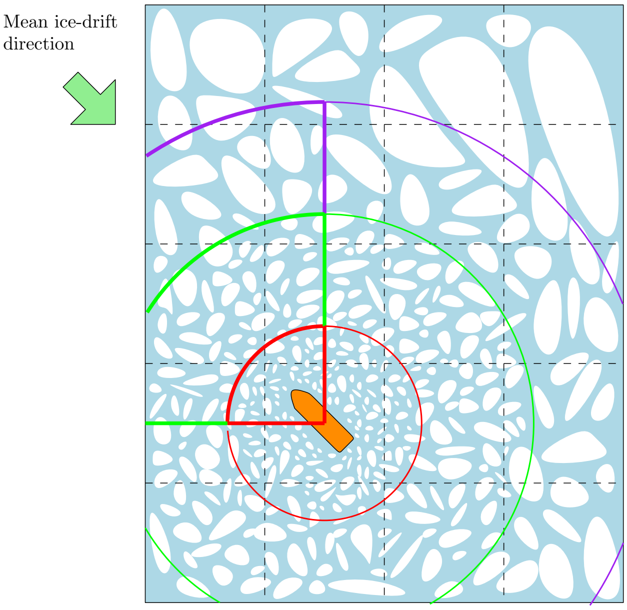


Figure 4.5: Important areas to identify cell states in.

4.3 Adding noise to the system

The full system consisting of the Hibler-model and the measurement vector is a dynamic system with continuous dynamics and discrete measurements.

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\Upsilon}) \tag{4.30a}$$

$$\boldsymbol{y}_k = \boldsymbol{C}\boldsymbol{x}_k \tag{4.30b}$$

In order to evaluate the performance of the EnKF, one needs to include process and measurement noise to the system. This process and measurement noise represents unmodeled process features and uncertainty introduced by the measurement methods, respectively. Both the process and measurement noise is additive, and they are included as if the system was discrete, through the following equations.

$$\boldsymbol{x}_{k+1} = \boldsymbol{f}_d\left(\boldsymbol{x}_k, \boldsymbol{\Upsilon}\left(t\right)\right) + \boldsymbol{w}_k \tag{4.31a}$$

$$\boldsymbol{y}_k = \boldsymbol{C}\boldsymbol{x}_k + \boldsymbol{v}_k \tag{4.31b}$$

where $\boldsymbol{w}_k$ and $\boldsymbol{v}_k$ are vectors of appropriate dimensions containing zero-mean white noise, and $\boldsymbol{f}_d(\cdot)$ si the discretized $\boldsymbol{f}(\cdot)$.

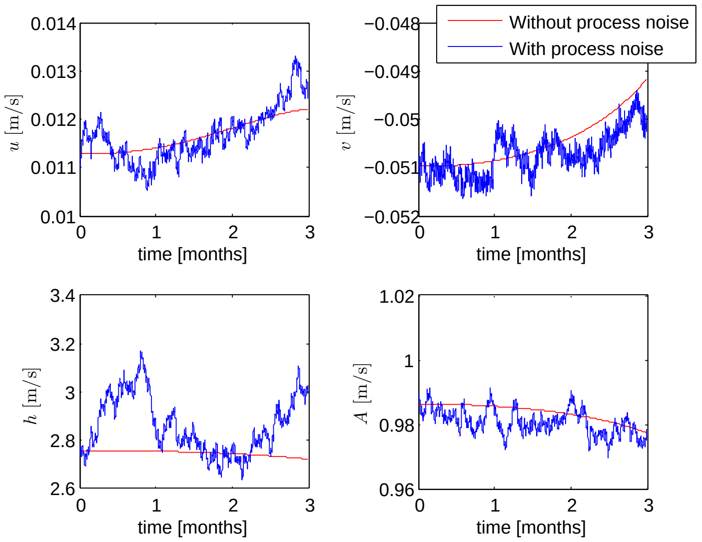


Figure 4.6: The states of cell number 7 with and without process noise.

However, the Hibler-model is a complex model, and discretizing it by finding the explicit expression for $\frac{\partial \boldsymbol{f}}{\partial x}$ would be time consuming and potentially quite difficult, or even impossible. The model was therefore discretized by simply using the MATLAB-solver `ode15s` to simulate from one time step to the next, with the previous state vector $\boldsymbol{x}_k$ as initial values. The system dynamics are therefore still continuous (the input Υ is for instance not constant through one time step), but from the noise's point of view it is discrete. The time step of the "discrete" system is set to 1 hour.

Process noise

The variance of the process noise is set to

$$\text{var}\left(\boldsymbol{w}_k\right) = \left[\text{var}\left(\boldsymbol{w}_{k,1}\right)^T \quad \cdots \quad \text{var}\left(\boldsymbol{w}_{k,p}\right)^T\right]^T \tag{4.32}$$

where p is the number of grid cells and

$$\text{var}(\boldsymbol{w}_{k,i}) = \text{var}\left[u_i \quad v_i \quad h_i \quad A_i\right]^T = \left[10^{-5} \quad 10^{-4} \quad 10^{-2} \quad 10^{-4}\right]^T \tag{4.33}$$

which it is equal for every cell i. These values were chosen based on simulating with and without process noise to see how much it effected the states. Plots of all the cell states with and without process noise can be seen in Appendix D. Here only the results for cell number 7 will be presented, see Figure 4.6. It clearly shows that the noise has an impact on the states, but not to severe, this is also the case for the other cells.

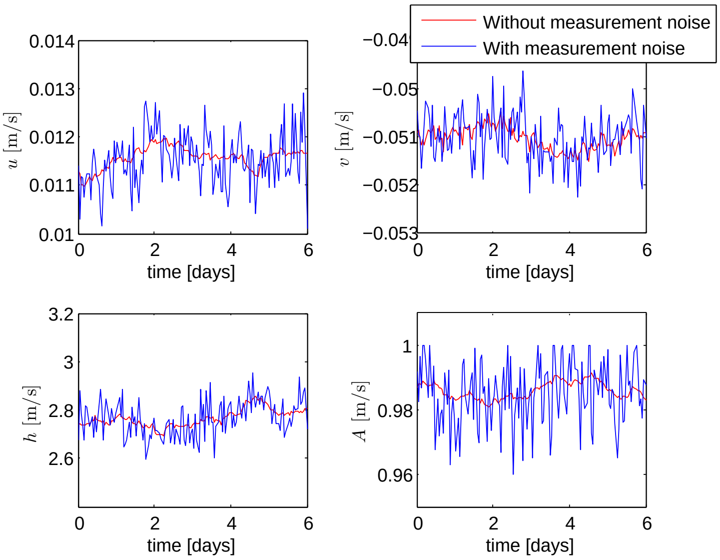


Figure 4.7: The states of cell number 7 (with process noise) with and without measurement noise.

The system was only simulated once, and the process noise is therefore equal for every Kalman filter.

Measurement noise

The dimension of the measurement noise $\boldsymbol{v}$ varies of course with the chosen measurement matrix, so in order to be able to use the same measurement noise for every measurement matrix, (4.31b) was rewritten to

$$\boldsymbol{y}_k = \boldsymbol{C}\boldsymbol{x}_k + \boldsymbol{C}\boldsymbol{v}_k = \boldsymbol{C}\left(\boldsymbol{x}_k + \boldsymbol{v}_k\right) \tag{4.34}$$

Thus, the measurement noise for every state is generated at once, and they are only generated one time. The variance of the measurement noise was set to

$$\mathrm{var}\left(\boldsymbol{v}_k\right) = \begin{bmatrix}\mathrm{var}\left(\boldsymbol{v}_{k,1}\right)^T & \cdots & \mathrm{var}\left(\boldsymbol{v}_{k,p}\right)^T\end{bmatrix}^T \tag{4.35}$$

where p is the number off grid cells and

$$\mathrm{var}(\boldsymbol{v}_{k,i}) = \mathrm{var}\begin{bmatrix}u_i & v_i & h_i & A_i\end{bmatrix}^T = \begin{bmatrix}10^{-4} & 10^{-4} & 0.06 & 0.01\end{bmatrix}^T \tag{4.36}$$

which is also equal for every cell i. These values were chosen based on the magnitude of the different states. See Figure 4.7 for plots of u, v, h and A of cell number 7 with and without measurement noise. When adding the measurement noise, actions were made to ensure that none of the As with measurement noise exceed $A = 1$.

4.4 Implementing the ensemble Kalman filter

In this section, the EnKF described in Section 4.1.3 will be implemented and tested for some of the measurement-matrix candidates presented in Section 4.2. The reason that the EnKF was chosen is that the system has many states, even with this quite small grid, and simulating many perturbations to find the partial derivatives in an EKF would be too time consuming.

Two of the measurement-matrices are chosen: $\boldsymbol{C}_3$ and $\boldsymbol{C}_{23}$, resulting in two different EnKFs. These two measurement matrices are chosen to attempt to show how the EnKF estimate depends on both the cell states that are being measured and the spatial location of the measured cells.

4.4.1 The EnKF system model

The model used in the EnKF is very similar to that used to simulate the “real world” with process and measurement noise. The continuous Hibler-model is discretized in the same way, the only difference being that the model this time has constant input $\boldsymbol{\Upsilon}$ through one time step, and the EnKF is thus based on a model which is entirely discrete. Also this time a time step of 1 hour was used.

Notice that the process and measurement noise of the EnKF are not the same as the process and measurement noise of the system. The process and measurement noise of the EnKF are tuning parameters which indicate how much one trusts the model and the measurements, respectively. The more one trusts the model, the less process noise is added, and the more one trusts the measurements, the less measurement noise is added. Hence, there is a connection between $\{\bar{\boldsymbol{w}}_k, \bar{\boldsymbol{v}}_k\}$ and $\{\boldsymbol{w}, \boldsymbol{v}_k\}$, but they do not necessarily share the same variance.

4.4.2 Tuning the EnKF

The EnKF has several tuning parameters;

- the number of ensemble members N,
- the initial value of the ensemble members $\hat{\boldsymbol{X}}_0$,

- the measurement noise $\bar{\boldsymbol{v}}$ and its covariance matrix $\boldsymbol{R}$ (these must of course coincide), and the process noise $\bar{\boldsymbol{w}}$.

The number of ensemble members and their initial values are common for all three EnKF, and are presented here, while $\boldsymbol{R}$, $\bar{\boldsymbol{v}}$ and $\bar{\boldsymbol{w}}$ vary somewhat from filter to filter, and are therefore presented individually later on.

There is no definite way of deciding how many members the ensemble should have. On one hand it needs to be large enough so that the scheme of the EnKF works, on the other hand it must be small enough so that the computational benefits of the EnKF are preserved. With a total of 80 states (4 per cell), the choice of ensemble members landed on $N = 10$. With 10 ensemble members, the the amount of integrations needed is much smaller than with an EKF and it is also big enough to give a reasonable ensemble representation of the error covariance matrix.

When it comes to choosing the initial ensemble, Evensen (2003) suggests that one should create the initial ensemble by adding some perturbation to an initial best-guess estimate, and then integrate the ensemble over a time to ensure that the system is in dynamical balance, and that proper multivariate correlations have developed. The initial best guess to perturb around is chosen to be

$$\bar{\boldsymbol{x}}_0 = \begin{bmatrix} 0 & 0 & 2.5 & 0.9 & \cdots & 0 & 0 & 2.5 & 0.9 \end{bmatrix}^T \tag{4.37}$$

The perturbation was done so that the different perturbed states of the state vector still was somewhat physically correct relative each other, and the magnitude of the perturbation was chosen so that it represented the uncertainty of the initial best guess $\bar{\boldsymbol{x}}_0$. The entire initial ensemble $\hat{\boldsymbol{X}}_0$ can be seen in Appendix C.

Each of these ensemble members were then simulated with constant air temperature $T_a = -20$ for 24 hours in order to insure dynamical balance. The results from these simulations are used as the actual initial ensemble.

4.4.3 C$_3$ as measurement matrix

$\boldsymbol{C}_3$ is the measurement matrix that measures u, v, h and A in cell number 3 of Figure 4.4.

The measurement noise $\bar{\boldsymbol{v}}$ and the process noise $\bar{\boldsymbol{w}}$ of the EnKF are zero mean white noise with the following variances:

$$\text{var}\left(\bar{\boldsymbol{w}}_k\right) = \begin{bmatrix}\text{var}(\bar{\boldsymbol{w}}_{k,1})^T & \cdots & \text{var}(\bar{\boldsymbol{w}}_{k,p})^T\end{bmatrix}^T \tag{4.38}$$

$$\text{var}\left(\bar{\boldsymbol{v}}_k\right) = \begin{bmatrix}\text{var}(\bar{\boldsymbol{v}}_{k,1})^T & \cdots & \text{var}(\bar{\boldsymbol{v}}_{k,p})^T\end{bmatrix}^T \tag{4.39}$$

where p is the number of grid cells and

$$\text{var}(\bar{\boldsymbol{w}}_{k,i}) = \text{var}\begin{bmatrix}u_i & v_i & h_i & A_i\end{bmatrix}^T = \begin{bmatrix}10^{-4} & 10^{-4} & 10^{-2} & 10^{-3}\end{bmatrix}^T \tag{4.40}$$

$$\text{var}(\bar{\boldsymbol{v}}_{k,i}) = \text{var}\begin{bmatrix}u_i & v_i & h_i & A_i\end{bmatrix}^T = \begin{bmatrix}5\cdot 10^{-4} & 10^{-3} & 0.6 & 0.01\end{bmatrix}^T \tag{4.41}$$

which is equal for every cell i. These variances were found by first tuning the EnKF for the system simulations with neither process nor measurement noise. Starting with $\text{var}(\bar{\boldsymbol{w}}_{k,i}) = \text{var}(\boldsymbol{w}_{k,i})$ and $\text{var}(\bar{\boldsymbol{v}}_{k,i}) = \text{var}(\boldsymbol{v}_{k,i})$, the variance of $\bar{\boldsymbol{w}}_{k,i}$ and $\bar{\boldsymbol{v}}_{k,i}$ was altered until the best estimate was attained. Then, some further tuning was done for the system simulations where both process and measurement noise were included. The covariance matrix of the measurement noise $\boldsymbol{R}$, is given directly from $\bar{\boldsymbol{v}}$.

How many hours ahead that need to be estimated accurately depends how long it takes to prepare the drillship for heavy ice forces, and the time it takes to remove icebergs et cetera, that is in danger of colliding with the drillship. How long it takes to prepare the drillship for heavy ice forces depends on the drillship as well as the ice environment and weather conditions in the operating area. In this thesis the system states are estimated 14 days, starting at day number 80 (approximately after 2.7 months). This probably far exceeds the required estimate window, but the reason that the system states are estimated for so long is so that there is time for the estimates to converge and follow the state.

The overall result of the estimation is given by means of the scaled estimate error, $\boldsymbol{e}_s$. The estimate error is defined as the error between the system simulation with process noise $\boldsymbol{x}$, and the EnKF estimate $\hat{\boldsymbol{x}}$. This error is then scaled by $\boldsymbol{x}$:

$$e_{s,i} = \frac{x_i - \hat{x}_i}{x_i} = \frac{e_i}{x_i} \tag{4.42}$$

where $i = 1, \cdots, n$, and n is the number of system states.

Plots of all the scaled state errors can be seen in Figure 4.8. It clearly shows that overall, the error stays relatively close to zero. The estimated $\boldsymbol{u}$, $\hat{\boldsymbol{u}}$, deviates about $5 - 10\%$ from the actual $\boldsymbol{u}$, $\hat{\boldsymbol{v}}$ deviates about $2 - 4\%$ from the actual $\boldsymbol{v}$, $\hat{\boldsymbol{h}}$ deviates about $5 - 15\%$ from the actual $\boldsymbol{h}$ and $\hat{\boldsymbol{A}}$ deviates

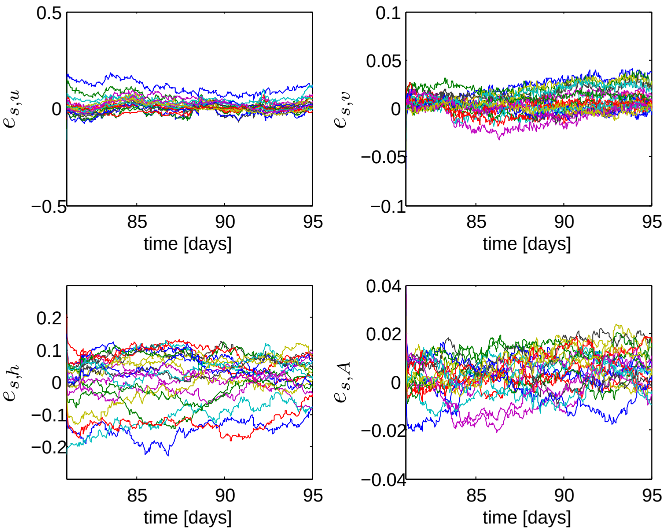


Figure 4.8: Scaled errors from estimating with $\boldsymbol{C} = \boldsymbol{C}_3$.

about $1-2\%$ from the actual $\boldsymbol{A}$. Hence, it is the estimates of $\boldsymbol{u}$ and $\boldsymbol{h}$ which deviate the most from the actual values. This is also supported by the sum of the euclidean norm of each property's scaled-error time vector

$$e_{tot,u} = \|\boldsymbol{e}_{s,u1}\|_2 + \cdots + \|\boldsymbol{e}_{s,u20}\|_2 \tag{4.43}$$

$$e_{tot,v} = \|\boldsymbol{e}_{s,v1}\|_2 + \cdots + \|\boldsymbol{e}_{s,v20}\|_2 \tag{4.44}$$

$$e_{tot,h} = \|\boldsymbol{e}_{s,h1}\|_2 + \cdots + \|\boldsymbol{e}_{s,h20}\|_2 \tag{4.45}$$

$$e_{tot,A} = \|\boldsymbol{e}_{s,A1}\|_2 + \cdots + \|\boldsymbol{e}_{s,A20}\|_2 \tag{4.46}$$

which is the following for the EnKF with $\boldsymbol{C} = \boldsymbol{C}_3$.

$$e_{tot,u} = 56.3312 \tag{4.47}$$

$$e_{tot,v} = 18.8200 \tag{4.48}$$

$$e_{tot,h} = 116.3967 \tag{4.49}$$

$$e_{tot,A} = 13.3633 \tag{4.50}$$

Overall, the EnKF based on $\boldsymbol{C} = \boldsymbol{C}_3$ is thus worst at estimating h. However, also this stays within 20% of the actual values, which is a decent, but not particularly good estimate. Some of the estimates of u also deviates quite a lot, but many of them keep within 5% deviation, which is not bad. All the estimates of both v and A stay well within 5% deviation, and can be considered good estimates. Further details of the estimation results will be given later on, when comparing the estimation results from both $\boldsymbol{C} = \boldsymbol{C}_3$ and $\boldsymbol{C} = \boldsymbol{C}_{23}$.

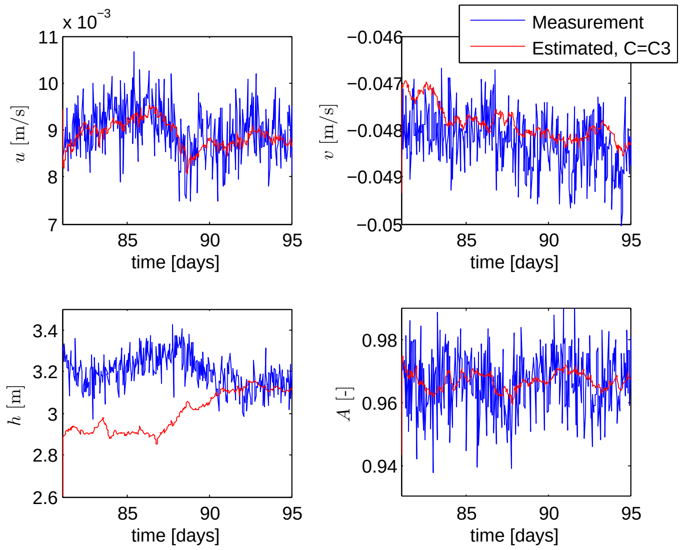


Figure 4.9: Measured and estimated states in cell number 3.

Plots of the measurements made in cell number 3 versus the estimates of the states in cell number 3 are given in Figure 4.9, it clearly shows (even though the h-estimate is somewhat off) that the measurement noise is filtered out by the EnKF.

4.4.4 C$_{23}$ as measurement matrix

$\boldsymbol{C}_{23}$ is the measurement matrix that measures h in cell number 16 of Figure 4.4. The measurement noise $\bar{\boldsymbol{v}}$ and the process noise $\bar{\boldsymbol{w}}$ of the EnKF were set to

$$\operatorname{var}(\bar{\boldsymbol{w}}_k) = \begin{bmatrix}\operatorname{var}(\bar{\boldsymbol{w}}_{k,1})^T & \cdots & \operatorname{var}(\bar{\boldsymbol{w}}_{k,p})^T\end{bmatrix}^T \tag{4.51}$$

$$\operatorname{var}(\bar{\boldsymbol{v}}_k) = \begin{bmatrix}\operatorname{var}(\bar{\boldsymbol{v}}_{k,1})^T & \cdots & \operatorname{var}(\bar{\boldsymbol{v}}_{k,p})^T\end{bmatrix}^T \tag{4.52}$$

where p is the number of grid cells and

$$\operatorname{var}(\bar{\boldsymbol{w}}_{k,i}) = \operatorname{var}\begin{bmatrix}u_i & v_i & h_i & A_i\end{bmatrix}^T = \begin{bmatrix}10^{-5} & 10^{-5} & 10^{-2} & 10^{-3}\end{bmatrix}^T \tag{4.53}$$

$$\operatorname{var}(\bar{\boldsymbol{v}}_{k,i}) = \operatorname{var}\begin{bmatrix}u_i & v_i & h_i & A_i\end{bmatrix}^T = \begin{bmatrix}10^{-4} & 10^{-4} & 0.06 & 0.01\end{bmatrix}^T \tag{4.54}$$

which is equal for every cell i. These were tuned in a similar manner as with $\boldsymbol{C} = \boldsymbol{C}_3$. However, since only h is being measured, there is no point in altering u_i, v_i and A_i of $\operatorname{var}(\bar{\boldsymbol{v}}_{k,i})$. The covariance matrix of the measurement noise $\boldsymbol{R}$ is given directly from $\bar{\boldsymbol{v}}$.

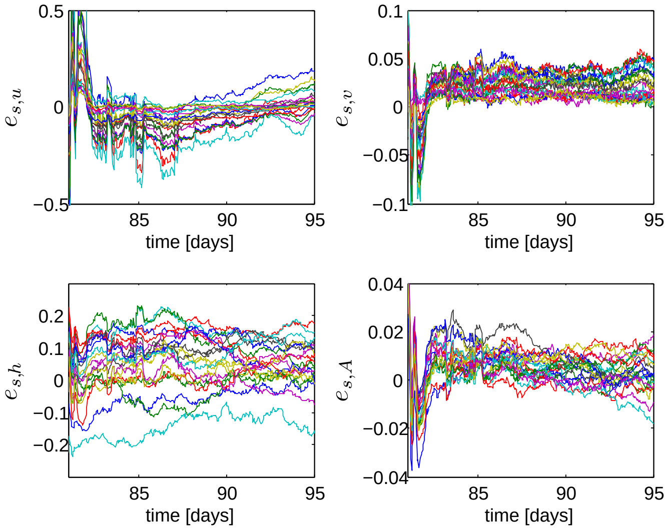


Figure 4.10: Scaled errors from estimating with $\boldsymbol{C} = \boldsymbol{C}_{23}$.

Plots of all the scaled state errors can be seen in Figure 4.10. It shows that also here, the error stays relatively close to zero. After the estimates has converged, $\hat{\boldsymbol{u}}$ deviates around $10 - 20\%$ from the actual $\boldsymbol{u}$, $\hat{\boldsymbol{v}}$ around $4 - 6\%$ from the actual $\boldsymbol{v}$, $\hat{\boldsymbol{h}}$ around $10 - 20\%$ from the actual $\boldsymbol{h}$ and $\hat{\boldsymbol{A}}$ around 2% from the actual $\boldsymbol{A}$. Hence, it is again the estimates of $\boldsymbol{u}$ and $\boldsymbol{h}$ which deviate the most from the the actual values. This is again supported by $\boldsymbol{e}_{tot}$. For the EnKF with $\boldsymbol{C} = \boldsymbol{C}_{23}$, the values are

$$e_{tot,u} = 152.4857 \tag{4.55}$$

$$e_{tot,v} = 39.2577 \tag{4.56}$$

$$e_{tot,h} = 148.0657 \tag{4.57}$$

$$e_{tot,A} = 13.4263 \tag{4.58}$$

Even though the numbers stated above would imply that the EnKF based on $\boldsymbol{C}_{23}$ is overall worst at estimating u, it could be argued by investigating Figure 4.10, that this is not case. Initially, the u-errors deviate quite a lot, but as they converge they seem to stay closer to zero than the h-errors. This implies that also the EnKF based on $\boldsymbol{C}_{23}$ is overall worst at estimating h. The u-estimates are not particularly good either, stying within 20% deviation. Some of the u-estimates does stay closer than this, but there are many which do not. The estimates of v and A are however quite good, stying within 6% and 2% deviation, respectively. Further details of the estimation results will be given in the next section, when comparing the estimation results from both $\boldsymbol{C} = \boldsymbol{C}_3$ and $\boldsymbol{C} = \boldsymbol{C}_{23}$.

The measured h in cell number 16 and its estimate can be seen in Figure 4.11.

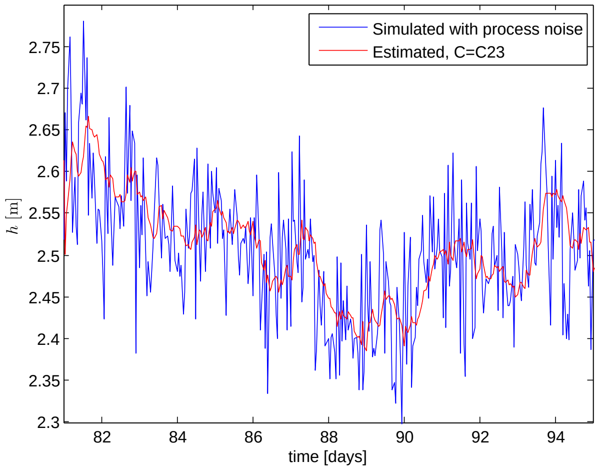


Figure 4.11: Measured and estimated h in cell number 16.

It clearly shows that the measurement noise is removed from the estimate by the EnKF.

4.4.5 Comparing the C$_3$ and C$_{23}$ estimates

As seen from the two previous sections, it is probably the estimates based on $\boldsymbol{C}_3$ which results in the overall best state estimator. But, as will be returned to in Chapter 5, it is not the overall best estimate that is of interest here. In this thesis, it is desirable to unveil *which states* that are better estimated using different measurement matrices, and this is therefore discussed in the following. The EnKF based on $\boldsymbol{C}_3$ is defined as EnKF$_3$ and the EnKF based on $\boldsymbol{C}_{23}$ as EnKF$_{23}$. Plots of all states can be seen in Appendix E, as only a selection of them are presented here.

A cell that represents a typical result is cell number 13, which can be seen in Figure 4.12. It shows that EnKF$_3$ is better at estimating u and v, while both EnKF$_3$ and EnKF$_{23}$ are somewhat off when estimating h, and equally good when estimating A. Generally, EnKF$_{23}$ also oscillates much more than EnKF$_3$ in the beginning, except for the h-estimates. This has the natural explanation that it has no direct measurements of any u, v or As, so it takes longer until it is able to "tune in", while EnKF$_3$ gets direct measurements of all four properties in cell number 3, and this oscillation does not appear. However, this is a typical result. Meaning that there are several other cells where this is not the case. Some individual cases, both confirming and denying this typical behavior, will be presented next.

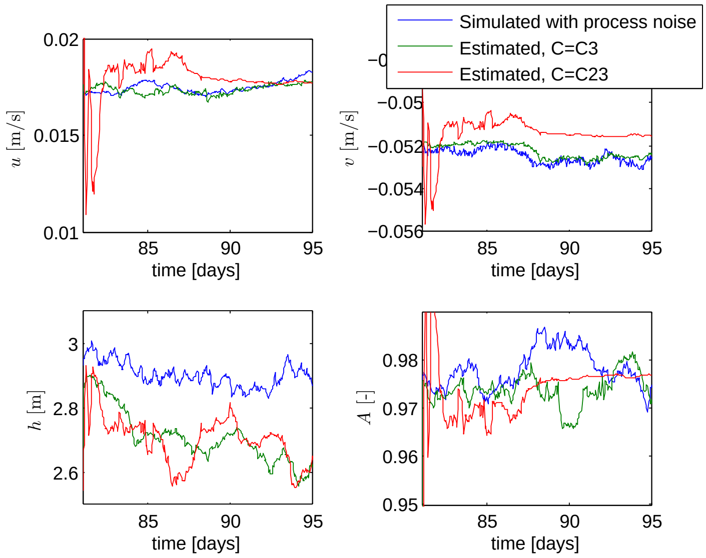


Figure 4.12: Simulated and estimated states in cell number 13.

Table 4.2 lists some of the states that EnKF$_3$ and EnKF$_{23}$ provides good estimates for. These are further elaborated in the following.

	Cell nr.	u	v	h	A
Good estimate with $\boldsymbol{C}_3$	3	$\surd$	$\surd$		$\surd$
	8		$\surd$		
	10	$\surd$			
	15		$\surd$		
Good estimate with $\boldsymbol{C}_{23}$	16			$\surd$	
	17			$\surd$	
	20	$\surd$		$\surd$	

Table 4.2: Summary of the well estimated states presented here.

Figure 4.13 and 4.14 shows the simulated and estimated states of cell number 3 and 16, respectively. They both show that each EnKF estimates their respective measured states better than the other EnKF, which is expected. However, u, v and A of Figure 4.14 also shows that, apart from the constant deviation in A, EnKF$_3$ estimates these states better than EnKF$_{23}$. This is probably because $\boldsymbol{C}_3$ gains some general knowledge of all u, v and As by measuring them in cell 3, while $\boldsymbol{C}_{23}$ receives no direct information about these properties whatsoever.

When it comes to the remaining cells, it is safe to say that EnKF$_3$ is better

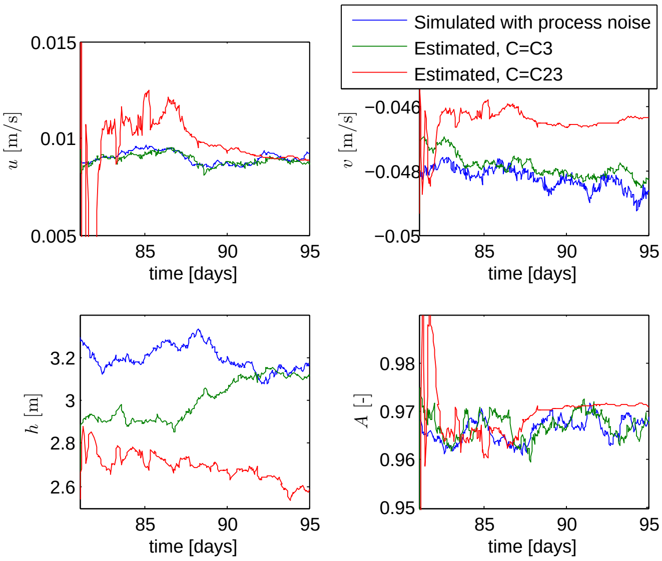


Figure 4.13: Simulated and estimated states in cell number 3.

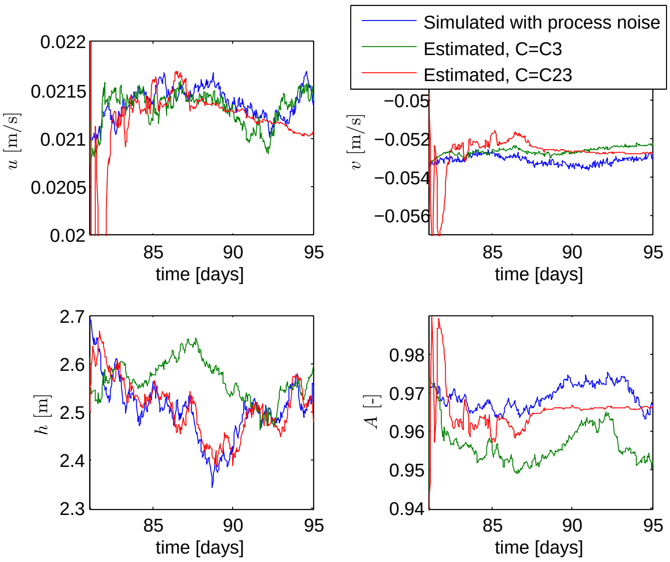
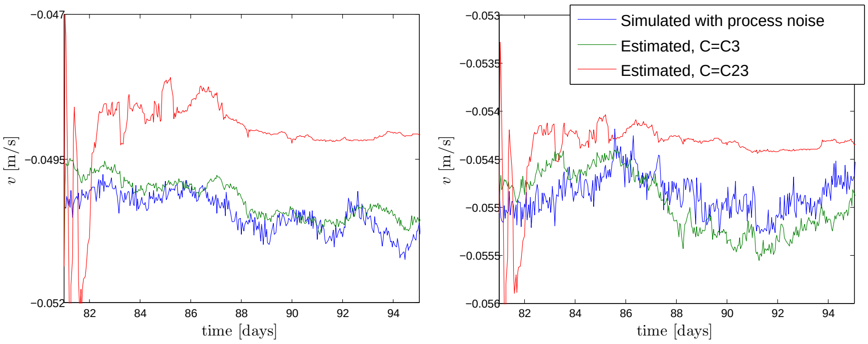


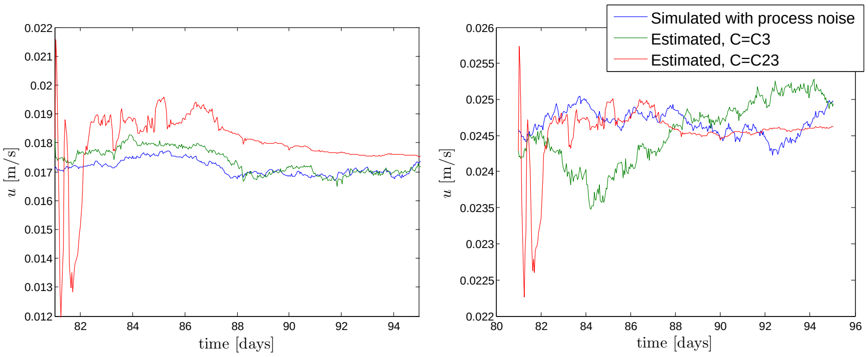
Figure 4.14: Simulated and estimated states in cell number 16.



(a) Cell number 8.

(b) Cell number 15.

Figure 4.15: Simulated and estimated v.

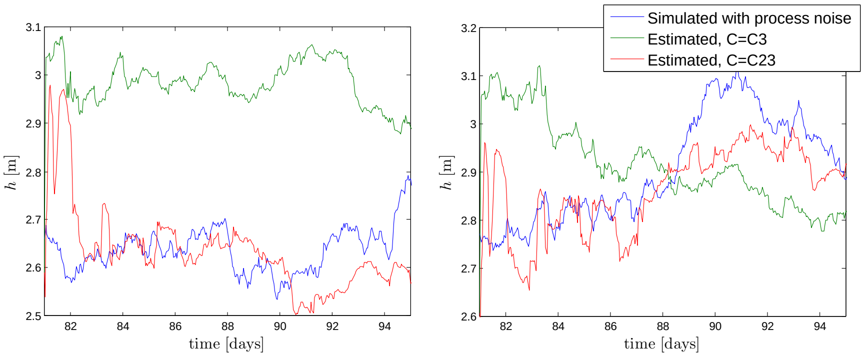


(a) Cell number 10.

(b) Cell number 20.

Figure 4.16: Simulated and estimated u.

at estimating v. In 11 out of 20 cells, EnKF_3 clearly outperforms EnKF_{23}. Figure 4.15 shows two examples (cell number 8 and 15) where EnKF_3's estimate is very good, and clearly outperforms that of EnKF_{23}. There are no examples where EnKF_{23} outperforms EnKF_3 with regards to estimating v. When it comes to estimating u, EnKF_3 is often better, but there are also examples where EnKF_{23}'s estimate is best. Figure 4.16 shows u in cell number 10 and 20. In cell 10, EnKF_3 is better and in cell 20 EnKF_{23} is better. Both of the EnKFs had some difficulties when estimating h in many cells, but there are some examples where EnKF_{23} shows good results. The simulated and estimated values of h in cell number 17 and 20 can be seen in Figure 4.17. In both of these cells, EnKF_{23} estimates h quite well, and outperforms EnKF_3. When estimating A, both EnKFs are close to the actual value. The



(a) Cell number 17.

(b) Cell number 20.

Figure 4.17: Simulated and estimated h.

estimate of EnKF$_3$ often follows the simulated curve better, but EnKF$_{23}$ is not far off, either. The results from cell 13 in Figure 4.12, is a good example of this.

EnKF$_3$ thus estimates not only u, v and A in cell 3 very well, but also (among others) v in cell 8 and 15 and u in cell 10. As expected, EnKF$_{23}$ estimates h in cell 16 very well, it also provides quite good estimates of u in cell 20 and h in cell 17 and 20 as well. See summary in Table 4.2. Apparently, it is not only the states that are actually measured that are better estimated by one EnKF than the other, but also other, perhaps seemingly random states.

4.5 Chapter summary

In this chapter the linear Kalman filter and the extended/ensemble Kalman filter have been presented as state estimators for linear and nonlinear systems, respectively. For large, complex systems, the EnKF is often superior to the EKF since the EKF often becomes a time consuming state estimator for such systems. The EnKF is based on a scheme which shortens the calculation time at each time step, making it much more applicable for large, complex systems, such as systems containing the Hibler-model.

A UAV-based platform is also presented as a possible platform for making measurements, and based on this, a suggestion for measurement-matrix candidates were given. The importance of accurate estimates upstream and close to the ship were also emphasized.

The EnKF was implemented and tested for two different measurement matrices, resulting in two different EnKFs: EnKF_3 and EnKF_{23}. In general one can say that the EnKF_3 gave the overall best estimate, but both of them produced estimates whose scaled error stayed relatively close to zero, and both of them estimated the measured states very well (except one). In addition to this, they both produced good estimates of different other states.

Chapter 5

Method for Selecting Measurement Matrix On-Line

In this chapter, a new way of deciding on-line which measurement matrix to apply, will be presented. This method is based on combing information from the Kalman filter covariance matrix and the observability gramian of the systems based on the different measurement matrices. Both of these topics are therefore revisited, and how they may contribute when choosing $\boldsymbol{C}$ is explained in Section 5.2 and 5.3, respectively.

A new measure for degree of observability based on the observability gramian is introduced, before the method which combines the covariance matrix and observability gramian in order to select a $\boldsymbol{C}$-matrix on-line, is presented in section 5.4.

5.1 The optimal measurement matrix

As seen by the estimates described in Section 4.4, different measurement matrices can be “best” at estimating different states, and singling out one measurement matrix as optimal for estimating the entire state vector can be difficult. There are of course some measurement matrices which lead to estimators that overall perform better than others, but in general it is safe to say that there is no optimal measurement matrix for a system as large and complex as the one presented in this thesis.

It is therefore not a question of selecting one optimal measurement matrix, but choosing the measurement matrix that is best suited at the current time.

An on-line method for selecting which measurement matrix to apply should be based on information about which states that are more important to estimate accurately (see Section 4.2.3) and information about which states that are poorly estimated at the current time. If a state that is important to estimate accurately is poorly estimated, a measurement matrix containing information about that state needs to be applied.

In the following, the question of how to obtain information about poorly estimated states at the at current time is treated, before the use of the observability gramian to determine the state information connected to each measurement matrices is discussed.

5.2 The Kalman filter covariance matrix

As mentioned in Section 4.1.1, the error covariance matrix of the Kalman filter is defined as the covariance of the error between the estimated and actual state vector.

$$\boldsymbol{P}_k = E\left[\left(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k\right)\left(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k\right)^T\right] \tag{5.1}$$

Defining $\boldsymbol{e}_k = \boldsymbol{x}_k - \hat{\boldsymbol{x}}_k$, $\boldsymbol{P}_k$ can be written out as

$$\boldsymbol{P}_k = E\begin{bmatrix} e_{k,1}^2 & e_{k,1}e_{k,2} & \cdots & e_{k,1}e_{k,n} \\ e_{k,1}e_{k,2} & e_{k,2}^2 & \cdots & e_{k,2}e_{k,n} \\ \vdots & \vdots & \ddots & \vdots \\ e_{k,1}e_{k,n} & e_{k,2}e_{k,n} & \cdots & e_{k,n}^2 \end{bmatrix} \tag{5.2}$$

The diagonal elements of $\boldsymbol{P}_k$ are hence the variance of the error between the estimated and actual state vector. The variance of $\boldsymbol{e}_k$ for an $\hat{\boldsymbol{x}}_k$ with values that are close to $\boldsymbol{x}_k$ is smaller than for an $\hat{\boldsymbol{x}}_k$ with values that differ considerably from $\boldsymbol{x}_k$. The diagonal elements of $\boldsymbol{P}_k$ therefore reflects the uncertainty of the estimate $\hat{\boldsymbol{x}}$.

States with a corresponding higher diagonal term in $\boldsymbol{P}_k$ thus have a weak estimate, and the covariance matrix of the Kalman filter can be applied on-line to determine which states that are poorly estimated at time k. Even though the covariance matrix of the EnKF is an approximation of the actual estimate-error covariance, it represents the same information and is therefore fully applicable for finding poorly estimated states.

5.3 The observability gramian

As mentioned earlier, Ersdal (2010) discusses how the observability gramian can be used to define MDOs, and also tests how these MDOs can be used to select an optimal measurement matrix. However, in Ersdal (2010), this choice is made off-line, and the state estimator (a Luenberger observer) uses this measurement matrix throughout the entire estimation.

The MDOs discussed in Ersdal (2010) are therefore not applicable when the measurement matrices are chosen on-line, and knowledge about the need for information about certain states is incorporated in the decision. The reason for this is that all of these MDOs consider every state as equally important, and the aim for them is to choose the measurement matrix that provides an overall best estimate, without favoring certain states. Here, a new method that withdraws knowledge about how much information a measurement matrix can provide about each individual state is needed.

5.3.1 Properties of the observability gramian

The linear observability gramian $\boldsymbol{W}_{o,linear}$ and the empirical observability gramian $\boldsymbol{W}_o$ were defined in Section 2.1.1. The output energy of a linear system (with real valued system matrices) can be found through the linear observability gramian in the following manner (Ersdal, 2010).

$$E_o = \boldsymbol{x}^T(0)\,\boldsymbol{W}_{o,linear}\boldsymbol{x}(0) \tag{5.3}$$

One can therefore say that the same expression based on the empirical observability gramian

$$E_o = \boldsymbol{x}^T(0)\,\boldsymbol{W}_o\boldsymbol{x}(0) \tag{5.4}$$

is an approximation of the output energy of a nonlinear system, which will hold locally (Singh and Hahn, 2006). Based on this, Singh and Hahn (2006) state that the singular values of $\boldsymbol{W}_o$ represents the part of the output energy along the direction of the corresponding singular vector, and that the sum of the singular vectors represents the total output energy of the system.

Two properties of linear observability gramians (based on real-valued system matrices) is that it is a symmetric and non-negative matrix (Ersdal, 2010). It holds for all symmetric and non-negative matrices that the eigenvalues λ_i are equal to its singular values ς_i

$$\varsigma_i = \lambda_j \quad \text{with} \quad i, j = 1, \cdots, n, \tag{5.5}$$

and that the left ($\boldsymbol{U}$) and right ($\boldsymbol{V}$) singular vectors of the singular value decomposition

$$\boldsymbol{W}_{o,linear} = \boldsymbol{USV}^T \tag{5.6}$$

are equal, but not necessarily unique. $\boldsymbol{S}$ is a diagonal matrix containing the singular values of $\boldsymbol{W}_o$.

This leads to the conclusion that also the eigenvectors of $\boldsymbol{W}_{o,linear}$ are equal to (one set of) its singular vectors:

$$\nu_i = \gamma_j \quad \text{with} \quad i, j = 1, \cdots, n, \tag{5.7}$$

where $\boldsymbol{\nu}_i$ are eigenvectors and $\boldsymbol{\gamma}_j$ singular vectors. Note that in both (5.7) and (5.5), i does not necessarily equals j.

Hence it is summarized that for $\boldsymbol{W}_{o,linear}$, the eigenvalues equals the singular values and the eigenvectors equals (one set of) singular vectors. Since all of this applies for the linear observability gramian, it also applies locally for the empirical observability gramian.

5.3.2 A new MDO

An idea for how to decided the amount of information a certain measurement matrix provides about each state will now be presented; the author of this thesis has not been able to find any literature suggesting that this idea has been tested previously.

Since the eigenvalues of $\boldsymbol{W}_o$ states the amount of output energy along the corresponding eigenvector, one can decompose each eigenvector into unit-state vectors, and multiply them with the corresponding eigenvalue. All of the eigenvalue/eigenvector-pairs are then summarized for each state. This will provide a measure of how much of the output energy that relates to each state, and hence how much information the output contains about each state.

Figure 5.1 shows how the eigenvector is decomposed for a system with three states. The amount of information in $\boldsymbol{\nu_i}$-direction is, as stated earlier, λ_i. A measure for the amount of information in $\boldsymbol{x}_1$-direction is $\lambda_i \cdot \boldsymbol{\nu}_i(1)\, \boldsymbol{x}_1$, in $\boldsymbol{x}_2$-direction $\lambda_i \cdot |\boldsymbol{\nu}_i(2)|\, \boldsymbol{x}_2$, and in $\boldsymbol{x}_3$-direction $\lambda_i \cdot \boldsymbol{\nu}_i(3)\, \boldsymbol{x}_3$, where $\{\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3\}$ are unit-state vectors.

Defining the amount of information about state $\boldsymbol{x}_i$ in the output signal as μ_i, the expression for it is thus given as

$$\mu_i = \sum_{j=1}^{n} \lambda_j \cdot |\boldsymbol{\nu}_j(i)| \tag{5.8}$$

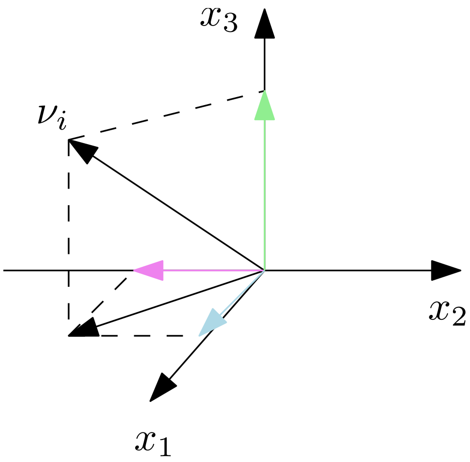


Figure 5.1: Decomposed eigenvector for a system with 3 states.

where n is the number of states, and λ_j and $\boldsymbol{\nu}_j$ are the eigenvalues and eigenvectors of $\boldsymbol{W}_o$, respectively. Since it does not matter if the information is in positive or negative direction, the absolute value of $\boldsymbol{\nu}_j\,(i)$ is applied.

Example 5.1 New MDO

Given a random observability gramian

$$\boldsymbol{W}_o = \begin{bmatrix} 10 & 7 & 3 \\ 7 & 15 & 5 \\ 3 & 5 & 6 \end{bmatrix} \tag{5.9}$$

Its eigenvalues, λ_i, and eigenvectors, ν_i, were found using the MATLAB function `[L,D] = eig(W)`*, and they are*

$$\lambda_1 = 3.7672 \textit{ and } \nu_1 = \begin{bmatrix} -0.0557 & 0.8433 & 0.5345 \end{bmatrix}^T \tag{5.10}$$

$$\lambda_2 = 5.1904 \textit{ and } \nu_2 = \begin{bmatrix} 0.4348 & -0.4614 & 0.7733 \end{bmatrix}^T \tag{5.11}$$

$$\lambda_3 = 22.0428 \textit{ and } \nu_3 = \begin{bmatrix} -0.8988 & -0.2754 & 0.3410 \end{bmatrix}^T \tag{5.12}$$

The measure of the amount of information about x_1 in the output energy is thus

$$\begin{aligned} \mu_1 &= \lambda_1 \cdot |\nu_1(1)| + \lambda_2 \cdot |\nu_2(1)| + \lambda_3 \cdot |\nu_3(1)| \\ &\approx 22.28 \end{aligned} \tag{5.13}$$

and equal for μ_2 and μ_3

$$\begin{aligned}\mu_2 &= \lambda_1 \cdot |\nu_1(2)| + \lambda_2 \cdot |\nu_2(2)| + \lambda_3 \cdot |\nu_3(2)| \\ &\approx 11.64 \qquad (5.14)\\ \mu_1 &= \lambda_1 \cdot |\nu_1(3)| + \lambda_2 \cdot |\nu_2(3)| + \lambda_3 \cdot |\nu_3(3)| \\ &\approx 15.60 \qquad (5.15)\end{aligned}$$

The measurement configuration that this observability gramian is based on thus provides most information about x_1.

5.4 Combining the covariance matrix and observability gramian

The covariance matrix of the EnKF thus provides information about which of the states that are poorly estimated, and the new MDO presented in the previous section provides knowledge of how much information the output contains about each state. If this is combined, one can choose to use the measurement matrix which provides most information about the states whose estimate needs improving.

Such a combination can be formulated as an optimization problem where one wants to maximize the process information $\boldsymbol{\mu}$, but where the diagonal terms of the covariance matrix $\boldsymbol{P}_{ii}$ are used to weight μ_i. Hence, information about a state whose estimate needs improving counts for more than information about a state whose estimate is fine. In addition to this, a scaling $\boldsymbol{S}$ is added. This can include scaling of $\boldsymbol{P}$ and $\boldsymbol{\mu}$ relative each other, as well as scaling that deals with which states that are more important to estimate well, such as states located upstream.

The following optimization problem is hence proposed as an on-line method for selecting the measurement matrix best suited for estimate updating at time k:

$$\max_{\boldsymbol{C}_j} \sum_{i=1}^{n} \boldsymbol{S}_{k,ii} \boldsymbol{P}_{k,ii} \cdot \mu_i \left(\boldsymbol{C}_j\right) \qquad (5.16)$$

where $\boldsymbol{C}_j$, $j = 1 \cdots m$, are the measurement matrix candidates and $\boldsymbol{P}_{k,ii}$ diagonal element i of the covariance matrix at time k. $\boldsymbol{\mu}$ is given by $\boldsymbol{W}_o$, which again is given by $\boldsymbol{C}$, which lead to $\boldsymbol{\mu}(\boldsymbol{C})$. This optimization problem can also be extended to include limitations/minimization of economic costs, flight distance, UAV properties, et cetera, but this is not further discussed in this thesis.

Example 5.2 New selection method

Given a random system with four system states, and two measurement-matrix candidates; $\boldsymbol{C}_a$ *and* $\boldsymbol{C}_b$. *This system is estimated by a Kalman filter which uses the method presented above to choose which of these measurement matrices to apply on-line.*

Assume that at a given time k, *the Kalman filter covariance matrix is as follows*

$$\boldsymbol{P}_k = \begin{bmatrix} 3 & 0 & 0 & 0 \\ 0 & 7 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 4 \end{bmatrix} \tag{5.17}$$

Hence, it is the estimate error for x_2 *and* x_4 *which is expected to be largest. Also assume that the system is scaled in advance, and that the "importance" scaling is given as*

$$\boldsymbol{S}_k = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 10 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 4 \end{bmatrix} \tag{5.18}$$

implying that it is most important to estimate x_2 *correctly followed by* x_4, x_1 *and finally* x_3, *whose estimate does not matter.*

Again, assume that $\boldsymbol{\mu}(\boldsymbol{C}_a)$ *and* $\boldsymbol{\mu}(\boldsymbol{C}_b)$ *are as follows*

$$\begin{aligned} \boldsymbol{\mu}(\boldsymbol{C}_a) &= \begin{bmatrix} \mu_1(\boldsymbol{C}_a) & \mu_2(\boldsymbol{C}_a) & \mu_3(\boldsymbol{C}_a) & \mu_4(\boldsymbol{C}_a) \end{bmatrix}^T \\ &= \begin{bmatrix} 0.1667 & 0.0333 & 2.1254 & 0.8722 \end{bmatrix}^T \qquad (5.19) \\ \boldsymbol{\mu}(\boldsymbol{C}_b) &= \begin{bmatrix} \mu_1(\boldsymbol{C}_b) & \mu_2(\boldsymbol{C}_b) & \mu_3(\boldsymbol{C}_b) & \mu_4(\boldsymbol{C}_b) \end{bmatrix}^T \\ &= \begin{bmatrix} 0.0625 & 0.1667 & 0.1250 & 0.0333 \end{bmatrix}^T \qquad (5.20) \end{aligned}$$

The optimization problem would then develop in the following manner

$$\max_{\boldsymbol{C}_j} \sum_{i=1}^{n} \boldsymbol{S}_{k,ii} \boldsymbol{P}_{k,ii} \cdot \mu_i \left(\boldsymbol{C}_j \right) \tag{5.21}$$

$$= \max \{16.79, 12.39\} \tag{5.22}$$

and based on this measurement-selection method, $\boldsymbol{C}_a$ should be applied at time step k.

The system described in Example 5.2 is not a real system. The calculation are based on fabricated observability gramians and covariance matrices, and it's sole purpose is to illustrate the optimization problem presented here. It is not an attempt to verify the method.

5.4.1 Calculating $\mathbf{W}_o$

In order to execute the optimization problem (5.16) at each time step, one must calculate the empirical observability gramian for each measurement matrix candidate at each time step.

It is seen from Section 2.1.1 that when calculating the EOG, $r \cdot s$ different output-trajectories are needed, and the system must therefore be simulated $r \cdot s$ times. Singh and Hahn (2005) state that $\boldsymbol{T}$ often is chosen equal to $\{I, -I\}$, and the number of simulations needed to find each $\boldsymbol{W}_o$ is thus $2s$. With m measurement-matrix candidates, the total number of simulations needed is $2sm$. Since the system in this thesis is a system with very high time constants, it must be simulated for quite some time before it reaches steady state. In addition to this it is a complex system which takes a while to simulate, and the entire process of calculating all the EOGs can therefore be very time consuming, most likely too time consuming to function in practice.

But as commented in Section 2.1.1, the EOG is applicable for nominal values of the input around the steady state $\boldsymbol{x}_{ss}$. This means that the EOGs does not need to be calculated again until the steady state has changed more than some pre-defined limit, which possibly takes quite some time since the steady state of the system varies slowly with the air temperature and boundary conditions. The initial EOGs can also be calculated off-line.

5.4.2 Tuning the EnKF

The question of tuning the EnKF after altering the $\boldsymbol{C}$ matrix also becomes apparent. The optimal tuning of the EnKF is dependent on the measurements, but tuning the EnKF each time a new measurement matrix is chosen, is not an option. This would lead to time gaps between the time the measurement matrix is chosen as the time it is actually implemented into the state-estimator/DP-loop. This is clearly not optimal since the chosen $\boldsymbol{C}$ is the best measurement matrix at the current time, not some time steps into the future.

But if the measurement matrices are defined in advance, such as they were in in this thesis, the EnKF could be tuned off-line, forming $\boldsymbol{C}$-matrix/tuning-pairs. The EnKF tuning would then be given directly from the chosen measurement matrix.

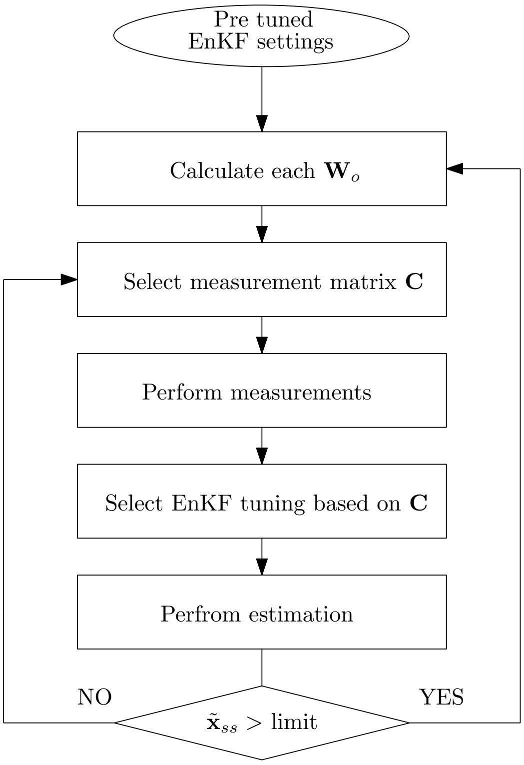


Figure 5.2: Flowchart for estimation loop.

Figure 5.2 shows the flowchart for the estimation and $\boldsymbol{C}$-selection loop. In the figure $\tilde{\boldsymbol{x}}_{ss}$ is defined as the absolute value of the difference between the current steady state $\boldsymbol{x}_{ss}$ and the steady state that the current EOG is based on $\bar{\boldsymbol{x}}_{ss}$: $\tilde{\boldsymbol{x}}_{ss} = |\boldsymbol{x}_{ss} - \bar{\boldsymbol{x}}_{ss}|$.

5.5 Chapter summary

In this chapter, suggestions for a new MDO based on the observability gramian, and a new on-line measurement-selection method combining this new MDO and the covariance matrix of a Kalman filter is presented.

The new MDO is a suggestion for how one can extract information about the amount of knowledge a given measurement matrix will provide about each individual system state, and not the entire state vector as a whole. In the selection method, this is then combined with information about which states that are poorly estimated, provided by the Kalman filter covariance matrix. Based on this the measurement matrix which provides more knowledge about those states that are poorly estimated is chosen. A scaling is also included, so that it is possible to identify some states as more important than others.

Ideally, this thesis would include the implementation and simulation of the optimization problem and estimation loop. But unfortunately there was no time for this.

Chapter 6

Discussion

In this chapter some concluding remarks on the main topics of the thesis are given, and the choices made throughout the report are deliberated. Some weaknesses are identified, and the results of the thesis are discussed.

All software developed for this thesis is available upon request.

6.1 The model

In Chapter 3, a general overview of the modeling of sea ice was given, and the dynamic thermodynamic model of Hibler III (1979) was presented as the model chosen for implementation in this thesis. Through simulations lasting one year, the Hibler-model proved to respond in an intuitive manner to the annually varying air temperature, by reaching a steady state through winter and by displaying ice melting during summer.

Hibler III (1979) uses this model with a cell size of 125 km, which is quite large. This cell size was also applied in this thesis. For some ice management schemes, this may not provide a sufficient level of detail, such as when more detailed information concerning the ice conditions near the ship is needed. But in order for the continuum approximation to be valid, the cell size must be much larger than the individual ice-floe sizes within the cells (Leppäranta, 2005), and when using the Hibler model to model ice-environments with large ice floes, cell sizes of this magnitude are necessary. The model described in this chapter is therefore not applicable for small scales with a high level of detail, in such cases a discrete model must be applied.

The implemented grid is of size (4 × 5), which is a fairly small grid. This leads to a state vector which is highly influenced by the boundary conditions, but when it comes to the simulation of the sea ice, this seems to work fine, as seen be the simulations of Section 3.5.

One important aspect of the Hibler-model, which should be noted, is that the expression for $\frac{D\boldsymbol{U}}{Dt}$ from the momentum balance becomes invalid when the mean ice thickness h approaches zero. Since the ice mass per unit area m is given as

$$m = \rho_{ice} h \tag{6.1}$$

and

$$\frac{D\boldsymbol{U}}{Dt} = \frac{1}{m}(\cdots) \tag{6.2}$$

$\frac{D\boldsymbol{U}}{Dt} \to \infty$ when $h \to 0$, which clearly leads to numerical problems as well as model instability. In this thesis, h never approaches zero, and this problem is therefore avoided here. But when simulating this model in conditions leading to mean ice thicknesses close to zero, precautions must be taken.

6.2 The state estimator

In Chapter 4 the EnKF was presented as the state estimator which would be applied for estimating the states of the Hibler-model. This state estimator was chosen because of the large number of states that is related to the Hibler-model, and because of the model's complexity. Using the EKF would be too time consuming to function properly in an ice management/DP-loop, such as the one depicted in Figure 2.3, and the less computationally demanding EnKF was chosen instead.

One aspect of the EnKF which should be mentioned is that it, unlike the EKF, uses particular $\bar{\boldsymbol{w}}$ and $\bar{\boldsymbol{v}}$ realizations, and not just their covariance matrices. These noise realizations are too extensive to tune directly, and they are therefore generated from their receptive variances, which function the actual tuning parameters. This should not affect the result much, but it does mean that there are some aspect of the tuning of the EnKF which cannot be controlled completely.

A table of possible measurement matrices was also listed, and some of the choices made when creating this list drastically simplified the UAV-based measurement scheme. It was assumed that the UAV should only measure one cell before returning to base, and the flight time should be ignored. These are assumptions which clearly does not hold in a real setting, as the question of where the UAV is located at the current time, and how long it takes for it to reach the different grid cells are important aspects of determining which measurement matrix to apply at the current time. But in this thesis, these aspects are ignored, mainly because it deals with the foundation for a more complete measurement-selection algorithm.

The EnKF was also tested for two of these measurement-matrix candidates. One measured u, v, h and A in a close cell situated upstream from the base cell (EnKF$_3$), and the other measured h in a distant cell situated downstream from the base cell (EnKF$_{23}$). Both of the EnKFs produced estimates that kept somewhat close to the actual states, but there were also differences between the two. EnKF$_3$ produced the overall best estimate, and often outperformed EnKF$_{23}$ at estimating u and v. At estimating A they were, however, often equally good. They both had more trouble estimating h, even though EnKF$_{23}$ did very well in some cells. The reason for this might be found by investigating Figure 4.6 (and the rest of the plots in Appendix D). Here it is clear that the process noise added to represent unmodeled aspects of the process, caused h to deviate more from the model which the EnKF is based on, than the other cell properties. Hence, the model applied in the EnKF to calculate the state's progression, is less accurate for the h-states than the others, leading to greater difficulties in estimating them.

That EnKF$_3$ had the overall best estimate is not unexpected. Both since EnKF$_3$ receives direct knowledge of representatives of all four cell properties, but also because the measurements that EnKF$_3$ is based on are made at an edge of the grid where the mean ice-drift enters. This means that many future states are highly influenced by these measurements. EnKF$_{23}$, on the other hand, receives direct information only about h in the corner of the grid where the mean ice-drift leaves. Next stop for the ice in this cell is outside the grid, without traveling through any other cells on its way. Hence, the measured h does not affect the properties of other cells as much as the measurements that EnKF$_3$ is based on.

Despite that there are differences between the two estimates, it might seem odd that they after all are so alike, and that EnKF$_{23}$ manages to estimate every state relatively well based solely on measuring h in cell 16. The reason for this is that the boundary conditions are considered to be known, hence they are given to the EnKFs as known input. In addition, this is a relatively small grid, and the boundary conditions affects the states quite a lot. Thus, no matter the measurement matrix, every EnKF has access to the boundary-condition information, which provides a good input on how the states of the system behaves. In a larger grid, or a model where the boundary conditions are included as unknown, one would probably see a greater difference between the two estimates.

As expected, both EnKFs estimated the measured values very well, except EnKF_3's h in cell 3. In addition to this, they both produced very good estimates of different other states. Even though some of these other well estimated states showed to be located somewhat near the measured cell, it was not the case for all of them. That EnKF_{23} would estimate u of cell 20 better than EnKF_3, would for example be difficult to predict based simply on the fact that it measures h in cell 16. It is thus not always intuitive which states that are better estimated with one EnKF than another.

6.3 The measurement-selection method

In Chapter 5 a new on-line measurement-selection method was proposed. This method is based on a new observability-gramian based MDO (also presented in this chapter), and the Kalman filter covariance matrix. It also contains a scaling factor which can include both scaling of the covariance matrix and observability gramian relative to each other, as well as scaling that represents the importance of having good estimates of the different states of the system.

One possible weakness of this method is that it does not account for the connection between the system states. Systems states are often closely related, and whether or not one is able to estimate a certain state properly often depends on how well these closely-related states are estimated. Relations such as these are not included in this on-line method, and even though the estimate-error covariance of such closely related states often are strongly connected themselves, meaning that if one estimate-error covariance is high so is the other and vice versa, one can always preponderate this through the scaling.

Unfortunately, there was no time to implement the method and test it in loop with the Hibler-model and ensemble Kalman filter. It is therefore important to emphasize that this on-line selection method, including the new MDO, is merely a theoretical suggestion. It has not been tested on any system, and the author can therefore not provide any results which neither supports that this could work nor concludes that it can not.

Chapter 7

Conclusion and Further Work

The work done in this report is motivated by the need for good ice management when carrying out DP operations in ice-infested environments, such as the Arctic. It was emphasized in Chapter 2 that knowledge of the surrounding ice environment is crucial for executing proper ice management and hence a successful DP operation. In order to gain this knowledge a good state estimator which can estimate the states of a sea-ice model is needed. The topic of this thesis has been to implement such a sea-ice model and state-estimator. It also includes a discussion on how to select the optimal measurement inputs to this state estimator on-line.

It has been shown in this report that the dynamic thermodynamic sea-ice model of Hibler III (1979) is a model that, despite some assumptions and the fact that the development of the model states are strongly affected by the boundary conditions, is a model well suited to simulate large scale sea-ice response to varying air temperature. However, since it is a continuum model, it is not suited to simulate small scale scenarios with a high level of detail. When applying the Hibler-model to areas with little ice, care must be taken since the model does not handle scenarios where the mean thickness h of a cell approaches zero.

Two ensemble Kalman filters (EnKF) were implemented as state estimators, based on two different measurement matrices. They both estimate the system states quite well, partially because of the state's dependency on the boundary conditions, which are assumed to be known. As expected, it was shown that the EnKF based on four measurements at the entrance edge of the mean ice drift performed better overall than the EnKF based on one measurement at the exit edge of the mean ice drift. But there were some states where the latter performed better, which shows that when trying to estimate certain states of a system better than others, it is not always the overall best estimator that is best suited for the task.

A new on-line measurement selection method has also been presented, in the form of a simple optimization problem. This method was only suggested theoretically, and there was no time for implementation and testing. The author is therefore aware that there might be aspects of the optimization problem which makes it unsuited for the presented task of on-line measurement-selection, but this can not be elaborated further until such implementation and testing has been performed.

7.1 Further work

The results presented in this thesis are mainly a starting point for further work regarding the problem presented in Chapter 2: estimating ice-model properties for proper execution of ice management. Suggestions for further work include

- Further tuning and optimization of the ensemble Kalman filters.
- Implementing ensemble Kalman filters for the remaining measurement-matrix candidates.
- Implementing and testing the on-line measurement selection method.
- Further extend the optimization problem of the on-line measurement selection method to include for example economical aspects.

Bibliography

Ames, W. F. (1992). *Numerical Methods for Partial Differential Equations*, 3rd edn, Academic Press, INC.

Bonnemaire, B., Jensen, A., Gudmestad, O., Lundamo, T. and Løset, S. (2007). Oil and gas developments in arctic and cold regions, challenges related to station-keeping in ice, *9th annual ITSOK Conference*.

Chen, C.-T. (1999). *Linear System Theory and Design*, 3rd edn, Oxford University Press.

Delhez, E. J. and Deleersnijder, E. (2006). Overshootings and spurious oscillations caused by biharmonic mixing, *Ocean Modelling* **17**(3): 183–198.

Eik, K. J. (2010). *Ice Management in Arctic Offshore Operations and Field Developments*, PhD thesis, NTNU - Norwegian University of Science and Technology.

Ersdal, A. M. (2010). Sensor selection based on observability gramians, Process report, NTNU - Norwegian University of Science and Technology.

Evensen, G. (1994). Sequential data assimilation with a nonlinear quasi-geostrophic model using monte carlo methods to forecast error statistics, *Journal of Geophysical Research* **99**(C5): 10143–10162.

Evensen, G. (2003). The ensemble kalman filter: theoretical formulation and practical implementation, *Ocean Dynamics* **53**(4): 343–367.

Evensen, G. (2009). *Data Assimilation*, 2nd edn, Springer.

Fossen, T. I. (2002). *Marine Control Systems*, Marine Cybernetics.

Gautier, D., Bird, K., Charpentier, R., Grantz, D., Klett, T., Moore, T., Pitman, J., Schenk, C., Schuenemeyer, J., Sørensen, K., Tennyson, M., Valin, Z. and Wandrey, C. (2009). Assessment of undiscovered oil and gas in the arctic, *Science* **324**(5931): 1175–1179.

Haugen, J., Imsland, L., Løset, S. and Skjetne, R. (2011). Ice observer system for ice management operations, *Proc. 21st Int. Offshore (Ocean) and Polar Eng. Conf., June 19-24, 2011, Maui, Hawaii, USA*.

Heinonen, J. (2004). *Constitutive Modeling of Ice Rubble in First-Year Ridge Keel*, PhD thesis, VTT - Technical Research Center of Finland.

Hibler III, W. (1979). A dynamic thermodynamic sea ice model, *Journal of Physical Oceanography* **9**(4): 815–846.

Holton, J. R. (2004). *An Introduction to Dynamic Meteorology*, 2nd edn, Elsevier Academic Press.

Hunke, E. and Dukowicz, J. (1997). An elastic-viscous-plastic model for sea ice dynamics, *Journal of Physical Oceanography* **27**(9): 1849–1867.

Jenssen, N., Muddesitti, S., Phillips, D. and Backstrom, K. (2009). DP in ice conditions, *Dynamic Positioning Conference*.

Julian Day (2011). Accessed 2011.05.30.
URL: *http://dictionary.sensagent.com/julian+day/en-en/*

Keinonen, A., Wells, H., Dunderdale, P., Pilkington, R., Miller, G. and Brovin, A. (2000). Dynamic positioning operation in ice, offshore sakhalin, may - june 1999, *Proceedings of the Tenth (2000) International Offshore and Polar Engineering Conference*.

Leppäranta, M. (1993). A review of analytical models of sea-ice growth, *Atmosphere-Ocean* **31**(1): 123–138.

Leppäranta, M. (2005). *The Drift of Sea Ice*, Springer.

Løset, S., Shkhinek, K., Gudmestad, O. and Høyland, K. (2006). *Actions From Ice on Arctic Offshore and Coastal Structures*, LAN.

Meteorological conditions (2011). Accessed 2011.04.04.
URL: *http://www.barentsportal.com/*

Moran, K., Backman, J. and Farrell, J. (2006). Deepwater drilling in the arctic ocean's permanent sea ice, *Proceedings of the Integrated Ocean Drilling Program*, Vol. 302.

Nocedal, J. and Wright, S. J. (2006). *Numerocal Optimization*, 2nd edn, Springer.

Reichle, R. H., Walker, J. P., Koster, R. D. and Houser, P. R. (2002). Extended versus ensemble kalman filtering for land data assimilation, *Journal of Hydrometerology* **3**(6): 728–740.

Riley, W., Sturges, L. and Morris, D. (2007). *Mechanics of Materials*, 6th edn, John Wiley and sons, inc.

Samuelson, R. and Vallis, G. K. (1997). Large-scale circulation with small diapycnal diffusion: The two-termocline limit, *Journal of Marine Research* **55**(2): 223–275.

Simon, D. (2006). *Optimal State Estimation*, John Wiley & Sons, Inc.

Singh, A. K. and Hahn, J. (2005). Determining optimal sensor locations for state and parameter estimation for stable nonlinear systems, *Industrial and Engineering Chemistry Research* **44**(15): 5645–5659.

Singh, A. K. and Hahn, J. (2006). Sensor location for stable nonlinear dynamic systems: Multiple sensor case, *Industrial and Engineering Chemistry Research* **45**(10): 3615–3623.

Süli, E. and Mayers, D. (2004). *An Introduction to Numerical Analysis*, 4th edn, Cambridge University Press.

Sørensen, A., Fossen, T., Strand, J. and Ronæss, M. (2001). Position control systems for offshore vessels, *in* F. El-Hawary (ed.), *The Ocean Engineering Handbook*, CRC Press, chapter 3.

Strikwerda, J. C. (2003). *Finite Difference Schemes and Partial Differential Equations*, SIAM.

Wang, L. and Ikeda, M. (2004). A lagrangian description of sea ice dynamics using the finite element method, *Ocean Modeling* **7**(1-2): 21–38.

Appendix A

Summary of model equations

The momentum equation

$$m\frac{D\boldsymbol{U}}{Dt} = -mf\boldsymbol{k}\times\boldsymbol{U} + \boldsymbol{\tau}_a + \boldsymbol{\tau}_w - mg\nabla H + \boldsymbol{F} \tag{A.1}$$

$$\frac{D\boldsymbol{U}}{Dt} = \frac{\partial \boldsymbol{U}}{\partial t} + \boldsymbol{U}\nabla\boldsymbol{U} \tag{A.2}$$

with air and water drag forces,

$$\boldsymbol{\tau}_a = \rho_a C_a|\boldsymbol{U}_g|\left(\boldsymbol{U}_g\cos(\phi) + \boldsymbol{k}\times\boldsymbol{U}_g\sin(\phi)\right) \tag{A.3}$$

$$\boldsymbol{\tau}_w = \rho_w C_w|\boldsymbol{U}_w - \boldsymbol{U}|\left((\boldsymbol{U}_w - \boldsymbol{U})\cos(\theta) + \boldsymbol{k}\times(\boldsymbol{U}_w - \boldsymbol{U})\sin(\theta)\right) \tag{A.4}$$

ice strength

$$P = P^\star h e^{-C(1-A)} \tag{A.5}$$

and internal forces.

$$\Delta = \sqrt{\left(\dot{\epsilon}_{xx}^2 + \dot{\epsilon}_{yy}^2\right)\left(1 + \frac{1}{e^2}\right) + \frac{4}{e^2}\dot{\epsilon}_{yx}^2 + 2\dot{\epsilon}_x^2\dot{\epsilon}_{yy}^2\left(1 - \frac{1}{e^2}\right)} \tag{A.6}$$

$$\zeta = \frac{P}{2\Delta}, \eta = \frac{\zeta}{e^2} \tag{A.7}$$

$$F_x = \frac{\partial}{\partial x}\left[(\eta+\zeta)\frac{\partial u}{\partial x} + (\zeta-\eta)\frac{\partial v}{\partial y} - \frac{1}{2}P\right] + \frac{\partial}{\partial y}\left[\eta\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right)\right] \tag{A.8}$$

$$F_y = \frac{\partial}{\partial y}\left[(\eta+\zeta)\frac{\partial v}{\partial y} + (\zeta-\eta)\frac{\partial u}{\partial x} - \frac{1}{2}P\right] + \frac{\partial}{\partial x}\left[\eta\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right)\right] \tag{A.9}$$

The ice mean thickness and compactness

$$\frac{\partial h}{\partial t} = -\frac{\partial(uh)}{\partial x} - \frac{\partial(vh)}{\partial y} + S_h + \xi_h \tag{A.10}$$

$$\frac{\partial A}{\partial t} = -\frac{\partial(uA)}{\partial x} - \frac{\partial(vA)}{\partial y} + S_A + \xi_A \tag{A.11}$$

with thermodynamic terms

$$S_h = g\,(h/A)\,A + (1 - A)\,g(0) \tag{A.12}$$

$$S_A = \begin{cases} \frac{g(0)}{h}\,(1 - A) & \text{if} \quad g(0) > 0 \\ 0 & \text{if} \quad g(0) < 0 \end{cases} + \begin{cases} 0 & \text{if} \quad S_h > 0 \\ \frac{A}{2h} S_h & \text{if} \quad S_h < 0 \end{cases} \tag{A.13}$$

and growth function.

$$g(h, T_a) = \frac{1}{\rho_i L}\left[\frac{\kappa_i}{h + \kappa_i/k_a}\,(T_f - T_a) - Q_w\right] \tag{A.14}$$

Appendix B

Spatial finite differences

The spatial finite differences (SFD) used in this thesis, and the expressions for them will be presented here. The SFDs were calculated using central differencing (Ames, 1992):

$$\frac{dz}{dn} \approx \frac{1}{\Delta n}(z_{n+\frac{1}{2}} - z_{n-\frac{1}{2}}) \tag{B.1}$$

Note that since the grid is square, $\Delta x = \Delta y = d$. Sometimes, attributes can be required at other grid points than where they are calculated, e.g. $u_{i+\frac{1}{2},j}$ is part of (B.5), even though the value of u at this grid point does not exists. To solve this, one simply uses the following average as an approximation.

$$u_{i+\frac{1}{2},j} \approx \frac{1}{2}\left(u_{i+1,j} + u_{i,j}\right) \tag{B.2}$$

First the derivatives connected to the momentum balance through the material derivative will be presented. These are calculated at (i,j) because u, and hence the momentum balance is calculated here.

$$\begin{aligned}
\left(\frac{\partial u}{\partial x}\right)_{i,j} &\approx \frac{1}{d}\left(u_{i+\frac{1}{2},j} - u_{i-\frac{1}{2},j}\right) \\
&\approx \frac{1}{d}\left(\frac{1}{2}(u_{i+1,j} + u_{i,j}) - \frac{1}{2}(u_{i,j} + u_{i-1,j})\right) \\
&= \frac{1}{2d}\left(u_{i+1,j} - u_{i-1,j}\right) && \text{(B.3)} \\
\left(\frac{\partial u}{\partial y}\right)_{i,j} &= \frac{1}{2d}\left(u_{i,j+1} - u_{i,j-1}\right) && \text{(B.4)}
\end{aligned}$$

and similar for $\frac{\partial v}{\partial x}$ and $\frac{\partial v}{\partial y}$.

Secondly the derivatives connected to the constitutive law, and hence the internal forces F are presented. These are also calculated at (i, j) since F is used in the momentum balance.

$$
\begin{aligned}
\left(\frac{\partial}{\partial x}\left(\eta\frac{\partial u}{\partial x}\right)\right)_{i,j} &\approx \frac{\partial}{\partial x}\left(\eta\frac{1}{d}\left(u_{i+\frac{1}{2},j}-u_{i-\frac{1}{2},j}\right)\right)\\
&\approx \frac{1}{d^2}\left(\eta_{i+\frac{1}{2},j}\left(u_{i+1,j}-u_{i,j}\right)-\eta_{i-\frac{1}{2},j}\left(u_{i,j}-u_{i-1,j}\right)\right)\\
&\approx \frac{1}{2d^2}\Big(\left(\eta_{i+\frac{1}{2},j+\frac{1}{2}}+\eta_{i+\frac{1}{2},j-\frac{1}{2}}\right)\left(u_{i+1,j}-u_{i,j}\right)\\
&\quad-\left(\eta_{i-\frac{1}{2},j+\frac{1}{2}}+\eta_{i-\frac{1}{2},j-\frac{1}{2}}\right)\left(u_{i,j}-u_{i-1,j}\right)\Big) \qquad \text{(B.5)}\\
\left(\frac{\partial}{\partial y}\left(\eta\frac{\partial u}{\partial y}\right)\right)_{i,j} &\approx \frac{1}{2d^2}\Big(\left(\eta_{i+\frac{1}{2},j+\frac{1}{2}}+\eta_{i-\frac{1}{2},j+\frac{1}{2}}\right)\left(u_{i,j+1}-u_{i,j}\right)\\
&\quad-\left(\eta_{i+\frac{1}{2},j-\frac{1}{2}}+\eta_{i-\frac{1}{2},j-\frac{1}{2}}\right)\left(u_{i,j}-u_{i,j-1}\right)\Big) \qquad \text{(B.6)}\\
\left(\frac{\partial}{\partial x}\left(\eta\frac{\partial u}{\partial y}\right)\right)_{i,j} &\approx \frac{1}{d^2}\Big(\eta_{i+\frac{1}{2},j}\left(u_{i+\frac{1}{2},j+\frac{1}{2}}-u_{i+\frac{1}{2},j-\frac{1}{2}}\right)\\
&\quad-\eta_{i-\frac{1}{2},j}\left(u_{i-\frac{1}{2},j+\frac{1}{2}}-u_{i-\frac{1}{2},j-\frac{1}{2}}\right)\Big)\\
&\approx \frac{1}{d^2}\Big(\frac{1}{2}\Big(\eta_{i+\frac{1}{2},j+\frac{1}{2}}\left(u_{i+\frac{1}{2},j+1}-u_{i+\frac{1}{2},j}\right)\\
&\quad+\eta_{i+\frac{1}{2},j-\frac{1}{2}}\left(u_{i+\frac{1}{2},j}-u_{i+\frac{1}{2},j-1}\right)\Big)\\
&\quad-\frac{1}{2}\Big(\eta_{i-\frac{1}{2},j+\frac{1}{2}}\left(u_{i-\frac{1}{2},j+1}-u_{i-\frac{1}{2},j}\right)\\
&\quad+\eta_{i-\frac{1}{2},j-\frac{1}{2}}\left(u_{i-\frac{1}{2},j}-u_{i-\frac{1}{2},j-1}\right)\Big)\Big)\\
&\approx \frac{1}{2d^2}\Big(\eta_{i+\frac{1}{2},j+\frac{1}{2}}\left(\frac{1}{2}\left(u_{i+1,j+1}+u_{i,j+1}\right)-\frac{1}{2}\left(u_{i+1,j}+u_{i,j}\right)\right)\\
&\quad+\eta_{i+\frac{1}{2},j-\frac{1}{2}}\left(\frac{1}{2}\left(u_{i+1,j}+u_{i,j}\right)-\frac{1}{2}\left(u_{i+1,j-1}+u_{i,j-1}\right)\right)\\
&\quad-\eta_{i-\frac{1}{2},j+\frac{1}{2}}\left(\frac{1}{2}\left(u_{i,j+1}+u_{i-1,j+1}\right)-\frac{1}{2}\left(u_{i,j}+u_{i-1,j}\right)\right)\\
&\quad-\eta_{i-\frac{1}{2},j-\frac{1}{2}}\left(\frac{1}{2}\left(u_{i,j}+u_{i-1,j}\right)-\frac{1}{2}\left(u_{i,j-1}+u_{i-1,j-1}\right)\right)\Big)
\end{aligned}
$$

$$
\begin{aligned}
&= \frac{1}{4d^2}\Big(\eta_{i+\frac{1}{2},j+\frac{1}{2}}\left(u_{i+1,j+1}+u_{i,j+1}-u_{i+1,j}-u_{i,j}\right)\\
&\quad+\eta_{i+\frac{1}{2},j-\frac{1}{2}}\left(u_{i+1,j}+u_{i,j}-u_{i+1,j-1}-u_{i,j-1}\right)\\
&\quad-\eta_{i-\frac{1}{2},j+\frac{1}{2}}\left(u_{i,j+1}+u_{i-1,j+1}-u_{i,j}-u_{i-1,j}\right)\\
&\quad-\eta_{i-\frac{1}{2},j-\frac{1}{2}}\left(u_{i,j}+u_{i-1,j}-u_{i,j-1}-u_{i-1,j-1}\right)\Big) \qquad \text{(B.7)}
\end{aligned}
$$

$$
\begin{aligned}
\left(\frac{\partial}{\partial y}\left(\eta\frac{\partial u}{\partial x}\right)\right)_{i,j} &\approx \frac{1}{4d^2}\Big(\eta_{i+\frac{1}{2},j+\frac{1}{2}}\left(u_{i+1,j+1}+u_{i+1,j}-u_{i,j+1}-u_{i,j}\right)\\
&\quad+\eta_{i-\frac{1}{2},j+\frac{1}{2}}\left(u_{i,j+1}+u_{i,j}-u_{i-1,j+1}-u_{i-1,j}\right)\\
&\quad-\eta_{i+\frac{1}{2},j-\frac{1}{2}}\left(u_{i+1,j}+u_{i+1,j-1}-u_{i,j}-u_{i,j-1}\right)\\
&\quad-\eta_{i-\frac{1}{2},j-\frac{1}{2}}\left(u_{i,j}+u_{i,j-1}-u_{i-1,j}-u_{i-1,j-1}\right)\Big) \qquad \text{(B.8)}
\end{aligned}
$$

$$
\begin{aligned}
\left(\frac{\partial P}{\partial x}\right)_{i,j} &\approx \frac{1}{d}\left(P_{i+\frac{1}{2},j}-P_{i-\frac{1}{2},j}\right)\\
&\approx \frac{1}{2d}\left(P_{i+\frac{1}{2},j+\frac{1}{2}}+P_{i+\frac{1}{2},j-\frac{1}{2}}-P_{i-\frac{1}{2},j+\frac{1}{2}}-P_{i-\frac{1}{2},j-\frac{1}{2}}\right)\\
\left(\frac{\partial P}{\partial y}\right)_{i,j} &\approx \frac{1}{2d}\left(P_{i+\frac{1}{2},j+\frac{1}{2}}+P_{i-\frac{1}{2},j+\frac{1}{2}}-P_{i+\frac{1}{2},j-\frac{1}{2}}-P_{i-\frac{1}{2},j-\frac{1}{2}}\right) \qquad \text{(B.9)}
\end{aligned}
$$

and similar for $\frac{\partial}{\partial x}\left(\zeta\frac{\partial u}{\partial x}\right)$, $\frac{\partial}{\partial y}\left(\zeta\frac{\partial u}{\partial y}\right)$, $\frac{\partial}{\partial x}\left(\zeta\frac{\partial u}{\partial y}\right)$, $\frac{\partial}{\partial y}\left(\zeta\frac{\partial u}{\partial x}\right)$, $\frac{\partial}{\partial x}\left(\eta\frac{\partial v}{\partial x}\right)$, $\frac{\partial}{\partial y}\left(\eta\frac{\partial v}{\partial y}\right)$, $\frac{\partial}{\partial x}\left(\eta\frac{\partial v}{\partial y}\right)$, $\frac{\partial}{\partial y}\left(\eta\frac{\partial v}{\partial x}\right)$, $\frac{\partial}{\partial x}\left(\zeta\frac{\partial v}{\partial x}\right)$, $\frac{\partial}{\partial y}\left(\zeta\frac{\partial v}{\partial y}\right)$, $\frac{\partial}{\partial x}\left(\zeta\frac{\partial v}{\partial y}\right)$ and $\frac{\partial}{\partial y}\left(\zeta\frac{\partial v}{\partial x}\right)$.

The derivatives connected to the ice-thickness distribution are as follows. They are calculated at $(i+\frac{1}{2}, j+\frac{1}{2})$ since h and A are calculated at these points.

$$
\begin{aligned}
\left(\frac{\partial(uh)}{\partial x}\right)_{i+\frac{1}{2},j+\frac{1}{2}} &\approx \frac{1}{d}\left(u_{i+1,j+\frac{1}{2}}\cdot h_{i+1,j+\frac{1}{2}}-u_{i,j+\frac{1}{2}}\cdot h_{i,j+\frac{1}{2}}\right)\\
&\approx \frac{1}{d}\Big(\frac{1}{2}\left(u_{i+1,j+1}+u_{i+1,j}\right)\cdot\frac{1}{2}\left(h_{i+\frac{3}{2},j+\frac{1}{2}}+h_{i+\frac{1}{2},j+\frac{1}{2}}\right)\\
&\quad-\frac{1}{2}\left(u_{i,j+1}+u_{i,j}\right)\cdot\frac{1}{2}\left(h_{i+\frac{1}{2},j+\frac{1}{2}}+h_{i-\frac{1}{2},j+\frac{1}{2}}\right)\Big)\\
&= \frac{1}{4d}\Big(\left(u_{i+1,j+1}+u_{i+1,j}\right)\left(h_{i+\frac{3}{2},j+\frac{1}{2}}+h_{i+\frac{1}{2},j+\frac{1}{2}}\right)\\
&\quad-\left(u_{i,j+1}+u_{i,j}\right)\left(h_{i+\frac{1}{2},j+\frac{1}{2}}+h_{i-\frac{1}{2},j+\frac{1}{2}}\right)\Big) \qquad \text{(B.10)}
\end{aligned}
$$

$$\begin{aligned}\left(\frac{\partial(vh)}{\partial y}\right)_{i+\frac{1}{2},j+\frac{1}{2}} &\approx \frac{1}{4d}\Big(\left(v_{i+1,j+1}+v_{i,j+1}\right)\left(h_{i+\frac{1}{2},j+\frac{3}{2}}+h_{i+\frac{1}{2},j+\frac{1}{2}}\right)\\ &\quad-\left(v_{i+1,j}+v_{i,j}\right)\left(h_{i+\frac{1}{2},j+\frac{1}{2}}+h_{i+\frac{1}{2},j-\frac{1}{2}}\right)\Big)\end{aligned} \tag{B.11}$$

$$\begin{aligned}\left(\frac{\partial^2 h}{\partial x^2}\right)_{i+\frac{1}{2},j+\frac{1}{2}} &\approx \frac{\partial}{\partial x}\cdot\frac{1}{d}\left(h_{i+1,j+\frac{1}{2}}-h_{i,j+\frac{1}{2}}\right)\\ &\approx \frac{1}{d^2}\left(h_{i+\frac{3}{2},j+\frac{1}{2}}-h_{i+\frac{1}{2},j+\frac{1}{2}}-h_{i+\frac{1}{2},j+\frac{1}{2}}+h_{i-\frac{1}{2},j+\frac{1}{2}}\right)\\ &=\frac{1}{d^2}\left(h_{i+\frac{3}{2},j+\frac{1}{2}}-2h_{i+\frac{1}{2},j+\frac{1}{2}}+h_{i-\frac{1}{2},j+\frac{1}{2}}\right)\end{aligned} \tag{B.12}$$

$$\left(\frac{\partial^2 h}{\partial y^2}\right)_{i+\frac{1}{2},j+\frac{1}{2}} \approx \frac{1}{d^2}\left(h_{i+\frac{1}{2},j+\frac{3}{2}}-2h_{i+\frac{1}{2},j+\frac{1}{2}}+h_{i+\frac{1}{2},j-\frac{1}{2}}\right) \tag{B.13}$$

$$\begin{aligned}\left(\frac{\partial^4 h}{\partial x^4}\right)_{i+\frac{1}{2},j+\frac{1}{2}} &\approx \frac{\partial}{\partial x}\cdot\frac{1}{d^3}\Big(\left(h_{i+\frac{3}{2},j}-2h_{i+\frac{1}{2},j}+h_{i-\frac{1}{2},j}\right)\\ &\quad-\left(h_{i+\frac{1}{2},j}-2h_{i-\frac{1}{2},j}+h_{i-\frac{3}{2},j}\right)\Big)\\ &\approx \frac{1}{d^4}\Big(\left(h_{i+2,j}-2h_{i+1,j}+h_{i,j}-h_{i+1,j}+2h_{i,j}-h_{i-1,j}\right)\\ &\quad-\left(h_{i+1,j}-2h_{i,j}+h_{i-1,j}-h_{i,j}+2h_{i-1,j}-h_{i-2,j}\right)\Big)\\ &=\frac{1}{d^4}\left(h_{i+2,j}-4h_{i+1,j}+6h_{i,j}-4h_{i-1,j}+h_{i-2,j}\right)\end{aligned} \tag{B.14}$$

$$\left(\frac{\partial^4 h}{\partial y^4}\right)_{i+\frac{1}{2},j+\frac{1}{2}} \approx \frac{1}{d^4}\left(h_{i,j+2}-4h_{i,j+1}+6h_{i,j}-4h_{i,j-1}+h_{i,j-2}\right) \tag{B.15}$$

$$\begin{aligned}\left(\frac{\partial^4 h}{\partial x^2 y^2}\right)_{i+\frac{1}{2},j+\frac{1}{2}} &\approx \frac{\partial}{\partial y}\cdot\frac{1}{d^3}\Big(\left(h_{i+\frac{3}{2},j+1}-2h_{i+\frac{1}{2},j+1}+h_{i-\frac{1}{2},j+1}\right)\\ &\quad-\left(h_{i+\frac{3}{2},j}-2h_{i+\frac{1}{2},j}+h_{i-\frac{1}{2},j}\right)\Big)\\ &\approx \frac{1}{d^4}\Big(\Big(h_{i+\frac{3}{2},j+\frac{3}{2}}-2h_{i+\frac{1}{2},j+\frac{3}{2}}+h_{i-\frac{1}{2},j+\frac{3}{2}}-h_{i+\frac{3}{2},j+\frac{1}{2}}\\ &\quad+2h_{i+\frac{1}{2},j+\frac{1}{2}}-h_{i-\frac{1}{2},j+\frac{1}{2}}\Big)-\Big(h_{i+\frac{3}{2},j+\frac{1}{2}}-2h_{i+\frac{1}{2},j+\frac{1}{2}}\\ &\quad+h_{i-\frac{1}{2},j+\frac{1}{2}}-h_{i+\frac{3}{2},j-\frac{1}{2}}+2h_{i+\frac{1}{2},j-\frac{1}{2}}-h_{i-\frac{1}{2},j-\frac{1}{2}}\Big)\Big)\end{aligned} \tag{B.16}$$

$$
\begin{aligned}
&= \frac{1}{d^4}\Big(h_{i+\frac{3}{2},j+\frac{3}{2}} - 2h_{i+\frac{1}{2},j+\frac{3}{2}} + h_{i-\frac{1}{2},j+\frac{3}{2}} \\
&\quad - 2h_{i+\frac{3}{2},j+\frac{1}{2}} + 4h_{i+\frac{1}{2},j+\frac{1}{2}} - 2h_{i-\frac{1}{2},j+\frac{1}{2}} \\
&\quad + h_{i+\frac{3}{2},j-\frac{1}{2}} - 2h_{i+\frac{1}{2},j-\frac{1}{2}} + h_{i-\frac{1}{2},j-\frac{1}{2}}\Big)
\end{aligned}
\tag{B.17}
$$

$$
\begin{aligned}
\left(\frac{\partial^4 h}{\partial y^2 x^2}\right)_{i+\frac{1}{2},j+\frac{1}{2}} &\approx \frac{1}{d^4}\Big(h_{i+\frac{3}{2},j+\frac{3}{2}} - 2h_{i+\frac{3}{2},j+\frac{1}{2}} + h_{i+\frac{3}{2},j-\frac{1}{2}} \\
&\quad - 2h_{i+\frac{1}{2},j+\frac{3}{2}} + 4h_{i+\frac{1}{2},j+\frac{1}{2}} - 2h_{i+\frac{1}{2},j-\frac{1}{2}} \\
&\quad + h_{i-\frac{1}{2},j+\frac{3}{2}} - 2h_{i-\frac{1}{2},j+\frac{1}{2}} + h_{i-\frac{1}{2},j-\frac{1}{2}}\Big)
\end{aligned}
\tag{B.18}
$$

and similar for A. It should be noted that (B.10) differ somewhat from the expression for $\frac{\partial(uh)}{\partial x}$ in Hibler III (1979). Hibler III (1979) claims that

$$
\begin{aligned}
\left(\frac{\partial(uh)}{\partial x}\right)_{i+\frac{1}{2},j+\frac{1}{2}} &= \frac{1}{4d}\Big(\left(u_{i+1,j+1} + u_{i+1,j}\right)\left(h_{i+\frac{3}{2},j+\frac{1}{2}} + h_{i+\frac{1}{2},j+\frac{1}{2}}\right) \\
&\quad - \left(u_{i-1,j+1} + u_{i-1,j}\right)\left(h_{i+\frac{1}{2},j+\frac{1}{2}} + h_{i-\frac{1}{2},j+\frac{1}{2}}\right)\Big)
\end{aligned}
\tag{B.19}
$$

but based on the calculations shown in (B.10) it is assumed that this probably is a misprint, and that (B.10) is the correct SFD for $\frac{\partial(uh)}{\partial x}$ and similar for A.

Appendix C

Initial ensemble

The initial ensemble for every EnKF was set to:

$$\hat{\boldsymbol{X}}_0 = \begin{bmatrix} \hat{\boldsymbol{x}}_0^1 & \hat{\boldsymbol{x}}_0^2 & \hat{\boldsymbol{x}}_0^3 & \hat{\boldsymbol{x}}_0^4 & \hat{\boldsymbol{x}}_0^5 & \hat{\boldsymbol{x}}_0^6 & \hat{\boldsymbol{x}}_0^7 & \hat{\boldsymbol{x}}_0^8 & \hat{\boldsymbol{x}}_0^9 & \hat{\boldsymbol{x}}_0^{10} \end{bmatrix}, \tag{C.1}$$

where

$$\begin{aligned}
\hat{\boldsymbol{x}}_0^1 = \Big[\; & 0.002 \quad 0.01 \quad 2.55 \quad 0.92 \quad 0.001 \quad 0.01 \quad 2.6 \quad 0.92 \\
& \cdots \quad 0.001 \quad 0.01 \quad 2.6 \quad 0.92 \quad 0.001 \quad 0.01 \quad 2.7 \quad 0.92 \\
& \cdots \quad 0.002 \quad 0.02 \quad 2.7 \quad 0.92 \quad 0.002 \quad 0.02 \quad 2.52 \quad 0.92 \\
& \cdots \quad 0.002 \quad 0.02 \quad 2.55 \quad 0.92 \quad 0.002 \quad 0.02 \quad 2.6 \quad 0.92 \\
& \cdots \quad 0.002 \quad 0.02 \quad 2.7 \quad 0.92 \quad 0.003 \quad 0.02 \quad 2.7 \quad 0.92 \\
& \cdots \quad 0.003 \quad 0.02 \quad 2.52 \quad 0.92 \quad 0.003 \quad 0.02 \quad 2.55 \quad 0.92 \\
& \cdots \quad 0.003 \quad 0.02 \quad 2.6 \quad 0.92 \quad 0.003 \quad 0.02 \quad 2.7 \quad 0.92 \\
& \cdots \quad 0.005 \quad 0.02 \quad 2.7 \quad 0.92 \quad 0.005 \quad 0.02 \quad 2.52 \quad 0.92 \\
& \cdots \quad 0.005 \quad 0.02 \quad 2.7 \quad 0.92 \quad 0.005 \quad 0.02 \quad 2.6 \quad 0.92 \\
& \cdots \quad 0.005 \quad 0.02 \quad 2.6 \quad 0.92 \quad 0.005 \quad 0.02 \quad 2.7 \quad 0.92 \quad \Big]^T
\end{aligned} \tag{C.2}$$

$$\begin{aligned}
\hat{\boldsymbol{x}}_0^2 = \Big[\; & -0.002 \quad -0.01 \quad 2.45 \quad 0.88 \quad -0.001 \quad -0.01 \quad 2.4 \quad 0.88 \\
& \cdots \quad -0.001 \quad -0.01 \quad 2.4 \quad 0.88 \quad -0.001 \quad -0.01 \quad 2.3 \quad 0.88 \\
& \cdots \quad -0.002 \quad -0.02 \quad 2.3 \quad 0.88 \quad -0.002 \quad -0.02 \quad 2.48 \quad 0.88 \\
& \cdots \quad -0.002 \quad -0.02 \quad 2.45 \quad 0.88 \quad -0.002 \quad -0.02 \quad 2.4 \quad 0.88 \\
& \cdots \quad -0.002 \quad -0.02 \quad 2.3 \quad 0.88 \quad -0.003 \quad -0.02 \quad 2.3 \quad 0.88 \\
& \cdots \quad -0.003 \quad -0.02 \quad 2.48 \quad 0.88 \quad -0.003 \quad -0.02 \quad 2.45 \quad 0.88 \\
& \cdots \quad -0.003 \quad -0.02 \quad 2.4 \quad 0.88 \quad -0.003 \quad -0.02 \quad 2.3 \quad 0.88 \\
& \cdots \quad -0.005 \quad -0.02 \quad 2.3 \quad 0.88 \quad -0.005 \quad -0.02 \quad 2.48 \quad 0.88 \\
& \cdots \quad -0.005 \quad -0.02 \quad 2.3 \quad 0.88 \quad -0.005 \quad -0.02 \quad 2.4 \quad 0.88 \\
& \cdots \quad -0.005 \quad -0.02 \quad 2.4 \quad 0.88 \quad -0.005 \quad -0.02 \quad 2.3 \quad 0.88 \quad \Big]^T
\end{aligned} \tag{C.3}$$

$$\begin{array}{l} \hat{\boldsymbol{x}}_0^3 = \Big[\ 0.004 \quad 0.02 \quad 2.6 \quad 0.94 \quad 0.002 \quad 0.02 \quad 2.7 \quad 0.94 \\ \begin{array}{ccccccccc} \cdots & 0.002 & 0.02 & 2.7 & 0.94 & 0.002 & 0.02 & 2.9 & 0.94 \\ \cdots & 0.005 & 0.03 & 2.9 & 0.94 & 0.005 & 0.03 & 2.53 & 0.94 \\ \cdots & 0.005 & 0.03 & 2.6 & 0.94 & 0.005 & 0.03 & 2.7 & 0.94 \\ \cdots & 0.005 & 0.03 & 2.9 & 0.94 & 0.006 & 0.03 & 2.9 & 0.94 \\ \cdots & 0.006 & 0.03 & 2.53 & 0.94 & 0.006 & 0.03 & 2.6 & 0.94 \\ \cdots & 0.006 & 0.03 & 2.7 & 0.94 & 0.006 & 0.03 & 2.9 & 0.94 \\ \cdots & 0.007 & 0.03 & 2.9 & 0.94 & 0.007 & 0.03 & 2.53 & 0.94 \\ \cdots & 0.007 & 0.03 & 2.9 & 0.94 & 0.007 & 0.03 & 2.7 & 0.94 \\ \cdots & 0.007 & 0.03 & 2.7 & 0.94 & 0.007 & 0.03 & 2.9 & 0.94\Big]^T \end{array} \end{array} \tag{C.4}$$

$$\begin{array}{l} \hat{\boldsymbol{x}}_0^4 = \Big[\ -0.004 \quad -0.02 \quad 2.4 \quad 0.86 \quad -0.002 \quad -0.02 \quad 2.3 \quad 0.86 \\ \begin{array}{ccccccccc} \cdots & -0.002 & -0.02 & 2.3 & 0.86 & -0.002 & -0.02 & 2.1 & 0.86 \\ \cdots & -0.005 & -0.03 & 2.1 & 0.86 & -0.005 & -0.03 & 2.47 & 0.86 \\ \cdots & -0.005 & -0.03 & 2.4 & 0.86 & -0.005 & -0.03 & 2.3 & 0.86 \\ \cdots & -0.005 & -0.03 & 2.1 & 0.86 & -0.006 & -0.03 & 2.1 & 0.86 \\ \cdots & -0.006 & -0.03 & 2.47 & 0.86 & -0.006 & -0.03 & 2.4 & 0.86 \\ \cdots & -0.006 & -0.03 & 2.3 & 0.86 & -0.006 & -0.03 & 2.1 & 0.86 \\ \cdots & -0.007 & -0.03 & 2.1 & 0.86 & -0.007 & -0.03 & 2.47 & 0.86 \\ \cdots & -0.007 & -0.03 & 2.1 & 0.86 & -0.007 & -0.03 & 2.3 & 0.86 \\ \cdots & -0.007 & -0.03 & 2.3 & 0.86 & -0.007 & -0.03 & 2.1 & 0.86 \quad \Big]^T \end{array} \end{array} \tag{C.5}$$

$$\begin{array}{l} \hat{\boldsymbol{x}}_0^5 = \Big[\ 0.006 \quad 0.03 \quad 2.65 \quad 0.96 \quad 0.004 \quad 0.03 \quad 2.8 \quad 0.96 \\ \begin{array}{ccccccccc} \cdots & 0.004 & 0.03 & 2.8 & 0.96 & 0.004 & 0.03 & 3 & 0.96 \\ \cdots & 0.007 & 0.04 & 3 & 0.96 & 0.007 & 0.04 & 2.54 & 0.96 \\ \cdots & 0.007 & 0.04 & 2.7 & 0.96 & 0.007 & 0.04 & 2.8 & 0.96 \\ \cdots & 0.007 & 0.04 & 3 & 0.96 & 0.008 & 0.04 & 3 & 0.96 \\ \cdots & 0.008 & 0.04 & 2.54 & 0.96 & 0.008 & 0.04 & 2.7 & 0.96 \\ \cdots & 0.008 & 0.04 & 2.8 & 0.96 & 0.008 & 0.04 & 3 & 0.96 \\ \cdots & 0.01 & 0.04 & 3 & 0.96 & 0.01 & 0.04 & 2.54 & 0.96 \\ \cdots & 0.01 & 0.04 & 3 & 0.96 & 0.01 & 0.04 & 2.8 & 0.96 \\ \cdots & 0.01 & 0.04 & 2.8 & 0.96 & 0.01 & 0.04 & 3 & 0.96 \quad \Big]^T \end{array} \end{array} \tag{C.6}$$

$$\begin{array}{lrrrrrrrrl}
\hat{\boldsymbol{x}}_0^6 = \Big[& -0.006 & -0.03 & 2.35 & 0.84 & -0.004 & -0.03 & 2.2 & 0.84 & \\
\cdots & -0.004 & -0.03 & 2.2 & 0.84 & -0.004 & -0.03 & 2 & 0.84 & \\
\cdots & -0.007 & -0.04 & 2 & 0.84 & -0.007 & -0.04 & 2.46 & 0.84 & \\
\cdots & -0.007 & -0.04 & 2.3 & 0.84 & -0.007 & -0.04 & 2.2 & 0.84 & \\
\cdots & -0.007 & -0.04 & 2 & 0.84 & -0.008 & -0.04 & 2 & 0.84 & \\
\cdots & -0.008 & -0.04 & 2.46 & 0.84 & -0.008 & -0.04 & 2.3 & 0.84 & \\
\cdots & -0.008 & -0.04 & 2.2 & 0.84 & -0.008 & -0.04 & 2 & 0.84 & \\
\cdots & -0.01 & -0.04 & 2 & 0.84 & -0.01 & -0.04 & 2.46 & 0.84 & \\
\cdots & -0.01 & -0.04 & 2 & 0.84 & -0.01 & -0.04 & 2.2 & 0.84 & \\
\cdots & -0.01 & -0.04 & 2.2 & 0.84 & -0.01 & -0.04 & 2 & 0.84 & \Big]^T
\end{array} \tag{C.7}$$

$$\begin{array}{lrrrrrrrrl}
\hat{\boldsymbol{x}}_0^7 = \Big[& 0.008 & 0.04 & 2.7 & 0.98 & 0.006 & 0.04 & 2.9 & 0.98 & \\
\cdots & 0.006 & 0.04 & 2.9 & 0.98 & 0.006 & 0.04 & 3.05 & 0.98 & \\
\cdots & 0.01 & 0.05 & 3.05 & 0.98 & 0.01 & 0.05 & 2.55 & 0.98 & \\
\cdots & 0.01 & 0.05 & 2.8 & 0.98 & 0.01 & 0.05 & 2.9 & 0.98 & \\
\cdots & 0.01 & 0.05 & 3.05 & 0.98 & 0.01 & 0.05 & 3.05 & 0.98 & \\
\cdots & 0.01 & 0.05 & 2.55 & 0.98 & 0.01 & 0.05 & 2.8 & 0.98 & \\
\cdots & 0.01 & 0.05 & 2.9 & 0.98 & 0.01 & 0.05 & 3.05 & 0.98 & \\
\cdots & 0.02 & 0.05 & 3.05 & 0.98 & 0.02 & 0.05 & 2.55 & 0.98 & \\
\cdots & 0.02 & 0.05 & 3.05 & 0.98 & 0.02 & 0.05 & 2.9 & 0.98 & \\
\cdots & 0.02 & 0.05 & 2.9 & 0.98 & 0.02 & 0.05 & 3.05 & 0.98 & \Big]^T
\end{array} \tag{C.8}$$

$$\begin{array}{lrrrrrrrrl}
\hat{\boldsymbol{x}}_0^8 = \Big[& -0.008 & -0.04 & 2.3 & 0.82 & -0.006 & -0.04 & 2.1 & 0.82 & \\
\cdots & -0.006 & -0.04 & 2.1 & 0.82 & -0.006 & -0.04 & 1.95 & 0.82 & \\
\cdots & -0.01 & -0.05 & 1.95 & 0.82 & -0.01 & -0.05 & 2.45 & 0.82 & \\
\cdots & -0.01 & -0.05 & 2.2 & 0.82 & -0.01 & -0.05 & 2.1 & 0.82 & \\
\cdots & -0.01 & -0.05 & 1.95 & 0.82 & -0.01 & -0.05 & 1.95 & 0.82 & \\
\cdots & -0.01 & -0.05 & 2.45 & 0.82 & -0.01 & -0.05 & 2.2 & 0.82 & \\
\cdots & -0.01 & -0.05 & 2.1 & 0.82 & -0.01 & -0.05 & 1.95 & 0.82 & \\
\cdots & -0.02 & -0.05 & 1.95 & 0.82 & -0.02 & -0.05 & 2.45 & 0.82 & \\
\cdots & -0.02 & -0.05 & 1.95 & 0.82 & -0.02 & -0.05 & 2.1 & 0.82 & \\
\cdots & -0.02 & -0.05 & 2.1 & 0.82 & -0.02 & -0.05 & 1.95 & 0.82 & \Big]^T
\end{array} \tag{C.9}$$

$$
\begin{array}{l}
\hat{\boldsymbol{x}}_0^9 = \Big[\begin{array}{cccccccc} 0.01 & 0.05 & 2.75 & 1 & 0.008 & 0.05 & 2.95 & 1 \end{array} \\
\quad \begin{array}{ccccccccc}
\cdots & 0.008 & 0.05 & 3 & 1 & 0.008 & 0.05 & 3.1 & 1 \\
\cdots & 0.015 & 0.06 & 3.1 & 1 & 0.015 & 0.06 & 2.56 & 1 \\
\cdots & 0.015 & 0.06 & 2.85 & 1 & 0.015 & 0.06 & 3 & 1 \\
\cdots & 0.015 & 0.06 & 3.1 & 1 & 0.02 & 0.06 & 3.1 & 1 \\
\cdots & 0.02 & 0.06 & 2.56 & 1 & 0.02 & 0.06 & 2.85 & 1 \\
\cdots & 0.02 & 0.06 & 3 & 1 & 0.02 & 0.06 & 3.1 & 1 \\
\cdots & 0.025 & 0.06 & 3.1 & 1 & 0.025 & 0.06 & 2.56 & 1 \\
\cdots & 0.025 & 0.06 & 3.1 & 1 & 0.025 & 0.06 & 3 & 1 \\
\cdots & 0.025 & 0.06 & 3 & 1 & 0.025 & 0.06 & 3.1 & 1
\end{array} \Big]^T
\end{array}
\tag{C.10}
$$

$$
\begin{array}{l}
\hat{\boldsymbol{x}}_0^{10} = \Big[\begin{array}{cccccccc} -0.01 & -0.05 & 2.25 & 0.8 & -0.008 & -0.05 & 2.05 & 0.8 \end{array} \\
\begin{array}{ccccccccc}
\cdots & -0.008 & -0.05 & 2 & 0.8 & -0.008 & -0.05 & 1.9 & 0.8 \\
\cdots & -0.015 & -0.06 & 1.9 & 0.8 & -0.015 & -0.06 & 2.44 & 0.8 \\
\cdots & -0.015 & -0.06 & 2.15 & 0.8 & -0.015 & -0.06 & 2 & 0.8 \\
\cdots & -0.015 & -0.06 & 1.9 & 0.8 & -0.02 & -0.06 & 1.9 & 0.8 \\
\cdots & -0.02 & -0.06 & 2.44 & 0.8 & -0.02 & -0.06 & 2.15 & 0.8 \\
\cdots & -0.02 & -0.06 & 2 & 0.8 & -0.02 & -0.06 & 1.9 & 0.8 \\
\cdots & -0.025 & -0.06 & 1.9 & 0.8 & -0.025 & -0.06 & 2.44 & 0.8 \\
\cdots & -0.025 & -0.06 & 1.9 & 0.8 & -0.025 & -0.06 & 2 & 0.8 \\
\cdots & -0.025 & -0.06 & 2 & 0.8 & -0.025 & -0.06 & 1.9 & 0.8
\end{array} \Big]^T
\end{array}
\tag{C.11}
$$

Appendix D

Process noise simulations

Plots of all the states with and without process noise are presented here.

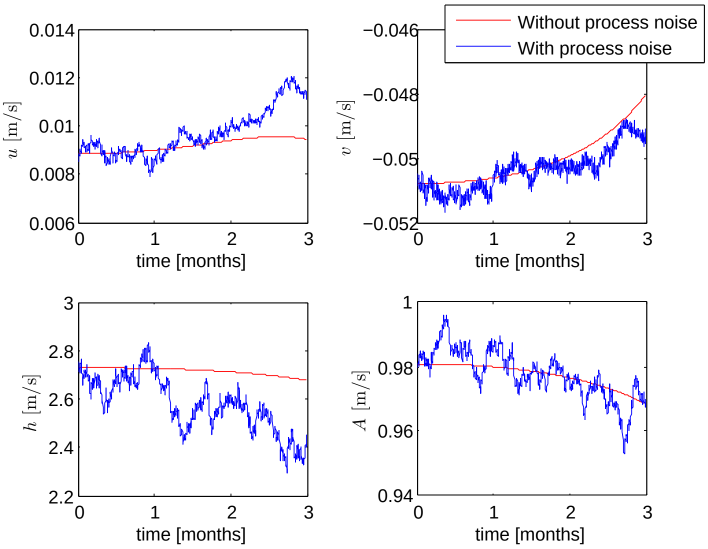


Figure D.1: The states of cell number 1 with and without process noise.

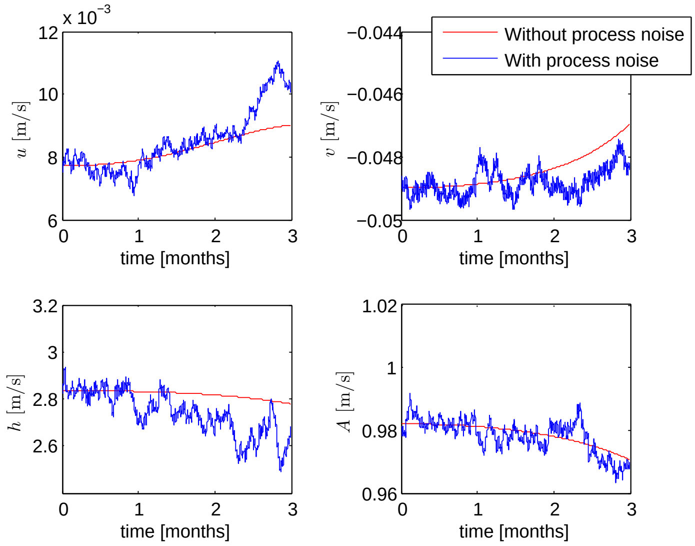


Figure D.2: The states of cell number 2 with and without process noise.

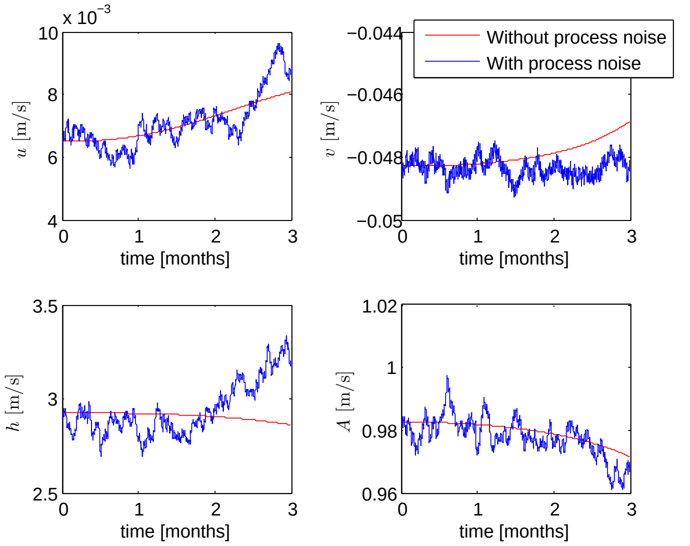


Figure D.3: The states of cell number 3 with and without process noise.

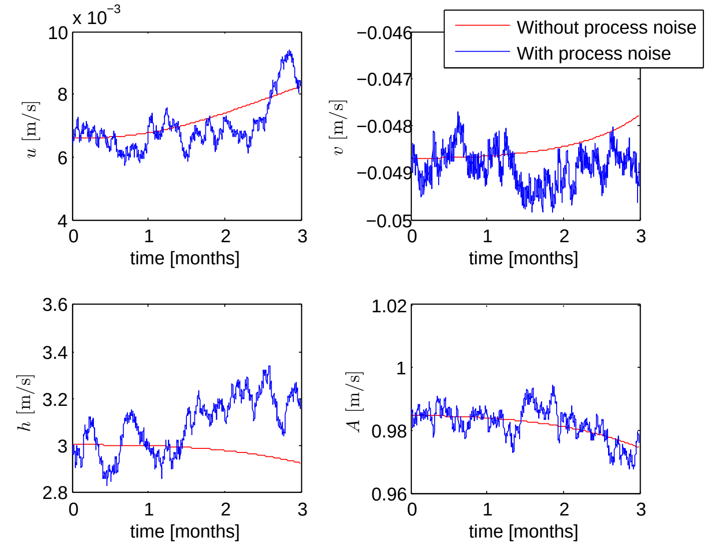


Figure D.4: The states of cell number 4 with and without process noise.

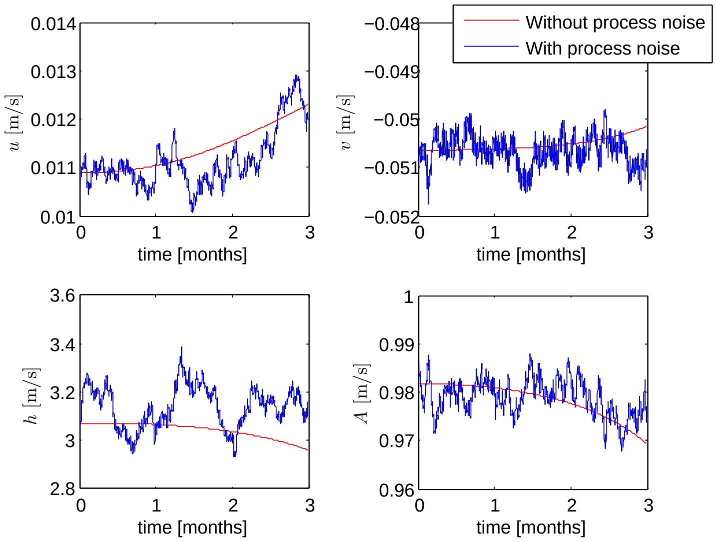


Figure D.5: The states of cell number 5 with and without process noise.

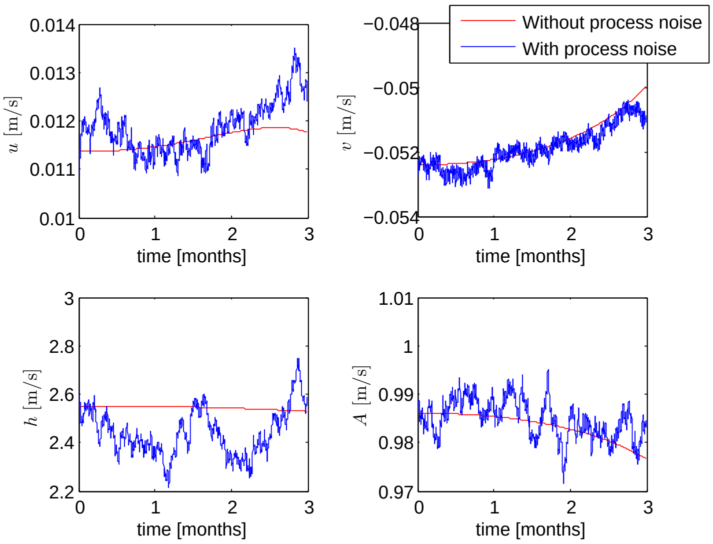


Figure D.6: The states of cell number 6 with and without process noise.

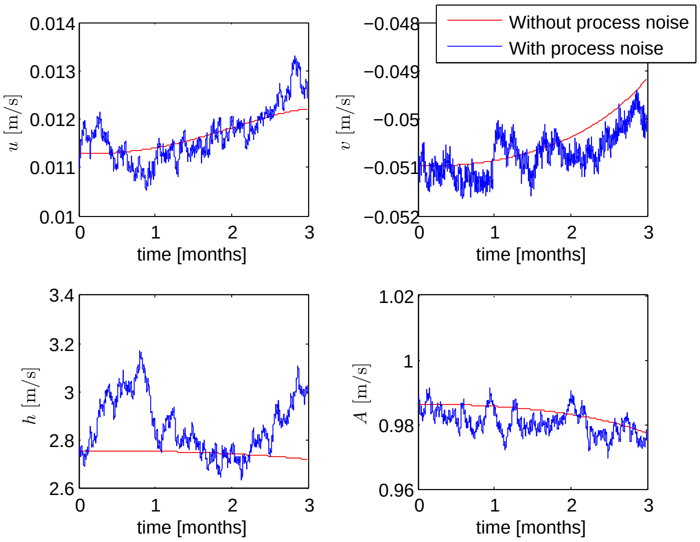


Figure D.7: The states of cell number 7 with and without process noise.

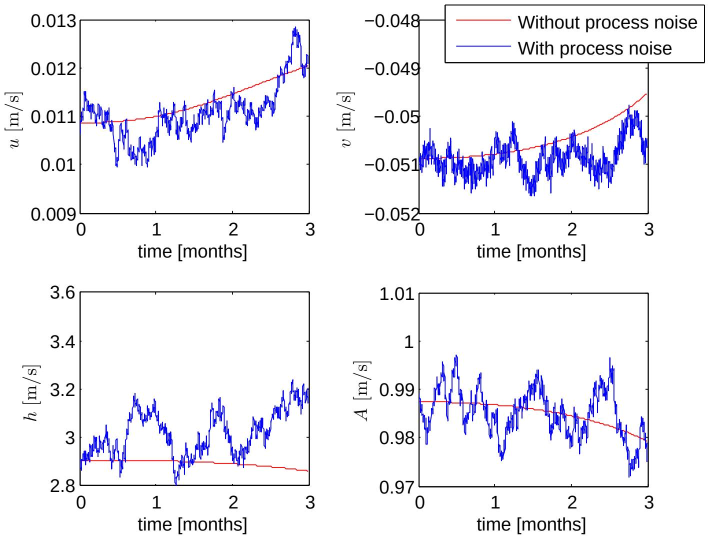


Figure D.8: The states of cell number 8 with and without process noise.

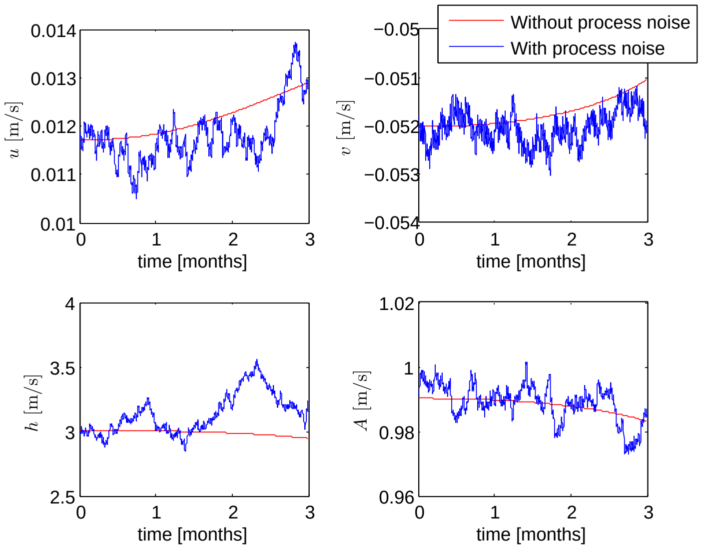


Figure D.9: The states of cell number 9 with and without process noise.

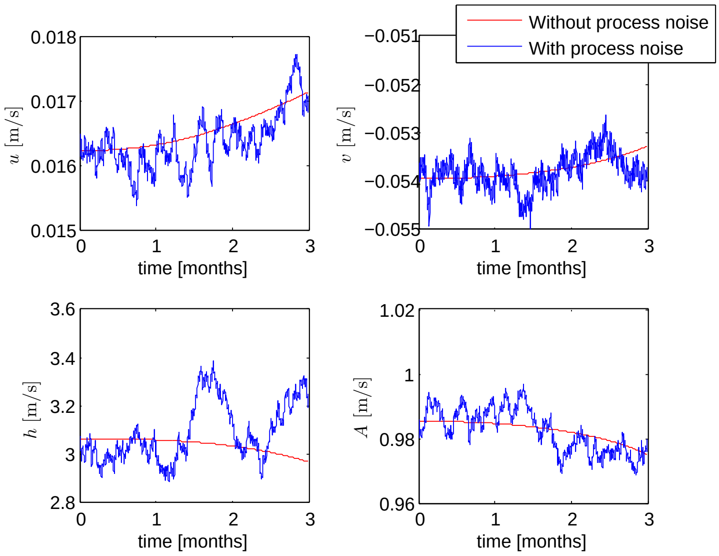


Figure D.10: The states of cell number 10 with and without process noise.

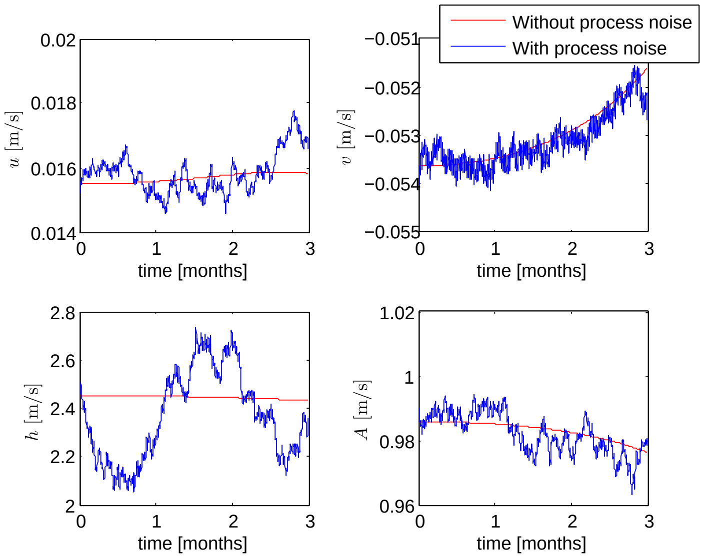


Figure D.11: The states of cell number 11 with and without process noise.

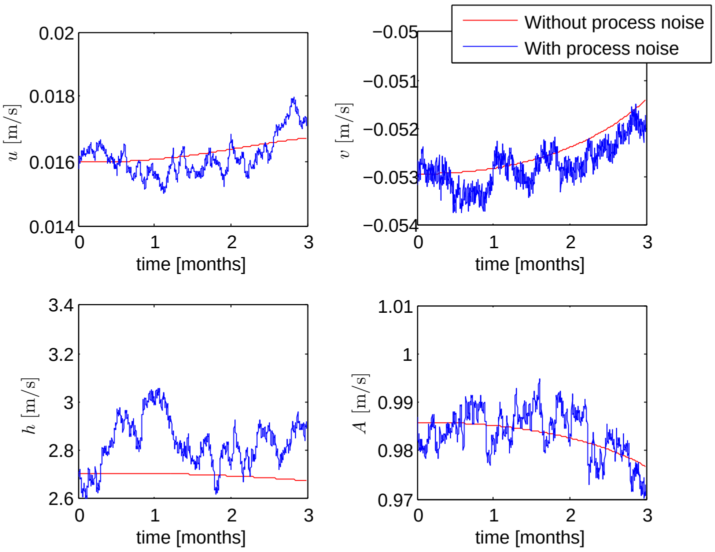


Figure D.12: The states of cell number 12 with and without process noise.

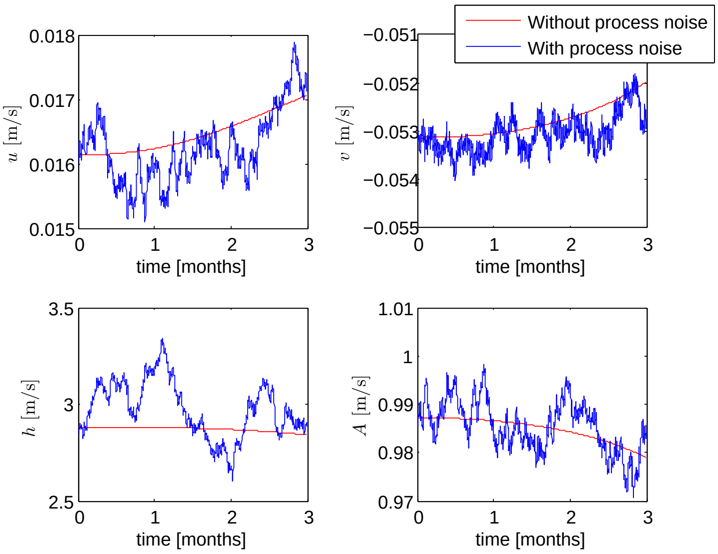


Figure D.13: The states of cell number 13 with and without process noise.

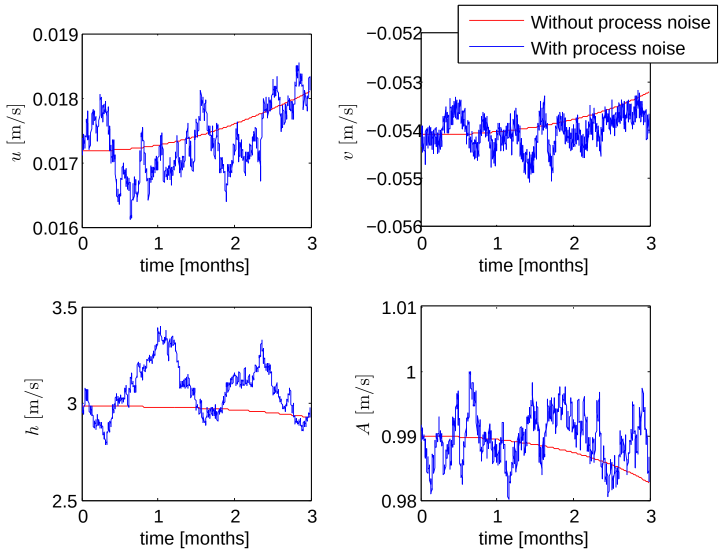


Figure D.14: The states of cell number 14 with and without process noise.

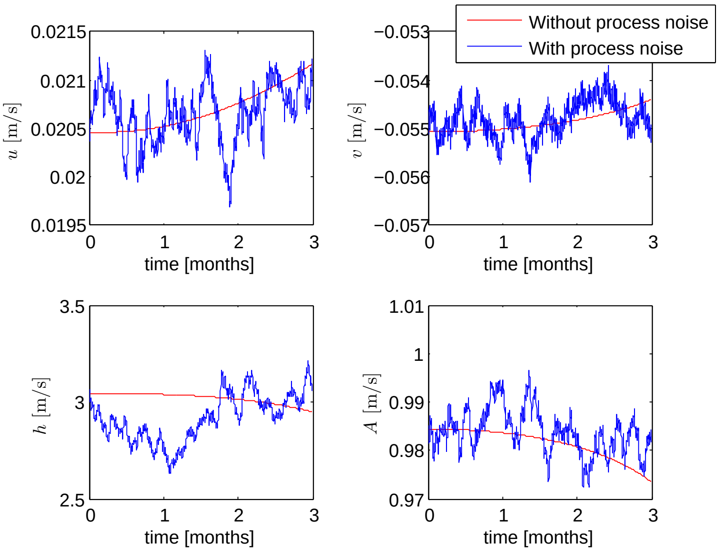


Figure D.15: The states of cell number 15 with and without process noise.

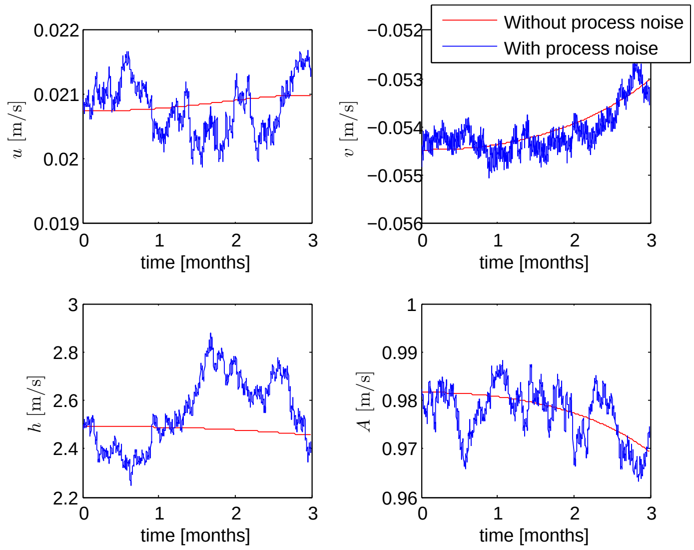


Figure D.16: The states of cell number 16 with and without process noise.

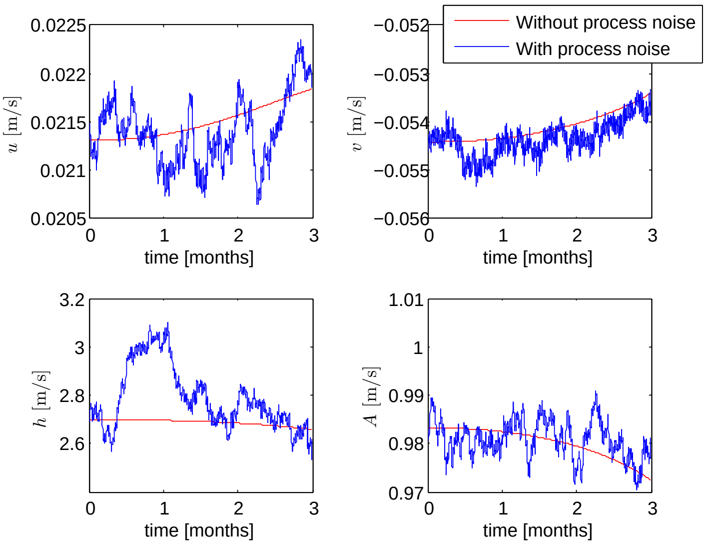


Figure D.17: The states of cell number 17 with and without process noise.

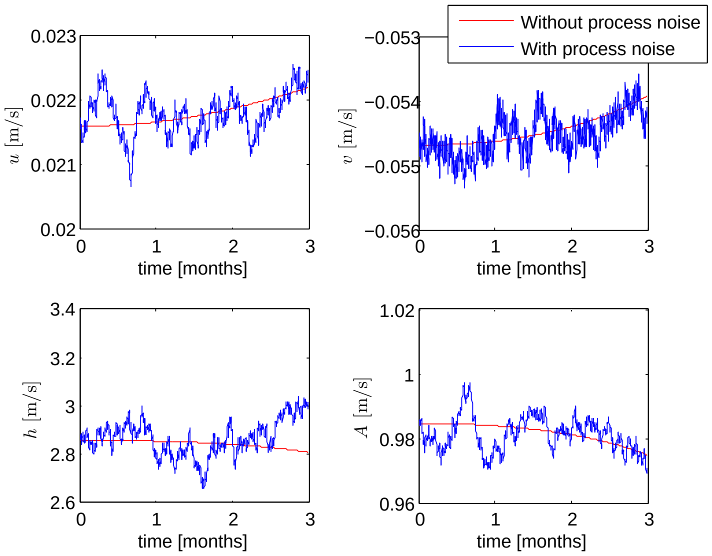


Figure D.18: The states of cell number 18 with and without process noise.

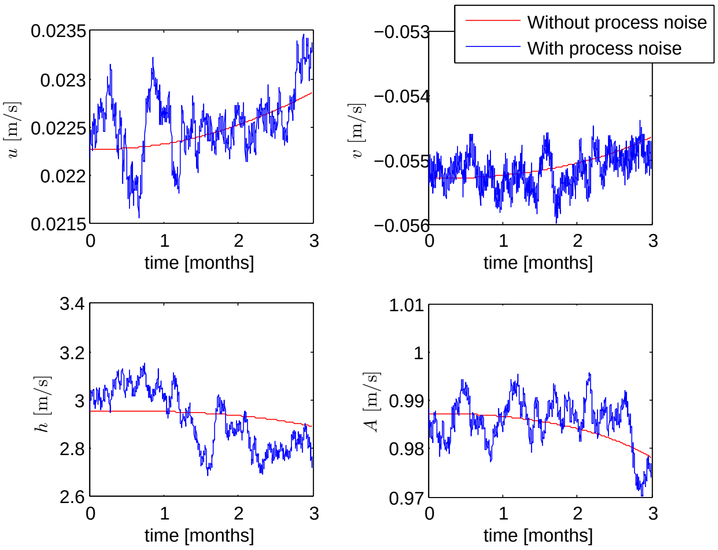


Figure D.19: The states of cell number 19 with and without process noise.

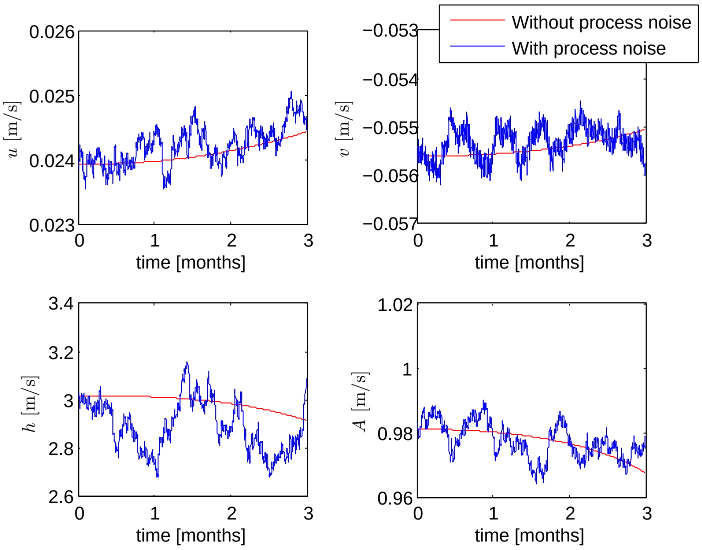


Figure D.20: The states of cell number 20 with and without process noise.

Appendix E

Estimation results

Plots of all the state estimates are presented here.

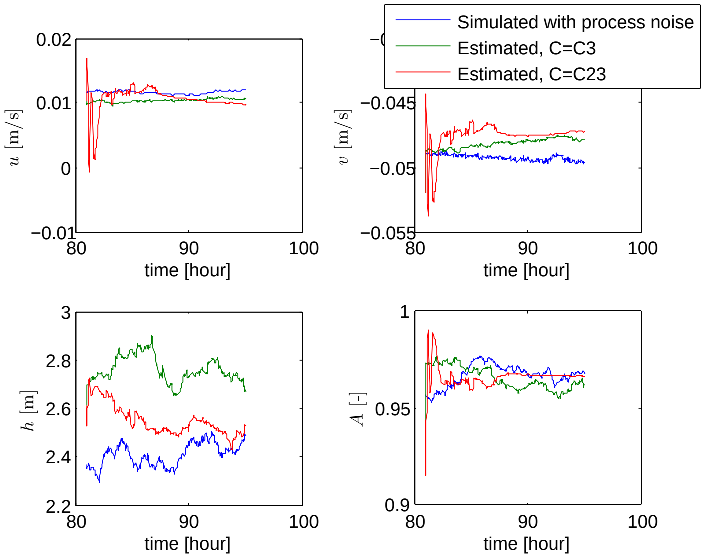


Figure E.1: Simulated and estimated states of cell number 1.

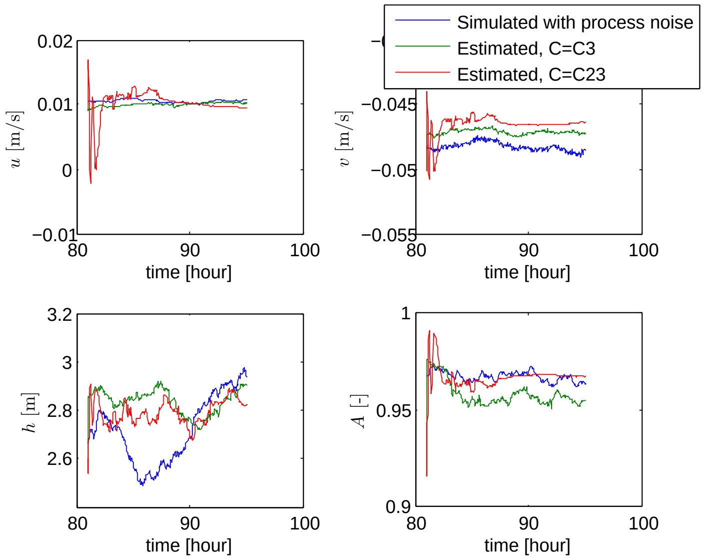


Figure E.2: Simulated and estimated states of cell number 2.

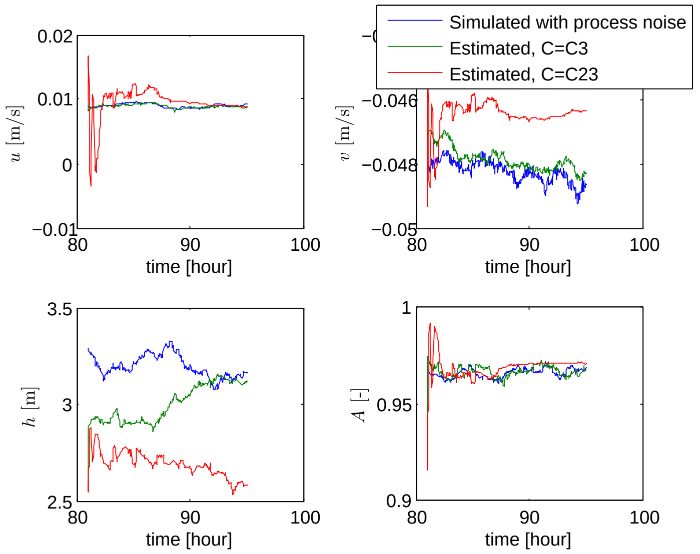


Figure E.3: Simulated and estimated states of cell number 3.

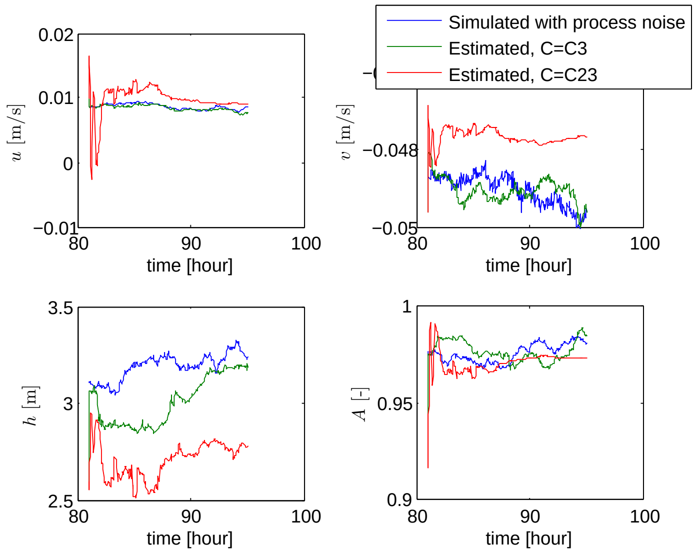


Figure E.4: Simulated and estimated states of cell number 4.

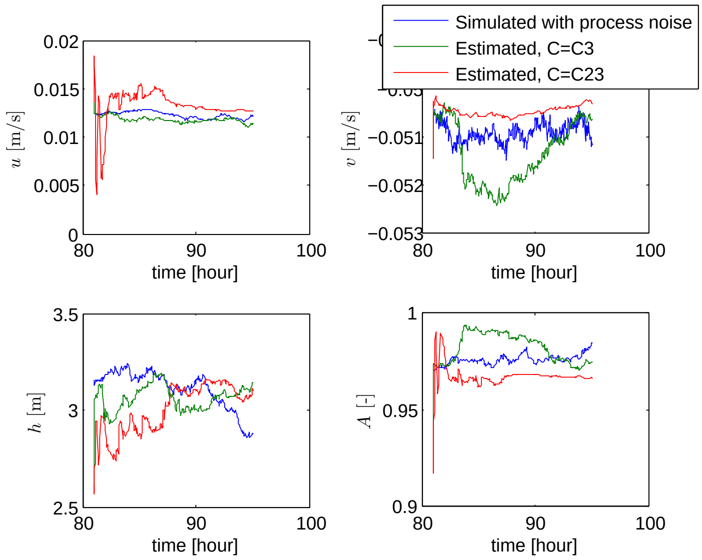


Figure E.5: Simulated and estimated states of cell number 5.

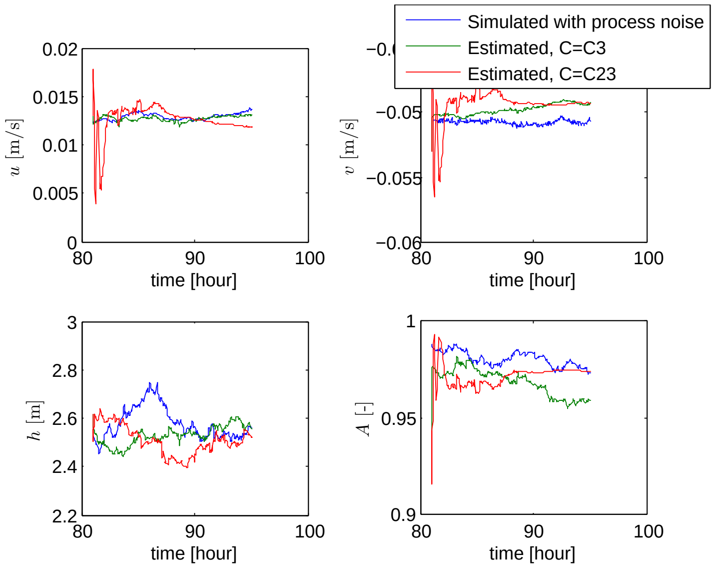


Figure E.6: Simulated and estimated states of cell number 6.

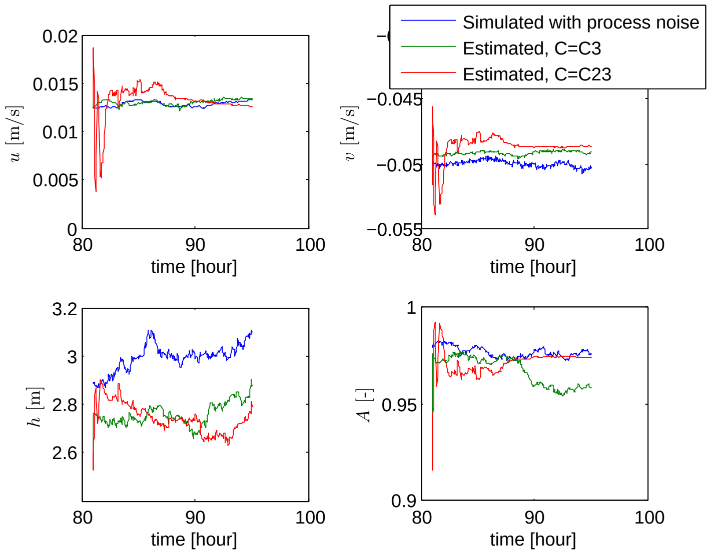


Figure E.7: Simulated and estimated states of cell number 7.

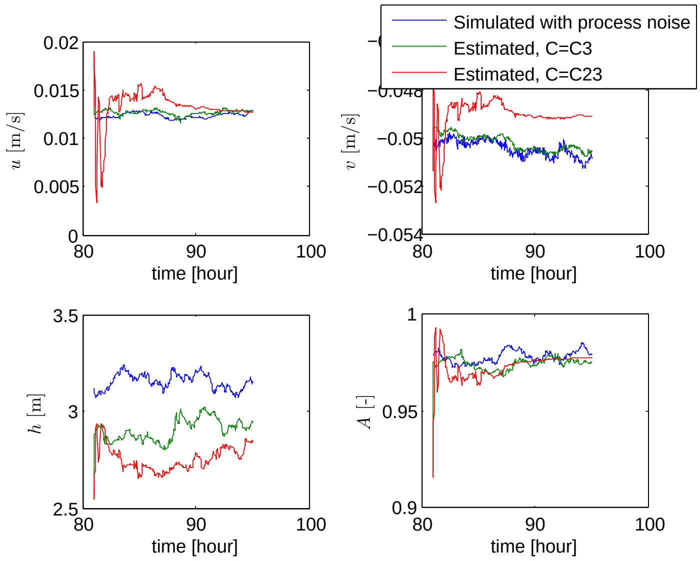


Figure E.8: Simulated and estimated states of cell number 8.

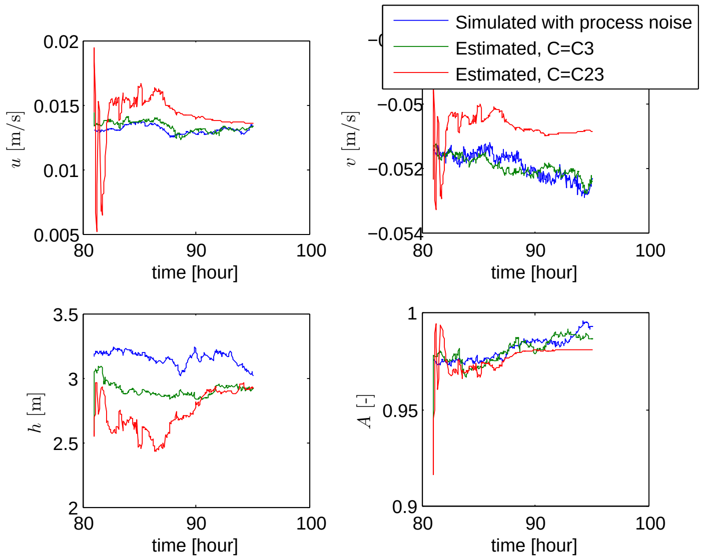


Figure E.9: Simulated and estimated states of cell number 9.

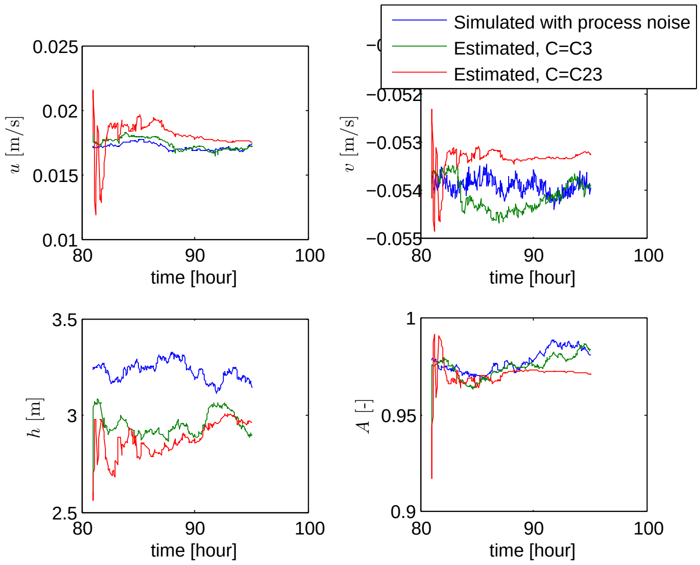


Figure E.10: Simulated and estimated states of cell number 10.

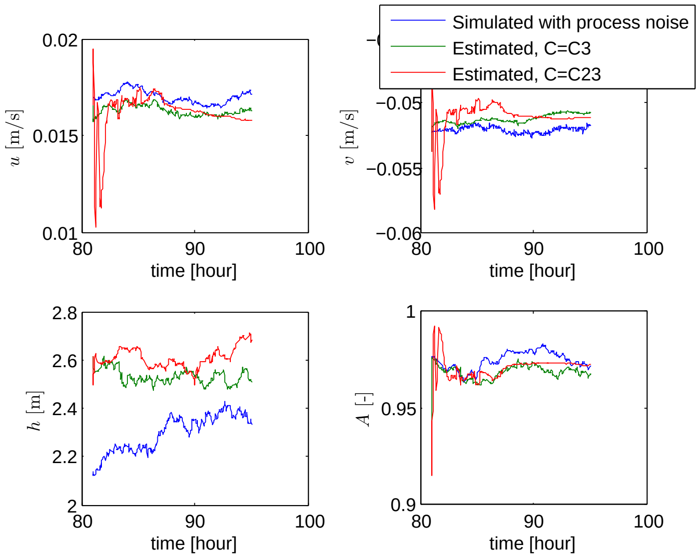


Figure E.11: Simulated and estimated states of cell number 11.

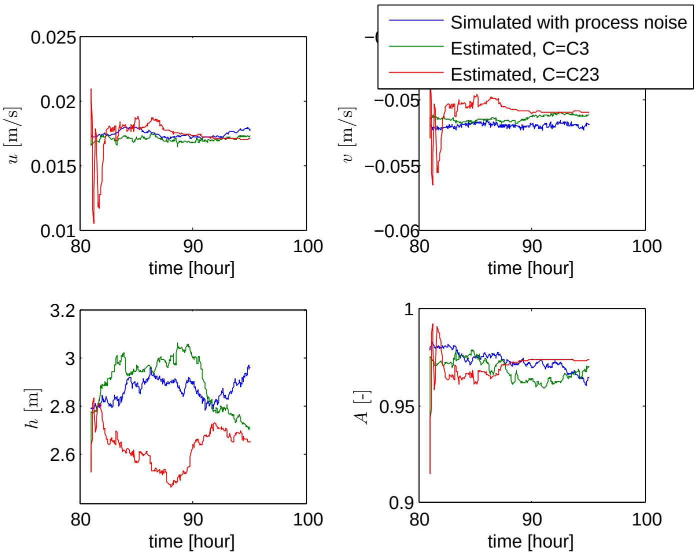


Figure E.12: Simulated and estimated states of cell number 12.

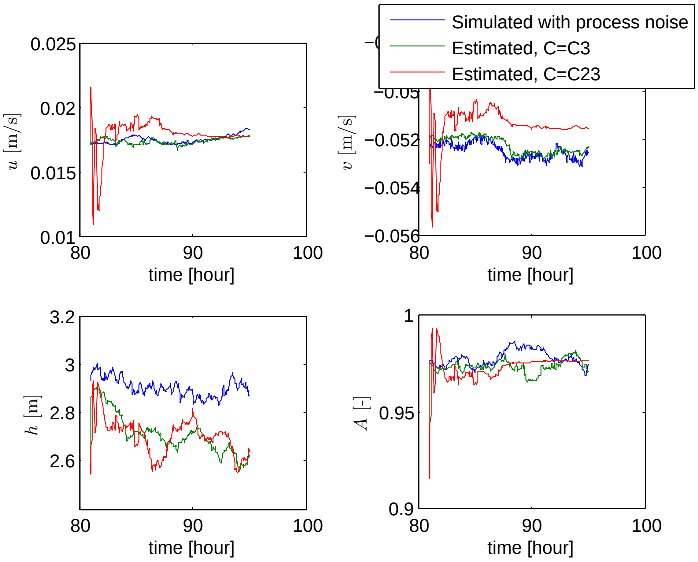


Figure E.13: Simulated and estimated states of cell number 13.

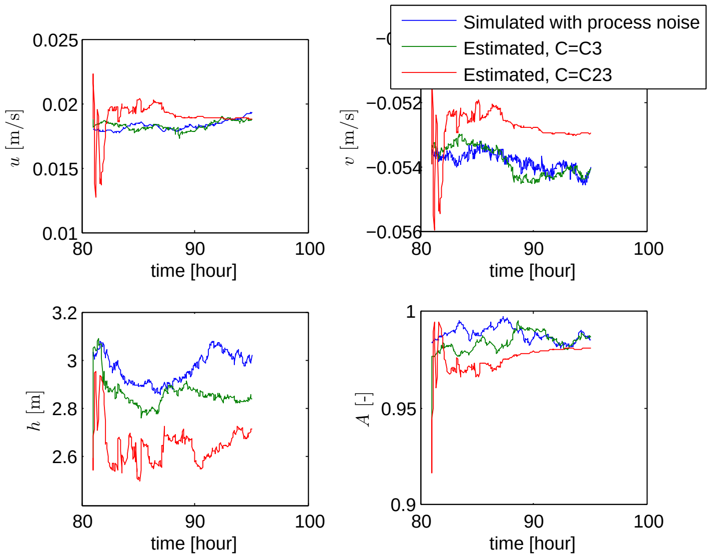


Figure E.14: Simulated and estimated states of cell number 14.

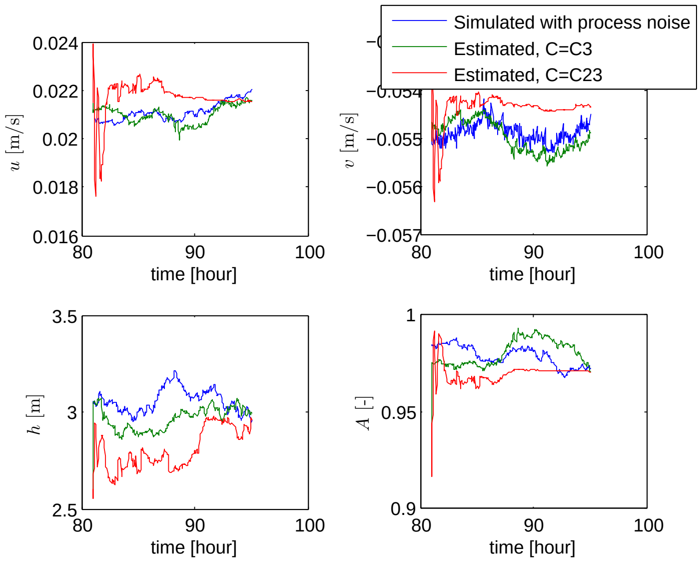


Figure E.15: Simulated and estimated states of cell number 15.

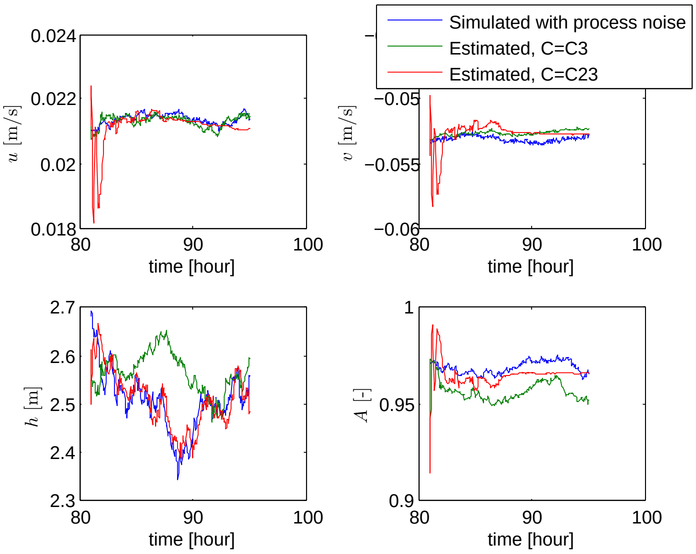


Figure E.16: Simulated and estimated states of cell number 16.

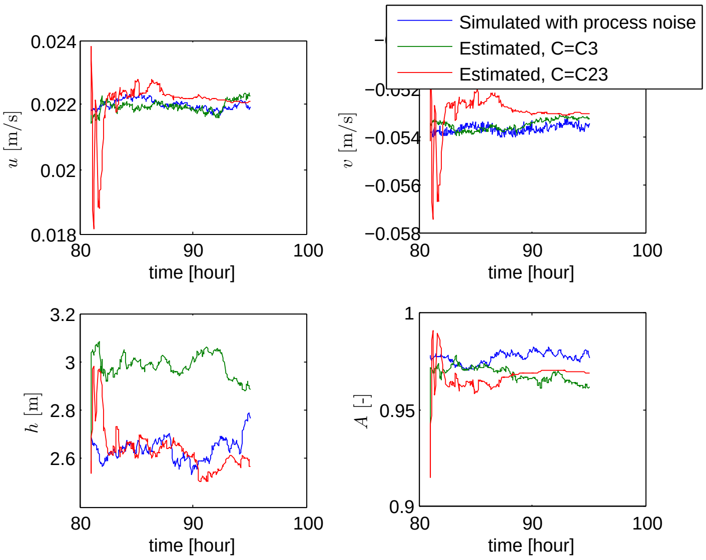


Figure E.17: Simulated and estimated states of cell number 17.

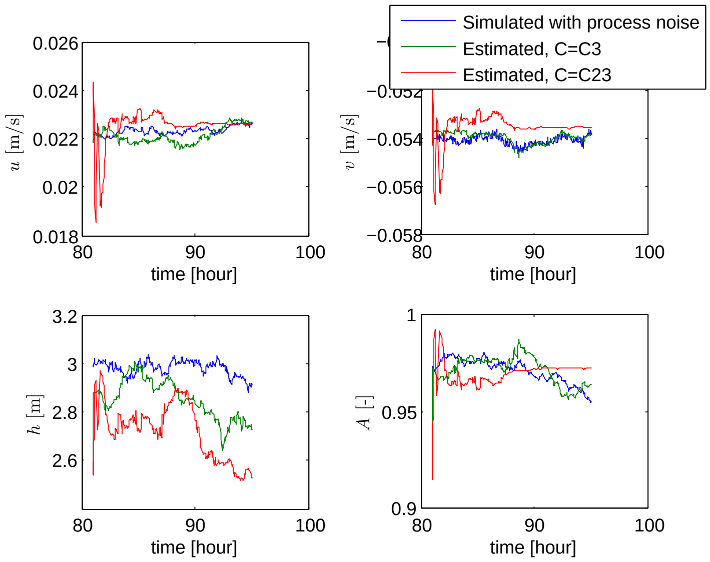


Figure E.18: Simulated and estimated states of cell number 18.

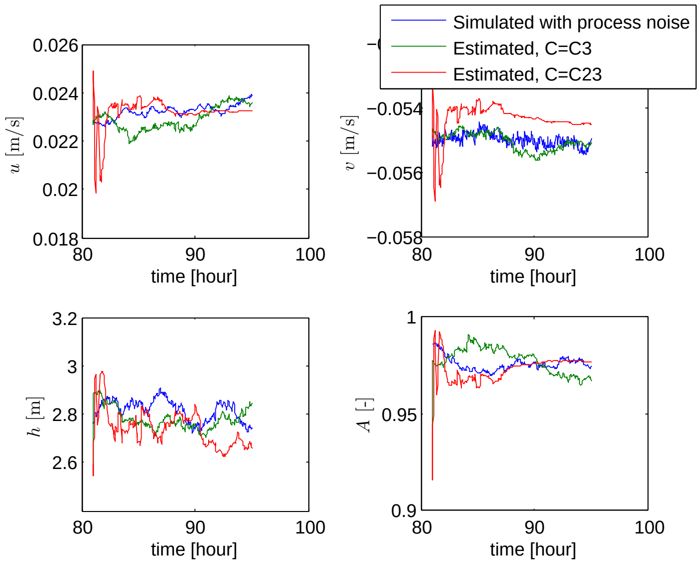


Figure E.19: Simulated and estimated states of cell number 19.

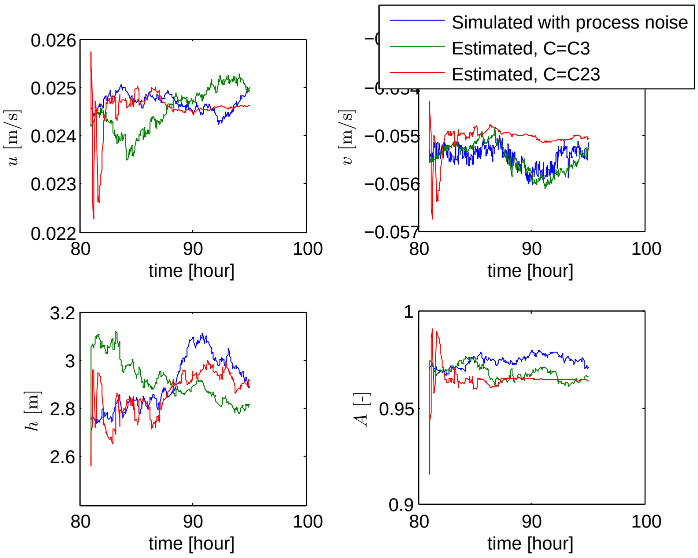


Figure E.20: Simulated and estimated states of cell number 20.