NTNU

Norwegian University of
Science and Technology

# Building Ice Maps Using Mobile Aerial Sensors and Grid-based Estimation

Cathrine Bruteig

Master of Science in Cybernetics and Robotics
Submission date: December 2015
Supervisor: Lars Imsland, ITK

Norwegian University of Science and Technology
Department of Engineering Cybernetics

NTNU

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

# PROJECT DESCRIPTION SHEET

**Name of the candidate:** Cathrine Bruteig

**Thesis title (English):** Building Ice Maps Using Mobile Aerial Sensors and Grid-based Estimation

## Background

As offshore oil- and gas production enters arctic seas, the presence of ice becomes a substantial challenge in station-keeping operations. An important part of such operations in the Arctic, is a system for doing *ice management*, that is, gather information about, and physically control, the ice environment. The required information about ice properties may come from (the combination of) several sources, but of interest in this project is sensors (e.g. optical or radar) that are carried by unmanned aerial vehicles (UAVs).

The topic of this project, is to investigate grid-based estimation methods as a tool for map-building and subsequently online, possibly autonomous, UAV path planning. The project will implement Occupancy Grid Mapping techniques from probabilistic robotics as a starting point, look into possibilities for extensions and improvements (as these techniques are developed for a somewhat different purpose), before finally (if time allows) propose ways that the obtained information can be used for UAV path planning.

## Work description

1. Give a *brief* overview of ice management challenges, and how UAVs can be used for surveillance and estimation of ice.
2. Describe Occupancy Grid Mapping (OGM); background, algorithm, assumptions for use (strengths and weaknesses). Make an implementation for building a grid-based map of ice/not-ice in a rectangular area based on conceivable processed data from UAV measurements.
3. Outline (and as far as time permits, implement and test) a procedure for using the information in the OGM methods for (autonomous) path planning for a mobile sensor (e.g. a UAV).
4. Discuss the findings, and conclude.

**Start date:** August 4, 2015 **Due date:** December 21, 2015

**Supervisor:** Lars Imsland
**Co-advisor(s):**

**Trondheim,** __August 4th, 2015_____

**Lars Imsland**
Supervisor

**Address**
Sem Sælandsvei 5
NO-7491 Trondheim

**Org.no.** 974 767 880
E-mail:
postmottak@itk.ntnu.no
http://www.itk.ntnu.no

**Location**
O.S. Bragstads plass 2D
NO-7034 Trondheim

**Phone**
+ 47 73 59 43 76
**Fax**
+ 47 73 59 45 99

Phone: + 47 47 23 19 49

# Preface

This thesis is the final work done in my master's degree in Engineering Cybernetics at Norwegian University of Science and Technology (NTNU) and was carried out in the autumn of 2015.

I would like to thank my supervisor Lars Imsland for all his guidance and support. I would also like to thank Anders Albert for his collaboration in the last part of the project.

Finally, I would like to thank all my friends for making my years at NTNU special and unforgettable.

Cathrine Bruteig
Trondheim, December 2015

# Abstract

Operations in the Arctic seas are becoming more common as the area contains large amounts of oil and gas. For these operations, control and detection of ice are necessary to avoid interference with the production. The focus of this project was to use the recursive grid-based estimation method, Occupancy Grid Mapping, to locate icebergs using an Unmanned Aerial Vehicle (UAV). The Occupancy Grid Mapping is not usually used for iceberg localization from the air. The goal of this project was, therefore, to show how Occupancy Grid Mapping can be used for iceberg localization with a UAV. The Occupancy Grid Mapping was also implemented and simulated with a path planner to show how Occupancy Grid Mapping can be used with other algorithms. The path planner algorithm and framework used in this project was created by the PhD student Anders Albert.

The UAV was simulated to fly over a given area, and the measurements from the UAV was used by the Occupancy Grid Mapping to calculate the probability of occupancy. Grid-based maps of the area were created based on the probability of occupancy calculated by the Occupancy Grid Mapping. These maps showed where icebergs were located and which parts of the area were empty. The accuracy of the Occupancy Grid Mapping estimation was best when the UAV measurement frequency was high. When the Occupancy Grid Mapping was simulated with the path planner, a proposed procedure was used to calculate the optimization variable of the path planner. The optimization variable was based on the probability of occupancy calculated from Occupancy Grid Mapping and time since the observation took place. The simulation results showed that the Occupancy Grid Mapping can be used by a path planner to find the optimal path between the icebergs.

This project showed that grid-based maps produced by Occupancy Grid Mapping can be used for iceberg localization. In further work, some adjustments of Occupancy Grid Mapping are proposed for further experimentations.

# Sammendrag

Operasjoner i Arktis blir stadig mer vanlig da området inneholder store mengder olje og gass. For disse operasjonene er kontroll og deteksjon av is nødvendig for å unngå forstyrrelse av produksjon. Fokuset i denne oppgaven var å bruke den rekursive grid-baserte beregningsmetoden, Occupancy Grid Mapping, for å finne isfjell ved hjelp av et ubemannet fly (UAV). Occupancy Grid Mapping brukes ikke for å lokalisere isfjell fra luften. Målet med denne oppgaven var derfor å vise hvordan Occupancy Grid Mapping kan brukes til å lokalisere isfjell ved hjelp av en UAV. Occupancy Grid Mapping ble også implementert og simulert med en baneplanlegger for å vise hvordan Occupancy Grid Mapping kan brukes sammen med andre algoritmer. Baneplanleggingsalgoritmen og rammeverket som ble brukt i dette prosjektet, ble utviklet av PhD studenten Anders Albert.

UAVen ble simulert til å fly over et gitt område, og målingene fra UAVen ble brukt av Occupancy Grid Mapping for å beregne sannsynligheten for isfjell. Et grid-basert kart over området ble laget basert på sannsynligheten for isfjell beregnet av Occupancy Grid Mapping. Disse kartene viste hvor isfjellene var og hvilke deler av området som var åpent. Nøyaktigheten av Occupancy Grid Mapping estimeringen var best når når målingene til UAVen ble utført ofte. Når Occupancy Grid Mapping ble simulert med banenplanleggeren, ble en prosedyre brukt til å beregne optimaliseringsvariabelen til banenplanleggeren. Optimaliseringsvariabelen var basert på sannsynligheten for isfjell beregnet fra Occupancy Grid Mapping og tiden siden observasjonen fant sted. Simuleringsresultatene viste at Occupancy Grid Mapping kan brukes av en baneplanlegger for å finne den optimale banen mellom isfjellene.

Dette prosjektet viste at grid-baserte kart som er produsert av Occupancy Grid Mapping, kan brukes til å lokalisere isfjell. Noen justeringer av Occupancy Grid Mapping foreslått for videre eksperimenter.

# Table of Contents

# List of Tables

# List of Figures

# Nomenclature

| | |
|---|---|
| $i$ | Grid cell index |
| $i'$ | Node index |
| $l_0$ | Prior of occupancy |
| $l_{i,t}$ | Log odds of occupancy for time $t$ and grid cell $i$ |
| $l_{\text{occ}}$ | Log odds of occupancy for occupied grids |
| $l_{\text{free}}$ | Log odds of occupancy for free grids |
| $\mathbf{m}_i$ | Map for each grid cell $i$ |
| $N$ | Number of nodes |
| $p(\mathbf{m}_i)$ | Initial probability |
| $p_{\text{occ}}$ | Probability of occupancy for occupied grids |
| $p_{\text{free}}$ | Probability of occupancy for free grids |
| $t$ | Time in seconds |
| $t'(i')$ | Nodes in visiting order |
| $u$ | Actuator of UAV |
| $U$ | Velocity of UAV |
| $v_i$ | Velocity of iceberg |
| $x_0$ | Initial position |
| $x_t$ | Position of observations at time $t$ |
| $x'_t$ | Pose of robot/UAV at time $t$ for OGM |
| $x_{UAV}$ | Position of UAV for the path planner |
| $y_{path}$ | UAV path |
| $z_t$ | Observation at time $t$ |

| | |
|---|---|
| $\epsilon_{\text{occ}}$ | Measurement error for occupied grids |
| $\epsilon_{\text{free}}$ | Measurement error for free grids |
| $\epsilon_{p_{\text{occ}}}$ | Probability of occupancy error for occupied grids |
| $\epsilon_{p_{\text{free}}}$ | Probability of occupancy error for free grids |
| $\xi_i$ | Position of iceberg |
| $\sigma_{nodes}$ | Position uncertainty |
| $\psi$ | Heading of UAV |

| | |
|---|---|
| **AUV** | Autonomous Underwater Vehicle |
| **GPS** | Global Positioning System |
| **MILP** | Mixed-Integer Linear Programming |
| **MPC** | Model Predictive Control |
| **OGM** | Occupancy Grid Mapping |
| **UAV** | Unmanned Aerial Vehicle |
| **TSP** | Travelling Salesman Problem |

Chapter 1

# Introduction

## 1.1 Background and Motivation

### 1.1.1 Ice management challenges

As the offshore industry moves closer into the Arctic, ice management becomes more important than before. Ice management is anything from ice breaking to the warning of closing icebergs and contributes to either reduced contact or ultimately avoiding contact of with the ice altogether. However, ice management faces some challenges, such as poor visibility and bad weather, as well as fast-moving ice. These challenges make it difficult to avoid the ice, due to inaccurate distance and positioning.

In the Arctic seas, major areas consist of ice and icebergs, which creates problems for station-keeping operations and offshore vessels. Ice and icebergs can cause significant damage, and interfere with expensive production. In worst case, the operation will be paused or even cancelled and aborted. An important part of the vessels and productions work is to have systems for ice management. An ice management system could be to use technology to monitor and estimate the positions of ice to notify the vessels in order to make it possible to avoid impact. There are different ways to avoid collision; the vessel can change course, and towing or destruction can relocate the iceberg. For station-keeping operations, changing course is not possible, and relocation of icebergs can be necessary.

### 1.1.2 Surveillance and estimation of ice with UAVs

Regardless of what kind of approach the vessels use to avoid impact, surveillance and estimation of the position of icebergs are essential for ice management. There are different

approaches and technologies for ice surveillance; satellites, airplanes, unmanned aerial vehicles (UAVs), radars, sonars, and autonomous underwater vehicles (AUVs). This project will address surveillance with UAVs. Today, UAVs are not used for ice management, but can play a significant role in ice surveillance in the future.

There are many advantages with the usage of UAVs. To monitor and search for ice and icebergs, UAVs can fly over the areas. UAVs are cheaper and easier to use than manned aerial vehicles such as airplanes. A UAV can be sent up from the vessel and start searching in the predefined area. Because the UAVs have limited flying time, predefined path planning is necessary to get most out of the flying time.

The UAV can send information directly or keep the information until it has landed on the vessel. In this project, a UAV will be used for the simulation of observing icebergs and estimating a probability of the iceberg position online, while flying.

## 1.2 Problem Description

For ice management and surveillance and estimation of ice, a UAV can be used. UAVs can use sensors from the air and take images to observe a specific area. This observation, with estimation methods, can be used to locate icebergs. Some estimations methods, called grids-based estimation methods, divide the area into grids to make the estimation possible. One grid-based estimation method is Occupancy Grid Mapping.

The main goal of this project is to look into if and how Occupancy Grid Mapping can be used for iceberg localization. Occupancy Grid Mapping will be implemented for building grid-based maps showing iceberg location based on UAV measurements. The UAV measurements will not be calculated from real observations, but will be simulated measurements and assumed to be known. Simulations of the iceberg localization will be conducted to test the implementation of Occupancy Grid Mapping.

The Occupancy Grid Mapping will also be tested with a UAV path planner, to see if and how Occupancy Grid Mapping can be used with other algorithms. The aim of the simulation of Occupancy Grid Mapping with the UAV path planner is to find the best way to use Occupancy Grid Mapping for iceberg localization. A procedure for using the information in Occupancy Grid Mapping for the path planning will be proposed. The purpose is not to develop a path planner, but to use Occupancy Grid Mapping together with a predeveloped path planner.

## 1.3 Previous Work

The grid-based estimation method Occupancy Grid Mapping is based on recursive Bayesian estimation theory, which have been researched for many years. Särkkä (2013) describes the Bayesian theory, and the project thesis Bruteig (2015) looks into grid-based Bayesian estimation for iceberg localization.

Occupancy Grid Mapping is developed for mapping of environments. A robot using sensors to locate obstacles in a room is the most common use of Occupancy Grid Mapping, which Thrun et al. (2006) describes. In this project, Occupancy Grid Mapping will be used for iceberg localization with the use of UAV, which is not a common usage of the method. For Occupancy Grid Mapping, robots use horizontal sensors to estimate the range to the obstacles, while UAVs use sensors to take images of the area from the air. In this project, the information from the images will be simulated and assumed known, but image processing is a familiar topic. Leira et al. (2015) looks into the usage of thermal cameras for detection, classification and tracking of objects in the ocean from UAVs.

UAV path planning between icebergs is the topic of Albert and Imsland (2015). As far as the writer knows, UAV path planning based on the output of Occupancy Grid Mapping have not yet been researched. In this project, Occupancy Grid Mapping with UAV path planning will thus be explored. The UAV path planner algorithm and framework, used in this project, was made by the PhD student Anders Albert. The UAV path planner does not consider the situations when the icebergs are not in the estimated positions. Andersson et al. (2015) discusses iceberg drift and forecast, which is a challenging process. Iceberg drift is thus a topic outside this project.

## 1.4 Overview

This project will be divided into two parts; simulation of Occupancy Grid Mapping, and simulation of Occupancy Grid Mapping with the UAV path planner.

The theory presented in Chapter 2 is the theoretical background of both the Occupancy Grid Mapping and the path planner algorithm used in this project.

Chapter 3 will explain how the Occupancy Grid Mapping was implemented and simulated for different cases. This chapter will give the reader an understanding of how the Occupancy Grid Mapping can be used for iceberg localization.

In Chapter 4, the Occupancy Grid Mapping , implemented in Chapter 3, will for different cases be tested and simulated together with the UAV path planning algorithm. This chapter will give the reader an insight into what can kind of algorithm the OGM be used with. This part of the project will be realistic simulations with actual UAV properties, such as the flight path, speed and turning radius.

Both Chapter 3 and Chapter 4 will include the simulation results, because the simulations in Chapter 4 is based on the knowledge of the simulations from Chapter 3.

The results from each simulation will be discussed in the result sections, while in Chapter 5 will the results from Chapter 3 and Chapter 4 be discussed against each other. The last chapter will conclude based on the work done in this project, and propose further work.

Chapter 2

# Theory

The main focus of this project is iceberg localization. A method for localization and presentation of the data will be implemented. The aim of the project is to see if this method can be used for iceberg localization using aerial images from a UAV. The localization method will also be implemented with a path planning algorithm to test the practical applicability. A theoretical framework is necessary to understand the usage of the algorithms.

## 2.1 Iceberg Localization

In this project, the iceberg localization will be based on a mapping algorithm from the robotics, the Occupancy Grid Mapping algorithm. To understand the theory of this algorithm, the theory of the recursive Bayesian estimation is necessary. An early version of Section 2.1.1 appeared in Bruteig (2015), while the theory in Section 2.1.2 is based on Thrun et al. (2006).

### 2.1.1 Recursive Bayesian Estimation

Bayesian filtering, also known as Bayes filtering, is a probabilistic recursive estimation algorithm. Filtering, in this context, refers to methods that use different measurements over time to produce estimates and predictions of unknown variables. The most basic form of the Bayesian filtering is the Kalman filter. Bayesian filtering can be used in fields like navigation, aerospace engineering, remote surveillance, control engineering and many more (Särkkä, 2013). Examples of applications of Bayesian filtering are:

- Global positioning system (GPS) is a satellite navigation system. The GPS receiver computes its position based on signals from several GPS satellites.

- Target tracking uses multiple sensors to determine the position and velocity of a target.
- Stochastic optimal control uses Bayesian filtering to reconstruct the state of the time-varying stochastic system through a set of sensors. These controllers are usually found in cars and airplanes.

Before describing the theory of Bayesian filtering, variables need to be presented.

$$x_t = \text{the state at time } t \text{ (unknown)} \tag{2.1}$$
$$z_t = \text{the observation at time } t \text{ (known)} \tag{2.2}$$
$$z_{1:t} = \{z_1, z_2, \ldots, z_t\} \tag{2.3}$$

When Bayesian filtering is used for localization, the state $x_t$ refers to the position of the observations at the time *t*, while $z_t$ is the measurements or observations, for example data from image processing.

The aim of Bayesian filtering is to calculate the best estimate of the state $x_t$. This is done by computing the statistical distribution of the state $x_t$ given the previous measurements $z_{1:t-1}$, which is called the *predictive distribution*. Before explaining the predictive distribution in detail, the Bayesian model needs to be presented.

The model for the states and the measurements consist of a *prior model*, a *dynamic model*, and a *measurement model*, which are described as following (Särkkä, 2013):

**The prior model** consists of information before anything happens. The prior distribution, $p(x_0)$, is the likelihood of initial parameter $x_0$.

**The dynamic model,** $p(x_t|x_{t-1})$, describes the system dynamic at the time $t$ given the state at the previous time $t-1$.

**The measurement model,** $p(z_t|x_t)$, describes the relationship of the true state and the measurements. $p(z_t|x_t)$ is the distribution of observation $z_t$ given state $x_t$.

Thus, a general probabilistic state space model can be rewritten in the following form.

$$x_0 \sim p(x_0) \tag{2.4}$$
$$x_t \sim p(x_t|x_{t-1}) \tag{2.5}$$
$$z_t \sim p(z_t|x_t) \tag{2.6}$$

The state $x_t$ is calculated from a possibly nonlinear function of the previous state $x_{t-1}$, while the measurements $z_t$ are based on a possibly nonlinear function of the state $z_t$. The objective is to recursively estimate the state $x_t$ from the measurements $z_t$.

$$x_t = f_k(x_{t-1}, u_t) \tag{2.7}$$
$$z_t = h_k(x_t) \tag{2.8}$$

In this project, the control inputs are of no relevance, and the theory to be presented will, therefore, not include the control inputs $u_t$. The state $x_t$ is assumed to be *complete*, which

means that the state $x_t$ is the best predictor of the future and no additional information about measurement or previous states is necessary. In other words, the state $x_t$ contains information of all the previous states and measurements. Processes with this assumption are called *Markov Chains*. The graphical model of the Markov Chain is illustrated in Figure 2.1.

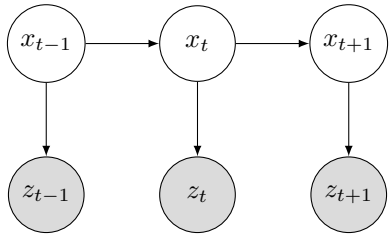


**Figure 2.1:** Graphical model of the Markov Chain with the unknown (white) and observed (grey) variables.

Other relevant probabilistic distributions are *posterior* distribution, *predictive* distribution and *predictive posterior* distribution (Särkkä, 2013):

**The posterior distribution,** $p(x_t|z_{1:t})$, is the distribution of the parameters $x_t$ given the observations $z_{1:t}$, after the observations $z_{1:t}$ has been measured. The posterior distribution is often called the *belief* of the state *x* at time *t*, $bel(x_t)$.

$$bel(x_t) := p(x_t|z_{1:t}) \tag{2.9}$$

**The predictive distribution,** $p(x_t|z_{1:t-1})$, describes the estimation of the predicted state at time *t* given the measurements at the previous time $t-1$. The predictive distribution have the following notation:

$$\overline{bel}(x_t) := p(x_t|z_{1:t-1}) \tag{2.10}$$

**The predictive posterior distribution,** $p(z_t|z_{1:t-1})$, is the distribution of the new measurements $z_t$ given the observed measurements $z_{1:t-1}$. This distribution is used to compute the posterior distribution. The predictive posterior distribution can also be replaced by a normalizations factor, $\eta$ , to make sure $bel(x_t)$ have the right probability distribution, i.e. integrates to 1.

Bayesian filtering uses a recursive filtering approach, which means data is processed sequentially rather than using the complete data set. This is done to prevent saving the whole data set and to avoid processing data multiple times for every new measurements (Arulampalam et al., 2002). The recursive calculations for the Bayesian filtering are obtained in the following steps (Särkkä, 2013):

**Initialization** The recursive computations starts with the prior distribution, $p(x_0)$.

**Prediction step** This step computes the predictive distribution, $p(x_t|z_{1:t-1})$. The predictive distribution for Bayesian filters is computed from the Chapman-Kolmogorov

equation:

$$p(x_t|z_{1:t-1}) = \int p(x_t|x_{t-1})p(x_{t-1}|z_{1:t-1})dx_{t-1} \tag{2.11}$$

With the belief representations the prediction step computes:

$$\overline{bel}(x_t) = \int p(x_t|x_{t-1})bel(x_{t-1})dx_{t-1} \tag{2.12}$$

**Update step** The posterior distribution, $p(x_t|z_{1:t})$, is computed in this step. For Bayesian filters the posterior distribution is computed from the Bayes' rule:

$$p(x_t|z_{1:t}) = \frac{p(z_t|x_t)p(x_t|z_{1:t-1})}{p(z_t|z_{1:t-1})} \tag{2.13}$$

where the predictive posterior distribution is

$$p(z_t|z_{1:t-1}) = \int p(z_t|x_t)p(x_t|z_{1:t-1})dx_t \tag{2.14}$$

With the belief representations the update step computes:

$$bel(x_t) = \eta p(z_t|x_t)\overline{bel}(x_t) \tag{2.15}$$

where $\eta$ is a normalizations factor such that $bel(x_t)$ integrates to 1.

To avoid the numerical problems that occur when the probabilities come close to 0 or 1, the log odds of the belief may be introduced. The log odds have values between -$\infty$ and $\infty$ and is the logarithm of the probability ratio of a positive event divided by the probability of the negative event.

$$l(x) := \log\frac{p(x)}{1-p(x)} \tag{2.16}$$

### 2.1.2 Occupancy Grid Mapping

Before explaining the Occupancy Grid Mapping (OGM) theory, some notational differences from the recursively Bayesian estimation need to be presented.

$$x'_t = \text{the pose of a robot at time } t \text{ (known)} \tag{2.17}$$
$$z_t = \text{the observation at time } t \text{ (known)} \tag{2.18}$$
$$\mathbf{m}_i = \text{the map for each grid cell } i \text{ (unknown)} \tag{2.19}$$

Note, the variables to be estimated is the new variable $\mathbf{m}_i$, which will be defined, and not the position $x_t$ as done in the recursively Bayesian estimation. The pose of the robot is known, and to avoid confusion, the pose is denoted with a mark $x'_t$ and not $x_t$ to emphasize that these two have different usage.

There are many reasons to use and not to use Occupancy Grid Mapping. The main advantage of using Occupancy Grid Mapping for mapping is the simplicity. The algorithm should be easy to understand and implement. One weakness of the method is the computational time. The recursive estimation method uses a lot of time to compute large areas, which can be a problem for online estimation.

Some assumptions for the usage of OGM need to be made. The assumptions are:

- The area or environment is divided into fixed grids.
- The pose of the robot is known.

OGM is an algorithm based on the recursively Bayesian estimation theory and is usually used for mapping of unknown environments, for instance, mapping a room with the use of sensors on a robot. The problem OGM solves is called *mapping with known poses*. OGM generates a gridded map from noisy and uncertain measurement data, under the assumption that the pose of the robot is known (Thrun et al., 2006).

The variable $\mathbf{m}_i$ was introduced in Equation (2.19). A *map* of an environment includes a list of the objects in the environment and their positions. A map *m* is, therefore, a list of object in the environment along with their properties.

$$m = \{m_1, m_2, \ldots, m_N\} \tag{2.20}$$

where *N* is the number of objects in the environment. OGM does not calculate the whole map under one calculation. The map is divided into grid cells, hence the name Occupancy *Grid* Mapping. For this project, the map *m* is thus denoted as

$$m = \{\mathbf{m}_i\} \tag{2.21}$$

where *i* is the grid cell index. Each $\mathbf{m}_i$ have a binary value to describe if the cell is occupied or not. The occupied value is denoted as "1" and the free value is denoted as "0". For instance, when $\mathbf{m}_i = 1$ the grid cell is occupied. An occupied grid cell means the cell has an object in the specific grid area, which for a robot mapping a room would be an obstacle, for instance, a wall.

The graphical model of the problem mapping with known poses is illustrated in Figure 2.2, where the measurements $z_t$ are observed and the pose of the robot $x'_t$ is known. For the OGM algorithm, the control input $u_t$ is irrelevant since the path and pose of the robot are predetermined and known.

The purpose of OGM is to locate obstacles or objects in an environment or area. To estimate the position of the objects, OGM estimates the probability of occupancy for each of the grids in the environment. Hence, the main goal of OGM is to calculate the posterior distribution. The probability distributions for OGM are as following:

**The prior distribution,** $p(\mathbf{m}_i)$ is only dependent on the map $\mathbf{m}_i$.

**The posterior distribution,** $p(\mathbf{m}_i|z_{1:t}, x'_{1:t})$, is the posterior over the map $\mathbf{m}_i$ given the observed measurements $z_t$ when the pose of the robot $x'_t$ is known at time *t*.

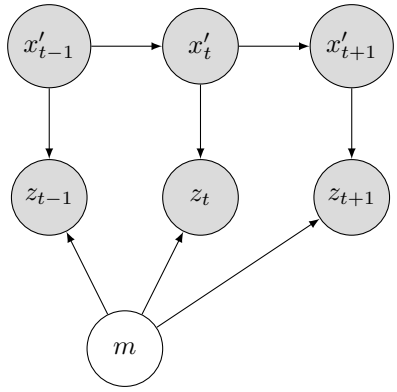


**Figure 2.2:** Graphical model of the mapping with known poses with unknown (white) and observed/known (grey) variables.

Note that because of the different notations, see Equation (2.17)-(2.19), the posterior distribution looks different from the posterior distribution in Section 2.1.1, but it is still based on the Bayesian theory. OGM calculates the posterior distribution for each grid cell. OGM will for each time $t$, base the calculations on one grid, and not the all the grids, which Bayesian filtering does. The grids are thus independent of each other. This will reduce the number of calculations, compared to Bayesian filtering.

Logarithmic representation of the probability distributions will be used for OGM. This is to avoid singularities when the probability is closing 0 or 1. The general calculation of logarithmic representation is showed in Equation (2.16), and for the posterior distribution for OGM, called log odds of occupancy, is:

$$l_{t,i} = \log \frac{p(\mathbf{m}_i | z_{1:t}, x'_{1:t})}{1 - p(\mathbf{m}_i | z_{1:t}, x'_{1:t})} \tag{2.22}$$

where *t* is the time and *i* is the grid cell index. The prior distribution in the logarithmic form, which called the prior of occupancy, is:

$$l_0 = \log \frac{p(\mathbf{m}_i = 1)}{p(\mathbf{m}_i = 0)} = \log \frac{p(\mathbf{m}_i)}{1 - p(\mathbf{m}_i)} \tag{2.23}$$

The log odds can be reconverted back into probabilistic representation:

$$p(\mathbf{m}_i | z_{1:t}, x'_{1:t}) = 1 - \frac{1}{1 + \exp\{l_{t,i}\}} \tag{2.24}$$

The OGM algorithm is presented in Algorithm 2.1. The algorithm will go through each grid cell *i*. If that current grid is a part of the observed grids in the observations $z_t$, the new $l_{t,i}$ is calculated. The new $l_{t,i}$ based on the value of the last time step, $l_{t-1,i}$ and the function INVERSE_SENSOR_MODEL(). The prior of occupancy, $l_0$, is subtracted from the new $l_{t,i}$ to avoid double inclusion of the initial value $l_0$. The value of $l_{t,i}$ will be between

-$\infty$ and $\infty$. Where a positive number indicates a high probability and a negative value indicates a low probability. An example can be a probability value at 0.9 will have the log odds value 2.1972, while 0.1 will have the log odds value -2.1972. In OGM, these values will be represented and plotted to locate the objects. The grey scaled colors represent the occupation of grids, which will be explained more in Chapter 3.

**Algorithm 2.1** The Occupancy Grid Mapping Algorithm

**procedure** OCCUPANCY_GRID_MAPPING($\{l_{t-1,i}\}, x'_t, z_t$)
  $i = 1$
  **for** all cells $\mathbf{m}_i$ **do**
    **if** $\mathbf{m}_i$ in perceptual field of $z_t$ **then**
      $l_{t,i} = l_{t-1,i}$+ INVERSE_ SENSOR_ MODEL$(\mathbf{m}_i, x'_t, z_t) - l_0$
    **else**
      $l_{t,i} = l_{t-1,i}$
    **end if**
  **end for**
  **return** $\{l_{t,i}\}$
**end procedure**

The INVERSE_SENSOR_MODEL() is calculated from the logarithmic representation of the inverse measurement model, $p(\mathbf{m}_i|z_t, x'_t)$:

$$\text{INVERSE\_SENSOR\_MODEL}(\mathbf{m}_i, x'_t, z_t) = \log \frac{p(\mathbf{m}_i|z_t, x'_t)}{1 - p(\mathbf{m}_i|z_t, x'_t)} \tag{2.25}$$

The probability distribution $p(\mathbf{m}_i|z_t, x'_t)$ is called the *inverse* measurement model, because a regular measurement model from the Bayesian theory, $p(z_t|x_t)$, describes the distribution of observations $z_t$ given $x_t$, while $p(\mathbf{m}_i|z_t, x'_t)$ describes the distribution of the map $\mathbf{m}_i$ given the observations $z_t$ and the pose $x'_t$. The measurement model for OGM is, therefore, the inverse of the measurement model for Bayesian filtering, when the new notations are introduced, see Equation (2.17)-(2.19).

The behaviour of the inverse measurement model is that the model has one probability if the grid is occupied, $p_{\text{occ}}$, and one if the grid is empty, $p_{\text{free}}$. In a perfect world, these probabilities would be 1 and 0, respectively, but these number are unrealistic due to uncertainties. The values for these probabilities will be presented in Chapter 3. The probabilities will be converted into log odds. The two log odds of occupancy will be $l_{\text{occ}}$ and $l_{\text{free}}$, respectively. Based on this, the output of INVERSE_SENSOR_MODEL() is either $l_{\text{occ}}$ or $l_{\text{free}}$. To use the INVERSE_SENSOR_MODEL() presented in Equation (2.25), an important assumption need to be made. The assumption is:

- The log odds $l_{\text{occ}}$ and $l_{\text{free}}$ need to be known to use the simplified model of the INVERSE_SENSOR_MODEL() in Equation (2.25).

Equation (2.25) is thus a simplified model of the INVERSE_SENSOR_MODEL() when $l_{\text{occ}}$ and $l_{\text{free}}$ is known and predefined. When $l_{\text{occ}}$ and $l_{\text{free}}$ are not known, complicated calculations need to be completed. These calculations are described in Thrun et al. (2006).

The OGM algorithm can be compared to the calculation steps for Bayesian filtering. The following steps are for the OGM algorithm:

**Initialization** To begin with each grid get the value of the prior of occupancy, $l_0$.

**Prediction step** This step can be compared to the calculation of the INVERSE_SENSOR_MODEL() in line 5 in Algorithm 2.1.

**Update step** This step updates the logarithmic representation of occupancy, $l_{t,i}$, based on the value calculated from INVERSE_SENSOR_MODEL() in the prediction step and the previous value $l_{t-1,i}$, see line 5 in Algorithm 2.1.

## 2.2 Path Planning Algorithm

In this project, a UAV path planning algorithm and framework will be used together with the OGM. The UAV path planner algorithm and framework is developed by the PhD student Anders Albert, and the theory in this section is based on Albert and Imsland (2015).

The goal of the path planner is to find the optimal path between several nodes, i.e. decide an optimal sequence of nodes. Each of these nodes, i.e. icebergs, should have a different uncertainty of the position. The path planning will make the UAV will fly to the iceberg with the highest uncertainty. In other words, the sequence of nodes will be sorted such that the node with the highest uncertainty is first, and the lowest is last. This position uncertainty in Albert and Imsland (2015) is based on the time since the observation.

### 2.2.1 Strategy

The strategy of the path planning is to solve the path planning problem often. When the UAV reaches the first node in the sequence, a new sequence of nodes is calculated, and the first node in the new sequence is given to the UAV. This strategy is based on model predictive control (MPC). MPC uses information about a model to minimize some criteria to calculate a sequence of inputs. The MPC applies the first input of the sequence and then calculates the sequence again.

To use this strategy some assumptions and simplifications are made:

- No UAV dynamics
- Initial estimate of position
- The icebergs move much slower than the UAV.

### 2.2.2 Optimization Formulation

The algorithm is made as a similar problem as the traveling salesman problem (TSP). Cormen et al. (2009) describes TSP as a problem where a salesman must visit *n* cities, and the salesman want to visit each city exactly once and finish in the city he started in. The salesman wants to travel with the lowest cost, and the algorithm chooses the lowest cost to get a minimum-cost tour. For the path planner algorithm, each position the UAV want to visit and the UAV is a node in a graph. If only the distance between the nodes are considered, and not the uncertainty of the nodes, finding the shortest path between these nodes is then a TSP. The path planning algorithm is, therefore, an optimization problem between finding the shortest path between the nodes and reduce the uncertainty of the nodes. The optimization problem was formulated to be in a Mixed Integer Linear Programming (MILP) framework, which solves the optimization problem when only some of the optimization variables are integers.

The two optimization variables are:

- The first is $y_{path}$, which is a $N \times N$ matrix, where $N$ is the number of nodes. The elements in the matrix represent the UAV path. The value is 1 if the node is included and 0 if not. This matrix will be multiplied by the position uncertainty of each node, $\sigma_{nodes}$, such that only the position uncertainty of the nodes that are included will be used in the optimization. A high value of the position uncertainty represents a high desire to visit the node.
- The second is an integer vector $t'(i')$, where $i' = 1 : N$, which contains the nodes in the visiting order. This vector will be multiplied with a $N \times N$ matrix which contains the distances between the nodes.

The optimization problem:

$$\min F(y_{path}, t'(i')) \tag{2.26}$$

This function will find the best path between the nodes based on the minimum distance and minimum of the negative position uncertainty, i.e. the maximum position uncertainty. A tuning variable, weighted between the reduction of position uncertainty and the shortest distance, will be included. If this value is too high, the problem becomes a TSP, and if too low, the order will just depend on the position uncertainty. For the calculations behind this optimization, see Albert and Imsland (2015).

### 2.2.3 Computational Complexity

TSP and MILP are NP-complete. The optimization problem is an MILP problem and therefore also an NP-complete problem. NP-complete problems have an exponential increase in computational time with the size of the problem. Albert and Imsland (2015) shows the computational time for a different number of nodes, and the number of nodes should be kept under 15 to have an acceptable and practical solution time. For a UAV computing the path online, is it important to keep the computational time as low as possible.

Chapter 3

# Occupancy Grid Mapping for Iceberg Localization

The purpose of this project is to make a map that shows the location of icebergs using Occupancy Grid Mapping. This map will be made such that a customer, for instance, a captain on a vessel can use it to avoid icebergs while operating the vessel on the sea. To localize the icebergs, the Occupancy Grid Mapping algorithm will be used, which will make a map based on the probability of iceberg location. The probability map will be analysed and converted into a functional map to give to a customer.

## 3.1 Implementation

As described in Section 2.1.2, Occupancy Grid Mapping (OGM) is usually used for robots with sensors to map a room. For this project, OGM will be used to locate icebergs from a UAV. The difference is, the UAV will use images straight above the area to locate icebergs, while, on the other hand, the robot would use sensors in front of it to measure the distance to the obstacle.

### 3.1.1 Cases of Flight Path

In this section, different cases of flight paths will be discussed. Later in this project, OGM will be implemented with a path planner algorithm, and the flight paths from this chapter will then not be used. For icebergs localization, a UAV will be simulated to be flying over a given area. The UAV will be given different flight paths to give the reader a better understanding of the usage of the OGM algorithm.

**Case 1** To begin with the UAV have no information about the area and where the icebergs can be located. The UAV can, therefore, fly over the whole area and search for icebergs. As known the OGM algorithm consists of grids. To make the UAV search the whole area, the UAV will be flying back and forth each row starting at a corner, see Figure 3.1 for illustration. This way the whole area will be guaranteed covered, and the estimation of iceberg position may, in theory, be a good estimation of where the icebergs are. The UAV will take images that will cover multiple grids. With this flight path and image with multiple grids, the OGM algorithm will do calculations to each grid multiple times. This may give an accurate estimation of the probability of the location of icebergs.

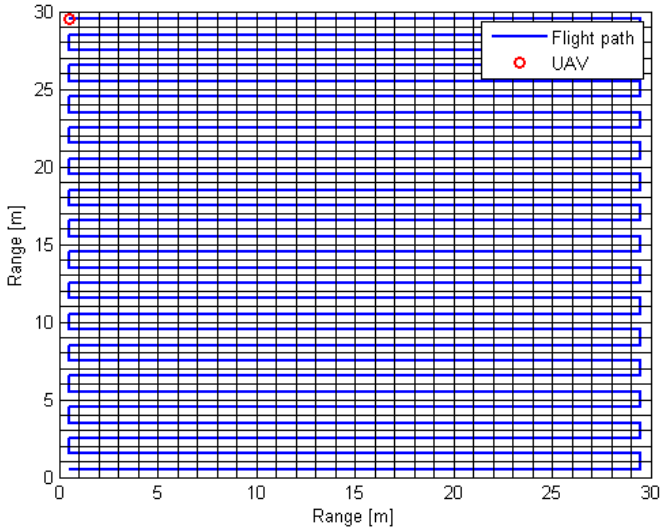


**Figure 3.1:** Illustration of flight path for Case 1.

In theory, the UAV path that covers the whole area with a back and forth method is an acceptable path, but in real life, this path may use too much time to fly back and forth. A more realistic path for the UAV can be to decide a path that will cover the areas where we have information that icebergs can be located. This information can, for example, be from a satellite picture or other information.

Different types of flight paths can for example be:

**Case 2** In a 30x30 meter area, a satellite picture gives the information that icebergs can be located in the north-east corner of the area. The UAV is located at the south-west corner. The UAV can fly diagonally half way up to the north-east corner and then begin to fly back and forth in the north-east area. For illustration see Figure 3.2. The UAV can then fly and investigate the other parts of the area.

**Case 3** From another satellite picture, the information can be that there is probably some icebergs on the diagonal and each side of the diagonal. The UAV can start by flying

on the diagonal and continue turning and fly downwards in the area. The UAV will, with this flight path, try to cover the area given the information from the satellite. For illustration of this flight path, see Figure 3.2. The UAV can continue to cover the parts that still needs to be investigated to get information about the whole area.

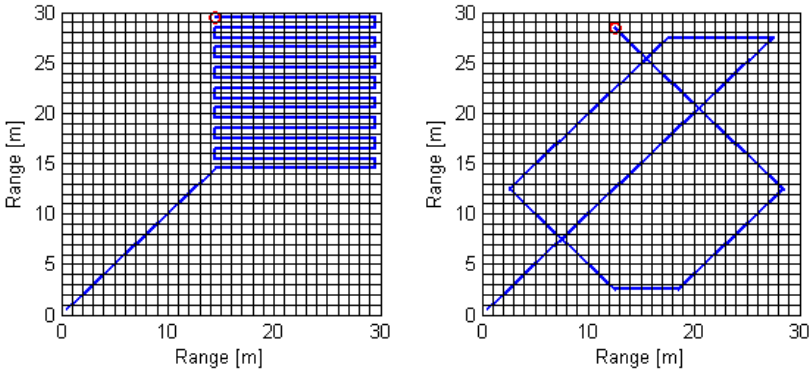


**Figure 3.2:** Illustration of flight path for Case 2 and Case 3.

### 3.1.2 Occupancy Grid Mapping Implementation

In Section 2.1.2, the theory of Occupancy Grid Mapping (OGM) was presented. The focus of the theory was on how OGM works for a robot mapping a room. In this project will the OGM theory be used for mapping an area from the air. There will be implemented some modifications to the theory in Thrun et al. (2006), to make the OGM fit this usage.

The OGM algorithm will be implemented as described in Section 2.1.2, see Algorithm 2.1. The inverse sensor model algorithm needs to be modified. Thrun et al. (2006) presents an inverse sensor model algorithm for robots with range finders. This algorithm uses the range to the center-of-mass of the grid cell $\mathbf{m}_i$ based the pose of the robot, to determine the if the cell is a part of $z_t$ and if the grid is occupied. In this project, the inverse sensor model will be implemented as Algorithm 3.1.

**Algorithm 3.1** The inverse sensor model algorithm

```
1: procedure INVERSE_SENSOR_MODEL(m_i, x'_t, z_t)
2:     if m_i is not a part of z_t then
3:         return {l_0}
4:     else if m_i is a part of z_t and occupied then
5:         return {l_occ}
6:     else
7:         return {l_free}
8:     end if
9: end procedure
```

As described in Section 2.1.2, the output value of the inverse sensor model is either $l_0$, $l_{\text{occ}}$ and $l_{\text{free}}$, which will be calculated the following way:

$$l_0 = \log \frac{p(\mathbf{m}_i)}{1 - p(\mathbf{m}_i)} \tag{3.1}$$

$$l_{\text{occ}} = \log \frac{p_{\text{occ}}}{1 - p_{\text{occ}}} \tag{3.2}$$

$$l_{\text{free}} = \log \frac{p_{\text{free}}}{1 - p_{\text{free}}} \tag{3.3}$$

When choosing these probabilities, the rule

$$l_{\text{free}} < l_0 < l_{\text{occ}} \tag{3.4}$$

is important for the OGM algorithm to work. The values for the probabilities will be presented in Section 3.2 together with the other simulation values.

In this project, the observed measurements $z_t$ will be a simulated aerial image taken from the UAV. This image will be divided into grids. The grid size and size of $z_t$ will be presented in Section 3.2. The grids in the image $z_t$ will either be occupied or not. The inverse sensor model in Algorithm 3.1 will, therefore, return either $l_{\text{occ}}$ or $l_{\text{free}}$ to the occupancy grid mapping algorithm, see Algorithm 2.1 and 3.1. If the grid $\mathbf{m}_i$ is not included in the image data $z_t$, $l_0$ will be returned to the occupancy grid mapping algorithm in Algorithm 2.1.

### 3.1.3 Probability Analysis

The finished map to be given to the customer will be made based on an analysis of the probabilities of occupancy in the estimated map. The estimated maps from OGM will contain a scale of log odds from -$\infty$ to $\infty$, i.e. the probability of the grid being occupied is from 0% to 100%. These probabilities will be displaced in different gray tones from black to white. As a proposal, the final map may only contain three colors to represent occupied, unoccupied and not examined grids, receptively white, dark blue and light blue. This way the customer can easily see where the icebergs are located and use this information to avoid contact.

The analysis of the estimated map will happen after the simulation, due to the variations of the probabilities for each simulation. A reference point is making grids with high probabilities be white and with low probabilities be dark blue. The challenge is to find where to divide the probabilities, or gray tones, in the middle of the scale. The main question is, how high do the probability have to be for the grid to be considered as occupied.

## 3.2 Simulation and Results

In this chapter, the OGM is tested for iceberg localization. The simulations are simulated for three different cases to make the functioning of OGM for iceberg localization understandable. Each case is simulated two times, where one includes measurement errors and one do not.

### 3.2.1 Simulation Variables

The variables presented here applies to all the cases. The area has the size of $30 \times 30$ meter. The grids have the size $1 \times 1$ meter, and the area is, therefore, containing $30 \times 30$ grids. The icebergs are predefined because this is simulations and not real experiments. The area contains five icebergs, and each iceberg is spread out over nine grids. The measurements, i.e. the image taken from the UAV, have the size $4 \times 4$ meter and therefore also $4 \times 4$ grids. For the simulations in Section 3.2, the UAV is observing one grid per second. This will be changed in Section 4.2 when more realistic simulations are completed.

The initial probability is chosen as the number of occupied icebergs divided by the number of grids:

$$p(\mathbf{m}_i) = \frac{5 \cdot 9}{30 \cdot 30} = 0.05 \qquad l_0 = -2.9444 \tag{3.5}$$

The initial value is just an indication on how high probability there is to find iceberg at each grid. If this would be done for real experiments, this value should be based on known information about the area, for instance, satellite pictures or based on experiences. The remaining probabilities, $p_{\text{occ}}$ and $p_{\text{free}}$, are given by the values:

$$p_{\text{occ}} = 0.9 \qquad l_{\text{occ}} = 2.1972 \tag{3.6}$$

$$p_{\text{free}} = 0.01 \qquad l_{\text{free}} = -4.5951 \tag{3.7}$$

When a grid is observed as occupied, there is a high probability of that grid being occupied. The opposite applies when the grid is observed as empty. There is a low probability for the empty grid being occupied. In a perfect world these would be 1 and 0, respectively. The reason for the deviation is to take calculation errors and other factors into consideration. To check if the chosen probabilities can be used, the Equation (3.4) need to be fulfilled, which it is for this case.

### 3.2.2 Measurement Error

As mentioned earlier, the OGM is using UAV images for the probability estimation of the iceberg location. This project only consists of simulations and not experiments from real data and image processing. Image processing and image interpretation is not always perfect. In this project, the simulations include errors to imitate image processing errors and other errors regarding the observations. The simulations, therefore, include errors

that will be referred to as *measurement errors*. These measurement errors are divided in two, where each measurement error depends on the occupation of the grids. The two measurement errors are denoted as:

**$\epsilon_{\text{occ}}$** This value represents the percentage of occupied grids which are observed wrong. For instance, if $\epsilon_{\text{occ}} = 0.1$, then 10% of the occupied grids will be observed as empty.

**$\epsilon_{\text{free}}$** The same applies to this value, which represents the percentage of free grids that are observed wrong. For instance, if $\epsilon_{\text{free}} = 0.1$, then 10% of the free grids will be observed as occupied.

Each case is simulated with and without the measurement errors, $\epsilon_{\text{occ}}$ and $\epsilon_{\text{free}}$. The values of the measurement errors for all the cases are:

$$\epsilon_{\text{occ}} = 0.1 \tag{3.8}$$

$$\epsilon_{\text{free}} = 0.01 \tag{3.9}$$

This way 10% of the occupied and 1% of the free grids are observed wrong, and the results may be more realistic. These measurement errors can be compared to the probabilities of occupancy $p_{\text{occ}}$ and $p_{\text{free}}$. Both the measurement errors and the probability of occupancy represents errors. While the measurement errors are included in the simulation to illustrate simulation and measurement errors, the probability of occupancy have an error which represents calculation errors. The probabilities of occupancy are given in Equation (3.6)-(3.7), and are 0.9 and 0.01. The probability of occupancy error is therefore:

$$\epsilon_{p_{\text{occ}}} = 0.1 \tag{3.10}$$

$$\epsilon_{p_{\text{free}}} = 0.01 \tag{3.11}$$

Both the measurement errors and the probability of occupancy errors are errors representing occupied and empty grids. The two types of errors are thus given the same value, see Equation (3.8) and (3.10) and Equation (3.9) and (3.11).

### 3.2.3 Case 1: The Whole Area

In this chapter, the results show how the OGM is acting when the UAV is flying and covering the whole area. The UAV is flying in a back and forth shape as described in Section 3.1.1. The first simulation for this case is not including the measurement errors $\epsilon_{\text{occ}}$ and $\epsilon_{\text{free}}$, while the second simulation includes these. This is done to see the difference in how the measurement errors affects the results of OGM.

Case 1 is simulated with no added measurement errors, $\epsilon_{\text{occ}}$ and $\epsilon_{\text{free}}$, and is shown in Figure 3.3. The plots show how the area is covered by the UAV. Dark areas are areas the UAV found empty, while the white are where the UAV found grids occupied by icebergs. The red circle represents the position of the UAV at the current time when the map is plotted. The grey areas represents the unexplored grids with the probability of occupancy $p(\mathbf{m}_i)$, see Equation (3.5). The output of OGM, presented in Figure 3.3, clearly show where the icebergs are located. The negative about this approach, it the simulation time.

This simulation uses one second per grid, and the total simulation time is therefore 900 seconds. For a UAV, the length of the simulation time is crucial, due to the time and range restrictions of the UAV flight.

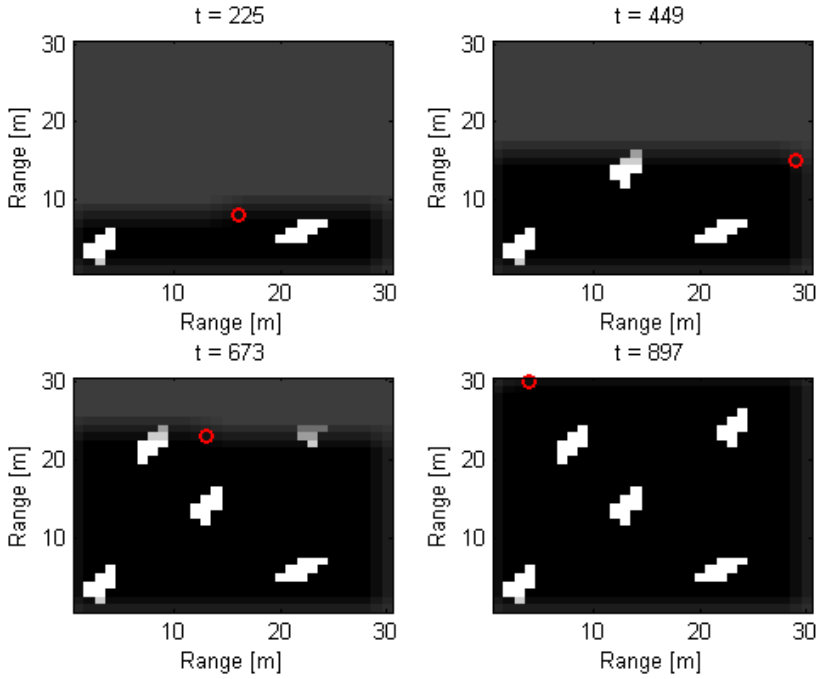


**Figure 3.3:** Case 1 without measurement errors. Estimated location of icebergs with flight path covering the whole area in a back and forth shape. The red circle represents the position of the UAV at the current time $t$. The white grids are occupied, while the black are empty. The grey grid represents unexplored grids.

The measurement error $\epsilon_{\text{occ}}$ and $\epsilon_{\text{free}}$ are also added to the simulations of Case 1, see Figure 3.4. The dark areas still represent free grids, white represents icebergs, i.e. occupied grids, and the grey grids are unexplored. For this simulation, the measurement errors have affected the outcome. Some of the grids inside the white icebergs have different light grey colors. As known, each grid is observed multiple times because of overlapping images of the area. 10% of these times the occupied grids are observed wrong, and the different light grey colors appear. The color is light grey, because the probability is still close to 100%. The same applies for the empty grids, where 1% is observed wrong. In the darkest area, some of the grids have a lighter grey color than the darkest grey color. The errors thus have an impact, but because of the frequently overlapping of images, the effect is relatively small and negligible.

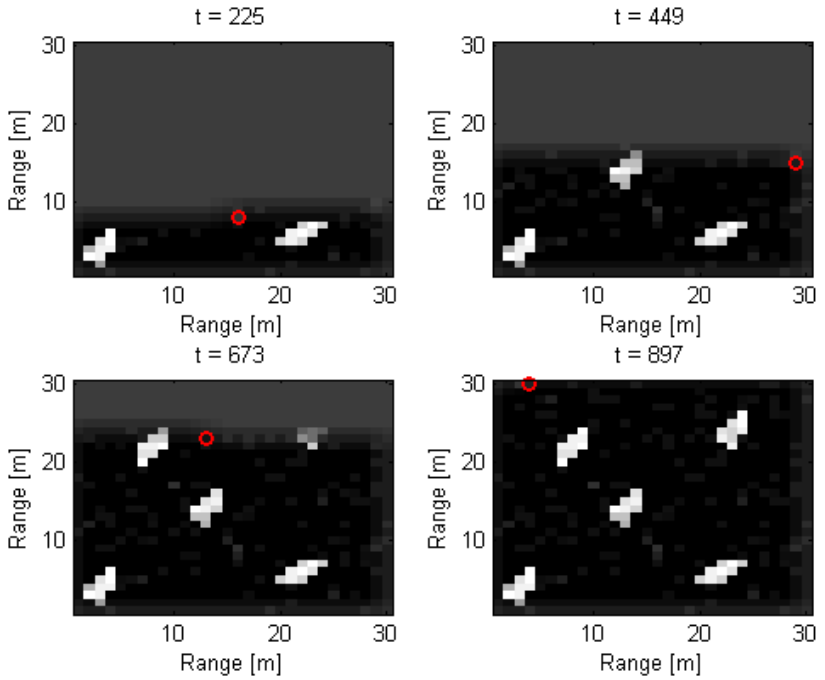


**Figure 3.4:** Case 1 with measurement errors. Estimated location of icebergs with flight path covering the whole area in a back and forth shape. The red circle represents the position of the UAV at the current time $t$. The white and light grey grids are occupied, while the black and dark grey are empty. The grey grid represents unexplored grids.

### 3.2.4 Case 2: Corner of the Area

This case is an example of usage when information about the area, from for instance satellite images, is available. Here, the UAV is flying and covering the corner of the area, due to information about icebergs being on the diagonal and in the corner. The UAV is starting in position (0,0), flying on the diagonal and then back and forth in the corner of the area. The simulation of this case without the measurement errors $\epsilon_{\text{occ}}$ and $\epsilon_{\text{free}}$ is shown in Figure 3.5. For this case total number time steps is 270, which is 270 seconds when the UAV fly over one grid per second. This simulation time is better than using 900 seconds for covering the whole area, see Section 3.2.3. The plots in Figure 3.5 clearly show the outcome of OGM. The OGM algorithm computes the probability of occupancy based on the observations. In this simulation, the iceberg in the corner is lighter than the others. This iceberg has a higher probability of occupancy than the other two icebergs, due to the fact that the probability of this iceberg have been calculated from overlapping more images than the other icebergs. This is because when the UAV is flying on the diagonal the images overlap less than when the UAV is flying back and forth. Despite this difference, the location of the icebergs are evident.

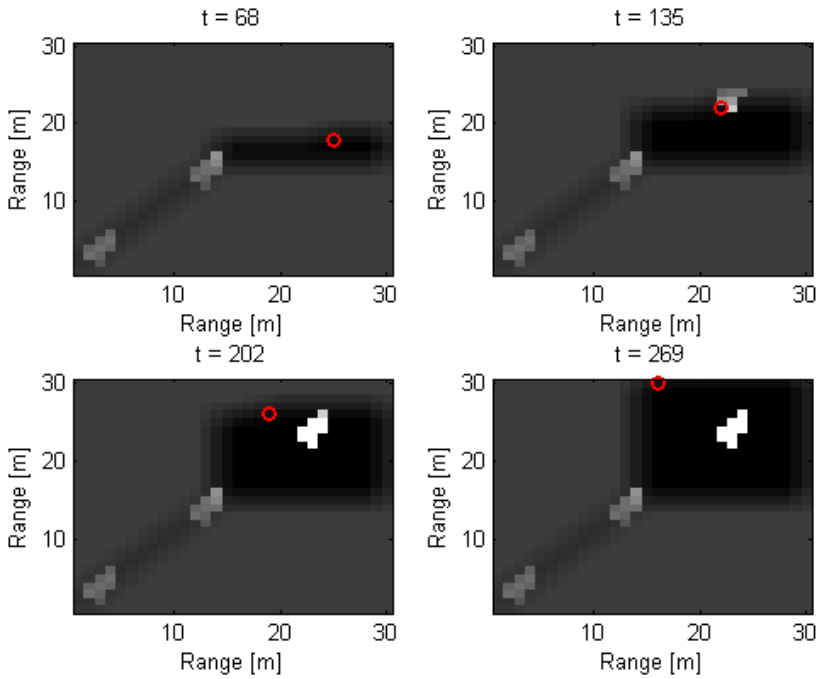


**Figure 3.5:** Case 2 without measurement errors. Estimated location of icebergs with flight path covering a corner of the area. The red circle represents the position of the UAV at the current time $t$. The white and light grey grids are occupied, while the black and dark grey are empty. The grey grid represents unexplored grids.

The result from the simulation of Case 2 with the measurement errors $\epsilon_{\text{occ}}$ and $\epsilon_{\text{free}}$, see Figure 3.6, also clearly showing where the icebergs are located. Some grids have different grey tones, and the measurement errors will therefore also for this method have an impact on the probability of occupancy for the grids. However, the goal of OGM is to locate the icebergs, and by looking at Figure 3.6, this is done. In Section 3.2.6, the probabilities of occupancy will for this simulation be discussed.

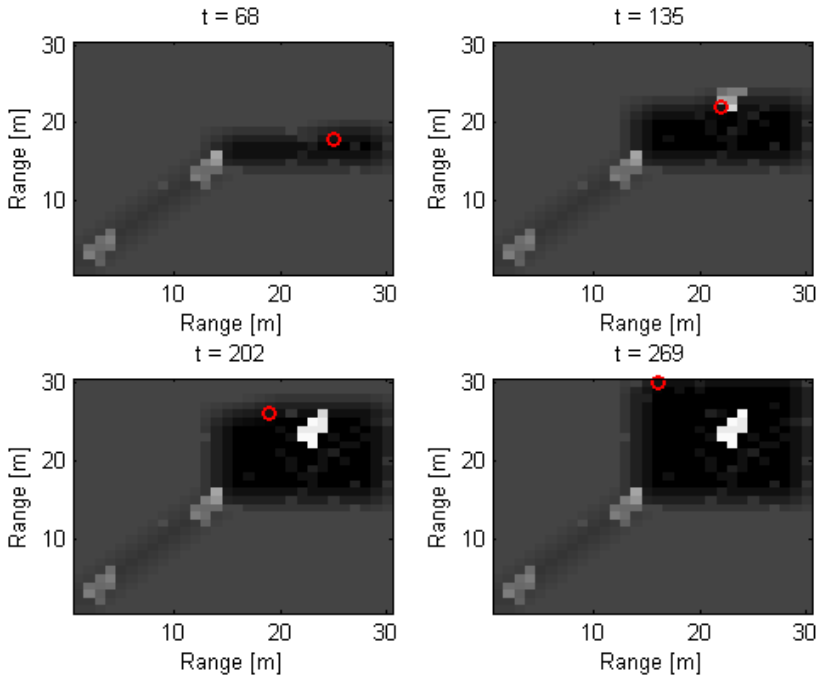


**Figure 3.6:** Case 2 with measurement errors. Estimated location of icebergs with flight path covering a corner of the area. The red circle represents the position of the UAV at the current time $t$. The white and light grey grids are occupied, while the black and dark grey are empty. The grey grid represents unexplored grids.

### 3.2.5 Case 3: Parts of the Area

For this case, information shows there may be some icebergs located in the diagonal and on the sides of the area. This simulation is done to show the reader the results when the UAV is not flying in a back and forth shape.

The results from this case simulated without the measurement errors $\epsilon_{\text{occ}}$ and $\epsilon_{\text{free}}$ are presented in Figure 3.7. The total number of time steps used for this simulation is 95, which is 95 seconds when the UAV uses one second per grid. The positive about this case is the short simulation time used to cover a big area, while the negative is not all

grids are explored. This method, therefore, can be used together with a path planner and information about iceberg positions, which will be done in Chapter 4. As Figure 3.7 shows, the icebergs are located by the OGM algorithm.

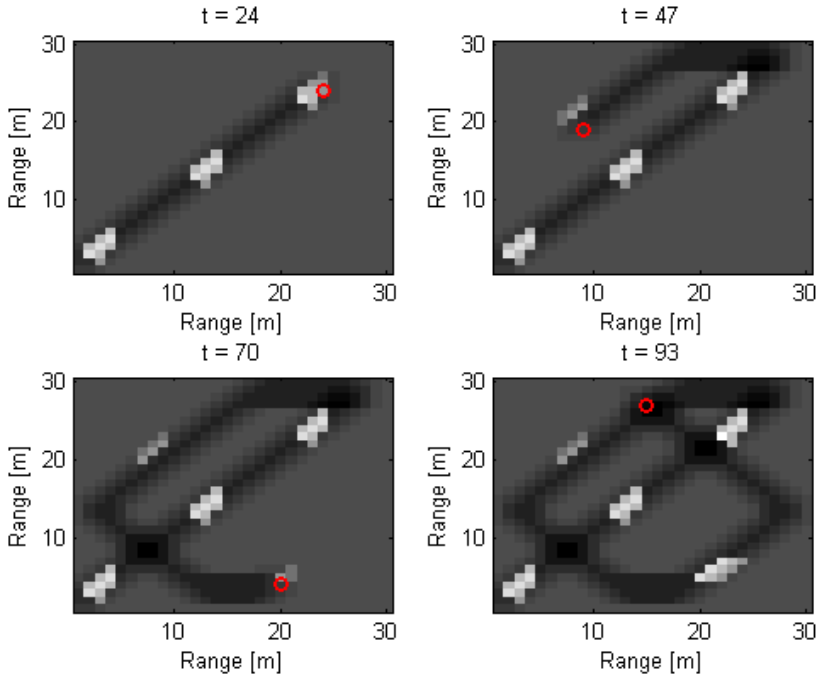


**Figure 3.7:** Case 3 without measurement errors. Estimated location of icebergs with flight path covering parts of the area. The red circle represents the position of the UAV at the current time $t$. The white and light grey grids are occupied, while the black and dark grey are empty. The grey grid represents unexplored grids.

Figure 3.8 show the results from Case 3 simulated with the measurement errors $\epsilon_{\text{occ}}$ and $\epsilon_{\text{free}}$. In this case, the grids are not investigated as many times as in the other cases. For this reason, this simulation are more affected by the measurement errors. This is shown by the single light grey grids in the dark grey path in Figure 3.8. Some of these single grids have almost the same probability as some grids which represents icebergs, which will be discussed more when analysing the probabilities in Section 4.1.

Note, the contrast between the black, grey and white grids is more noticeable in this case than in Case 1 and Case 2. The map is plotted such that the highest probability is given the color white. For this case, the highest probability is lower than for the other cases. Both Case 1 and Case 2 use more overlapping images to calculate the probabilities of occupancy, which increases the probability each time the grid is found occupied. Due to the use of log odds can 100% be represented in log odds values up to $\infty$, which mean the probability continues to increase for each calculation. Even though white grids represent

the probability of approximately 100%, the log odds can have different values. The log odds value of the white grids in Case 1 and Case 2 is, therefore, higher than for Case 3. Note, the same reasoning applies to the black grids. This is the reason for the difference in color contrast between Case 3 and the other cases.

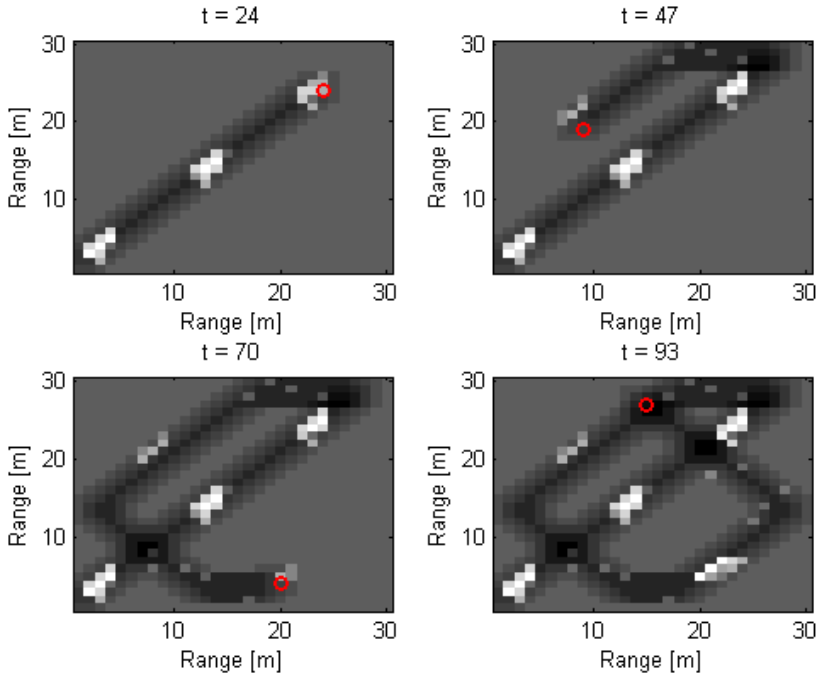


**Figure 3.8:** Case 3 with measurement errors. Estimated location of icebergs with flight path covering parts of the area. The red circle represents the position of the UAV at the current time $t$. The white and light grey grids are occupied, while the black and dark grey are empty. The grey grid represents unexplored grids.

### 3.2.6 Probability Analysis for Map Building

The maps showing the estimated location of icebergs from Case 1-3 with measurement errors are analysed in this chapter. The measurement errors $\epsilon_{occ}$ and $\epsilon_{free}$ represent errors connected to the measurement, for instance, image processing errors, see Section 3.2.2, and is, therefore, more realistic than simulations of the cases without the errors. In the maps showing the estimated location of icebergs with measurement errors, are affected by the measurement errors, and have different grey tones in the map, as discussed in Case 1-3. A new map can be produced to make, for instance, a customer uses the map to navigate in the area. The new maps will be presented in a blue scale, instead of the grey scale to show that this is a different kind of map. In other words, the blue scaled map is not a map produced by the OGM.

### Case 1: The Whole Area

The simulation results of Case 1, where the UAV is searching the whole area in a back and forth shape, are presented in Section 3.2.3. As discussed, the simulation with the added measurement errors, $\epsilon_{occ}$ and $\epsilon_{free}$, clearly show where the icebergs are located. The measurement errors $\epsilon_{occ}$ and $\epsilon_{free}$ are explained in Section 3.2.2. When the UAV is flying over every single grid, the calculations are very accurate. The analyse for this simulation is, therefore, easy. The probability of occupancy for each grid in Figure 3.4 decides the limits for building the new map. In Table 3.1, the maximum and minimum of the probability of occupancy and log odds for the light and dark grids in Figure 3.4 are presented.

| | | Log odds | Probability of occupancy |
|---|---|---|---|
| Light grids | Max | 79.32 | $\approx 100\%$ |
| | Min | 38.38 | $\approx 100\%$ |
| Dark grids | Max | -8.98 | 0.0126% |
| | Min | -29.36 | $\approx 0\%$ |

**Table 3.1:** Log odds and probability of occupancy for grids in Case 1 with measurement errors.

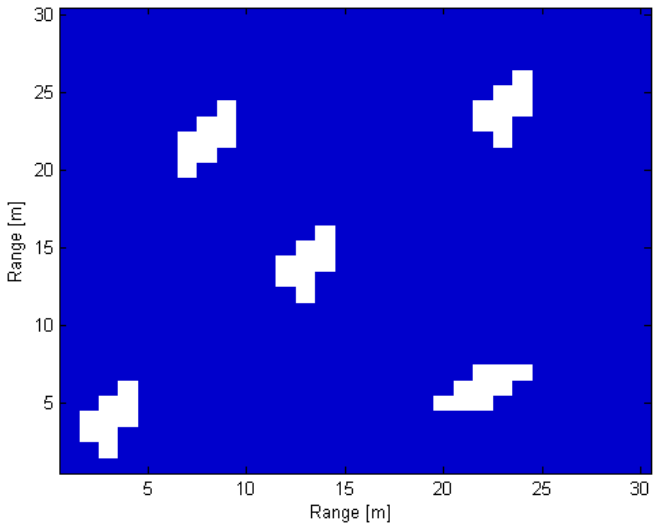


**Figure 3.9:** New map based on the analysis of the log odds and the probability of occupancy for grids in Case 1 with measurement errors. The dark blue area represents the free area, and the white are icebergs.

The lightest grid of the dark grids in Figure 3.4 represents the grid with the highest probability of the grids with the lowest. For Case 1, the log odds for this grid is -8.98, and the probability of occupancy is 0.0126%. This means, there is approximately 0% change for icebergs occupying the dark grids for this case. The grids with log odds -8.98 and under

can be registered as empty in the new map and get the color dark blue. The darkest grid for the lighter grids in Figure 3.4, i.e. the minimum log odds of the grids with the highest log odds, have the value 38.38, which is approximately 1. For all the light grids is the probability of occupancy 100%. In the new map, the light grids is, therefore, registered as occupied with the color white. In case there are some grids with log odds in between -8.98 and 38.38, the limit is sat to be log odds -1. All grids with log odds under -1 are presented as free and over are presented as occupied. This is to avoid false negatives, which is assuming free grids when occupied. For this case, the new map is not including light blue grids, because the UAV is covering the whole area. The new map showing location of icebergs is showed in Figure 3.9.

### Case 2: Corner of the Area

The results of the simulation of Case 2, when the UAV is only searching over the corner and on the diagonal, show that the probability of occupancy is more varying than when the UAV is only flying back and forth. The reason for this is the UAV do not calculate each grid an equal number of times when on the diagonal as for the back and forth approach. The analysis includes, therefore, different probabilities, and not only close to 0 and 1, as for the last case. In Table 3.2, the maximum and minimum of the probabilities of occupancy and the log odds for the grids in Figure 3.6, are presented.

| | | Log odds | Probability of occupancy |
|---|---|---|---|
| Light grids | Max | 65.74 | $\approx 100\%$ |
| | Min | 0.546 | 63.3% |
| Dark grids | Max | -2.755 | 5.98% |
| | Min | -29.36 | $\approx 0\%$ |
| Unexplored | | -2.944 | 5% |

**Table 3.2:** Log odds and probability of occupancy for grids in Case 2 with measurement errors.

The maximum probability of occupancy for the dark grids is approximately 6%, and the minimum for the light grids is 63%. It is, therefore, safe to say that grids with probability of occupancy between 63-100% are occupied with icebergs, and 0-6% means free grids. Also, for this case, the limit is sat to be log odds -1, in case there are grids with probability of occupancy between the log odds -2.755 and 0.546. Grids with log odds under -1 are presented as free with dark blue color, and over -1 are occupied with the color white. There is one exception, and that is the unexplored grids. The initial probability $p(\mathbf{m}_i)$, in this project is 5% for the unexplored grids, which is log odds -2.944. The grids with log odds -2.944 are, therefore, presented as unexplored in the new map with the color light blue. The new map based on this analysis is shown in Figure 3.10.

As shown in Figure 3.10 there is one single grid which have the color white, which is grid (9,12). The reason for this single white grid is that the grids at the edges of the diagonal observations are observed less times than in the grids in corner of the area. If the measurement error $\epsilon_{\text{free}}$ is activated for the empty grids at the diagonal, the grids get a high probability of occupancy, even though the grid is empty. This grid have the probability of

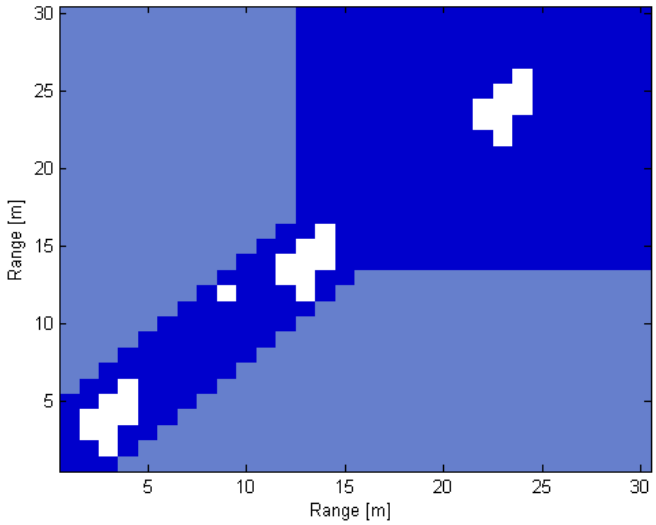


**Figure 3.10:** New map based on the analysis of the log odds and the probability of occupancy for grids in Case 2 with measurement errors. The dark blue area represents the free area, the light blue is the unexplored area and the white are icebergs.

occupancy of 63%, and is accepted as occupied. Some grids will thus be observed wrong, but it is better to assume a grid to be occupied when it is not, rather than the opposite. This can only be check if the UAV fly over the grid again and do new observations. The probability occupancy can then be decreased when the lower and right probability of occupancy for that current grid is added to the current probability of occupancy. This will actually be demonstrated when simulating the OGM with the path planner in Chapter 4. The estimation in the top-right corner is quite accurate, due to the back and forth flying of the UAV, which can be compared to the last analysis in Section 3.2.6.

### Case 3: Parts of the Area

The last map created is based on the simulation results for Case 3 when the UAV is flying to the icebergs in a predefined path. In Section 3.2.5, the effect of the measurement errors $\epsilon_{\text{occ}}$ and $\epsilon_{\text{free}}$ were discussed. When the UAV is flying such that only parts of the area are covered, the images are not covering the grids as good as when flying back and forth. The measurement errors are then affecting the output of the OGM. The log odds and probability of occupancy for the grids in Figure 3.8, are given by Table 3.3.

The minimum probability of occupancy of the light grids is 63%. The grids with the probability of occupancy between 63-100% are, therefore, presented as occupied in the new map. The unexplored grids, with the probability of occupancy of 5%, get the color light blue. The lightest of the dark grids have the probability of occupancy at 24%, which means

| | | Log odds | Probability of occupancy |
|---|---|---|---|
| Light grids | Max | 17.62 | $\approx 100\%$ |
| | Min | 0.5465 | 63.3% |
| Dark grids | Max | -1.104 | 24% |
| | Min | -14.5 | $\approx 0\%$ |
| Unexplored | | -2.944 | 5% |

**Table 3.3:** Log odds and probability of occupancy for grids in Case 3 with measurement errors.

some grids have 24% probability of being occupied by icebergs. 24% is low compared to, for instance, 63%, but 24% may not be low enough to assume the grids with this probability of occupancy to be empty and free. The limit for the other cases has been log odds -1, which is 27%. If this limit is used, the grids with 24% probability of occupancy will be known as empty. The map created in Figure 3.11 is based on this limit. In other words, all grids with log odds under -1 are presented as empty.

In Figure 3.11, there are several single grids with the wrong color. The reason for these errors is the same as for Case 2. The grids on the edges are observed fewer times than grids in the middle of the path, and the measurement errors $\epsilon_{\text{occ}}$ and $\epsilon_{\text{free}}$ thus have a big influence. The single white grids are affected by the measurement error $\epsilon_{\text{free}}$, while the grids (4,4), (8,22) and (23,23), which are parts of icebergs, are affected by the measurement error $\epsilon_{\text{occ}}$.

When the OGM calculations are based on few images and measurements, the measurement errors $\epsilon_{\text{occ}}$ and $\epsilon_{\text{free}}$ have a bigger effect on the probability of occupancy, compared to a case where the OGM would use multiple overlapping images. In Chapter 4, the OGM will be tested with a path planner program, and the UAV flight path will be more realistic than in this chapter. The results in Chapter 4 will discuss the image taking frequency and the effect of measurement errors more.

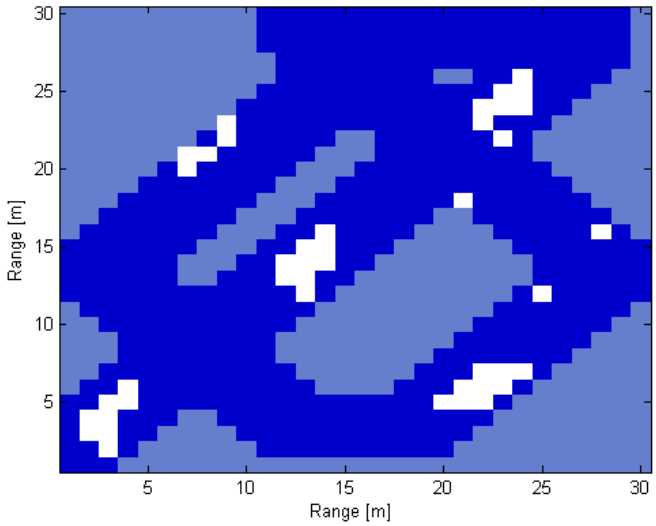


**Figure 3.11:** New map based on the analysis of the log odds and the probability of occupancy for grids in Case 3 with measurement errors. The dark blue area represents the free area, the light blue is the unexplored area and the white are icebergs.

Chapter 4

# Occupancy Grid Mapping with UAV Path Planning

In this chapter, the Occupancy Grid Mapping Algorithm, which was implemented and tested for iceberg localization in Chapter 3, will be tested together with the UAV path planning algorithm from Albert and Imsland (2015). The purpose of implementing the Occupancy Grid Mapping (OGM) algorithm with the UAV path planning, is to show various ways OGM can be used with other algorithms and programs. The UAV will also fly in a more realistic path compared to the simulations of OGM in Chapter 3. This way the reader may get a clearer view of how the OGM algorithm can be used for experiments. The path planner algorithm and framework used in this chapter is developed by the PhD student Anders Albert.

## 4.1 Implementation

In Chapter 3, the OGM was used to find the location of icebergs in a given area. Some simulations were based on some knowledge about the area and satellite pictures to know where to fly and search for the icebergs. When the icebergs were located, a map was generated to show the positions of the icebergs at the time when the UAV flew over the area. Icebergs moves over time due to current and drift, and the map, i.e. the positions of icebergs, need to be updated. The OGM will be used together with the UAV path planning program to update the position of the iceberg. The different cases for simulation of this task will be presented in Section 4.1.1.

As a reminder from Section 2.2, the optimal path is calculated based on two values:

1. The best path between the given positions the UAV should visit, which is represented as $t'(i')$ in Section 2.2.

2. The position uncertainty value, $\sigma_{nodes}$, which is can be calculated based on the probability of the node and time since the last visit.

The value of the best path, i.e. item no.1, will be calculated by the path planning algorithm. The implementation of the UAV path planning framework will be presented in Section 4.1.2. The position uncertainty value, $\sigma_{nodes}$, i.e. item no.2, will be calculated based on the output of OGM. The proposed calculations of $\sigma_{nodes}$ is explained in Section 4.1.3.

### 4.1.1 Cases of Usage

The UAV can fly different paths depending on where and what should be explored. As described in Section 2.2, the UAV path planning algorithm uses maximum 15 nodes, i.e. predefined positions, to calculate an optimal path. The UAV can therefore not fly over the whole area as simulated in Section 3.2.3. The predefined nodes can be the last observed position of the icebergs or new estimated positions. The cases below describe how the UAV path planner can be used with the different predefined positions, and how the cases will be implemented to be tested by simulation in Section 4.2.

**Case A** The first case uses information about the earlier positions of the icebergs in the area. This information may be satellite pictures or an earlier localization done by the UAV. If the time since the last estimation of location is relatively close to the current time or if the icebergs are moving slow, the UAV can fly to the last positions given by the information. The last positions of icebergs found can be added to the matrix "points". This matrix is a $N \times 3$ matrix, where $N$ is the number of icebergs. The two first elements are the position of the current iceberg, and the third element is the position uncertainty, $\sigma_{nodes}$, that the path planner algorithm will use. To begin with for this case, the value of the position uncertainty $\sigma_{nodes}$ is high and equal for all icebergs. The values will be presented in Section 4.2. The path planner will therefore in the beginning only use the distance between the icebergs for estimation of an optimal path. When the iceberg are found with the OGM, a new and lower value for that current iceberg is calculated. This calculation will be described in Section 4.1.3. The new value of the position uncertainty $\sigma_{nodes}$ is added to the matrix "points". The path planner will then calculate a new and optimal path based the new values in "points" and the distance between the nodes. When all the icebergs in the list are found, the UAV will continue to fly to the iceberg with the highest value in the matrix "points" to double check the position, and so on. This will be simulated to show how the OGM works with the path planner algorithm.

**Case B** If there have been some time since the update from satellite or last estimation of localization, the icebergs will have moved. Where the icebergs have moved is based on current, wind and drift. The iceberg drifting is difficult to estimate (Andersson et al., 2015). In this project, the iceberg will, therefore, be simulated to follow the current without estimating the drift. For this case, the UAV will fly to the new estimated positions of the icebergs, calculated from the current and speed of the icebergs. When the icebergs are moving, some icebergs stay inside the area, and some move outside the area. The information about the last position of icebergs

does not necessarily include information about incoming icebergs. The unknown edges of the area, therefore, need to be searched. One way this can be done is to place nodes along the edges where the UAV should fly. The UAV path planner can then calculate the optimal path between the nodes on the edges and the iceberg positions inside the area. For this case, the simulation will be divided into two parts. In the first simulation, the UAV will get information about iceberg positions from satellite pictures and fly to these positions. The UAV will find all the icebergs once and update the position uncertainty $\sigma_{nodes}$ along the way. In the second simulation, the icebergs are moved, and the new nodes for the icebergs and on the edge are added to the matrix "points". The UAV will then locate the icebergs again and update the position uncertainty as usual.

**Case C** Sometimes, the OGM estimates the probability of occupancy wrong due to the measurement errors $\epsilon_{\text{occ}}$ and $\epsilon_{\text{free}}$ presented in Section 3.2.2. This case can be used to double check the grids with a probability of occupancy over a certain value, which will be presented in Section 4.2.5. The simulation for this case is also divided in two. The first simulation finds a path between the icebergs given by the information from a satellite. After this simulation, all the grids in the path, with the probability of occupancy over the given value, will be added as nodes in the matrix "points". In the next simulation, the UAV will fly to the new nodes to check if there are icebergs at these positions or just wrong observed grids.

**Case D** The choice of OGM variables is important to accentuate. If the frequency of image taking is too low, the OGM may be more affected by the measurement errors $\epsilon_{\text{occ}}$ and $\epsilon_{\text{free}}$. For the simulations in this project, the image taking frequency is low, due to the simulations time will rapidly increase when increasing this frequency. A simulation with increased frequency of image taking is, therefore, in this project included to show how the OGM is affected by changing frequency of image taking, i.e. the UAV gets position information from the path planner more often per second. The path planner will work as for the other cases.

### 4.1.2 Path Planning Framework

A path planning framework based on the algorithm in Albert and Imsland (2015) is used to combine the UAV path planning algorithm and the OGM algorithm. This framework has the structure as Algorithm 4.1 show.

**Algorithm 4.1** Path Planning Framework

```
1: User input
2: for run=1:Nupdate do
3:     Update map and "points" with OGM
4:     Calculate visiting sequence
5:     Run simulation for time of UpdateTime
6: end for
```

Each line in Algorithm 4.1 is described as following:

1. The user input consists of variables that can vary from simulation to simulation. The variable ”SimulationTime” is the total time the simulation will run, while ”UpdateTime” decides how often the OGM algorithm is updated. The speed and minimum turning radius of the UAV are also chosen here, while ”TimeStep” shows how often the positions of the UAV are stored. The ”TimeStep” value will affect the results of OGM because the OGM needs overlapping images to make accurate estimations.
2. The for-loop will be running from 1 to ”Nupdate”, where

$$\text{Nupdate} = \left\lceil \frac{\text{SimulationTime}}{\text{UpdateTime}} \right\rceil . \tag{4.1}$$

3. In this line, the OGM program is run, which includes the OGM algorithm and the calculation of the positions uncertainty $\sigma_{nodes}$. The OGM program is updating the map of the iceberg positions, with the position of the UAV for the last seconds of the simulation depending on ”UpdateTime” as input. The output is the updated matrix ”points”, i.e. the iceberg positions and position uncertainty $\sigma_{nodes}$, that the path planner use. The OGM algorithm will be implemented as in Chapter 3, only the measurements and user input will be changed. These changes will be presented in Section 4.2. The calculation of the positions uncertainty $\sigma_{nodes}$ will be explained in Section 4.1.3.
4. Here, the optimal path between the nodes is calculated by the UAV path planning algorithm. This is based on the theory described in Section 2.2.
5. The simulation will run for the time of ”UpdateTime”, i.e. the UAV is flying for the value of ”UpdateTime”. The simulation output will be the sequence of positions the UAV has visited the last second depending on ”UpdateTime”, which will be the input for the OGM algorithm in the next iteration.

### 4.1.3 Calculation of Position Uncertainty used for Path Planning

The following calculation is a suggestion to how the positions uncertainty $\sigma_{nodes}$ can be calculated. The calculation of the positions uncertainty $\sigma_{nodes}$ is the equal for all the cases in this chapter. The positions uncertainty $\sigma_{nodes}$ was presented in Section 2.2.2, and is the value the path planner will use for the optimization to find the optimal path between the nodes. A high value of $\sigma_{nodes}$ represents high uncertainty. The path planner has a desire to visit the nodes with the highest value of $\sigma_{nodes}$ first. The positions uncertainty $\sigma_{nodes}$ is based on two values:

$$\sigma_{nodes} = \text{ProbabilityValue} + \text{TimeValue} \tag{4.2}$$

The ”ProbabilityValue” is calculated from the probability of occupancy from OGM. For icebergs consisting of multiple grids, the probability of occupancy for the each grid of the iceberg need to be added and dived on the number of grids the icebergs consists of.

Each iceberg then get one probability of occupancy, which is the average value of the probabilities. The probability of occupancy given by the OGM is represented by the log odds. The log odds is thus converted before the calculations. The "ProbabilityValue" may be decided by the following equation:

$$\text{ProbabiltyValue} = \left\lceil \frac{100 - \text{probability of occupancy}}{5} \right\rceil \cdot 100 \tag{4.3}$$

This will increase the "ProbabilityValue" by 100 points for every 5% of the probability of occupancy from OGM. Some examples are displayed in Table 4.1.

| Probability of occupancy | ProbabilityValue |
|---|---|
| 100% | 0 |
| 95%-100% | 100 |
| 90%-95% | 200 |
| ⋮ | ⋮ |
| 0%-5% | 2000 |

**Table 4.1:** Examples of calculation of the "ProbabilityValue" from the probability of occupancy.

The "TimeValue" is based on the time since the icebergs were observed. For this project, the TimeValue is calculated from the following equation:

$$\text{TimeValue} = \left[ 50 - \frac{t}{5} \right] \cdot 5 \tag{4.4}$$

where *t* is the simulation time. This equation rounds the number inside the parenthesis to the nearest integer. For instance, if a iceberg is found after 200 seconds the "TimeValue" is 50. This results in the first observed icebergs get higher "TimeValue" than the last observed icebergs. A examples of calculation of the position uncertainty for a iceberg is:

$$\text{Probability of occupancy} = 93\% \tag{4.5}$$
$$\text{ProbabilityValue} = 200 \tag{4.6}$$
$$\text{Simulation time} = 80 \text{ s} \tag{4.7}$$
$$\text{TimeValue} = 170 \tag{4.8}$$
$$\sigma_{nodes} = 370 \tag{4.9}$$

The positions uncertainty $\sigma_{nodes}$ for this iceberg is thus 370, which will be added to the matrix "points" and given to the path planner. The path planner will use positions uncertainty $\sigma_{nodes}$ and the distance between the nodes to calculate the optimal path.

## 4.2 Simulation and Results

In this chapter, the cases presented in Section 4.1.1 are simulated. The results showing the usage of OGM with a path planner. The path planner framework with OGM is presented in Section 4.1.2.

### 4.2.1 Iceberg and UAV Models

For the UAV, a Dubins Vehicle is used as a model (Albert and Imsland, 2015):

$$\begin{bmatrix} \dot{x}_{UAV} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} U\cos(\psi) \\ U\sin(\psi) \\ u \end{bmatrix} \tag{4.10}$$

where $x_{UAV}$ is the position, $U$ is the velocity, $\psi$ is the heading and $u$ is the actuator of the UAV. The UAV uses line of sight (LOS) algorithm from Fossen (2011) as a autopilot. In the cases to be simulated, the UAV starts in the position (0,0) for all cases, and fly with the speed of 22 m/s. The minimum turning radius of the UAV is 105.8 meter, which is a typical value for the UAV Penguin B which is used in Albert and Imsland (2015).

The model of the each iceberg is (Albert and Imsland, 2015):

$$\dot{\xi}_i = v_i \tag{4.11}$$

where $\xi_i$ is the position and $v_i$ is the known velocity. For Case A, C and D will $\xi_i$ be constant, while in Case B will the icebergs move.

### 4.2.2 Simulation Variables

The "SimulationTime" vary from each case, while the "UpdateTime" is for all cases 10 seconds. To begin with the "TimeStep" is be 0.5 seconds, which means that two images of the area are taken per second. The icebergs are spread in a $1300 \times 1300$ meter area. The grids are $100 \times 100$ meters each, i.e. there are $130 \times 130$ grids in the area. The number of grids in the UAV image, or measurement, are $5 \times 5$ grids. For real experiments, this size can be higher, but for simulation time purposes in this project the measurement grid size are chosen to be $5 \times 5$ grids. Due to the large number of grids in the area, the initial probability $p(\mathbf{m}_i)$ is not calculated as Equation (3.5) in Section 3.2.1. The value of $p(\mathbf{m}_i)$ will thus be sat 0.05, which is the same value as used in Chapter 3.

The measurement errors are sat to be the same as for the simulation in Chapter 3:

$$\epsilon_{\text{occ}} = 0.1 \tag{4.12}$$

$$\epsilon_{\text{free}} = 0.01 \tag{4.13}$$

This way 10% of the occupied grids and 1% of the free grids will be observed wrong, see Section 3.2.2.

### 4.2.3 Case A: Fly to Last Iceberg Location

For the simulation of Case A, the satellite pictures show seven icebergs in the $1300 \times 1300$ meter area. As described in Section 4.1.1 the position uncertainty $\sigma_{nodes}$ is to begin with be a high value. This high value of the position uncertainty $\sigma_{nodes}$ should be higher than the highest possible value the icebergs can get from the calculations described in Section 4.1.3. The initial matrix "points" is, therefore:

$$\text{points} = \begin{pmatrix} 1100 & 1100 & 5000 \\ 200 & 1100 & 5000 \\ 1100 & 500 & 5000 \\ 100 & 600 & 5000 \\ 700 & 900 & 5000 \\ 600 & 300 & 5000 \\ 400 & 800 & 5000 \end{pmatrix} \tag{4.14}$$

where the two first columns are the position and the third is the position uncertainty $\sigma_{nodes}$. Each line represents one iceberg. The UAV is for this case visiting seven icebergs, and seven nodes are thus used to calculate the optimal path. The initial iceberg positions are illustrated in Figure 4.1.

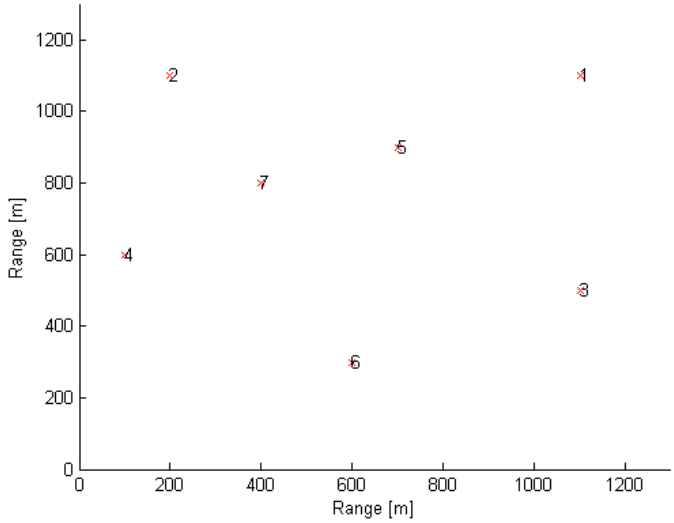


**Figure 4.1:** Initial node positions used in the path planning.

The "SimulationTime" for this case is 400 seconds to make the UAV fly two rounds to visit each iceberg twice. As explained in Section 4.1.1, this is done to show the functioning of OGM with a path planner. The UAV is flying to the last positions observed by the satellite. Figure 4.2 show the path of the UAV as the output of OGM for simulation time

$t = [100, 200, 300, 400]$. The UAV path relative to the iceberg numbering is illustrated in Figure 4.3.

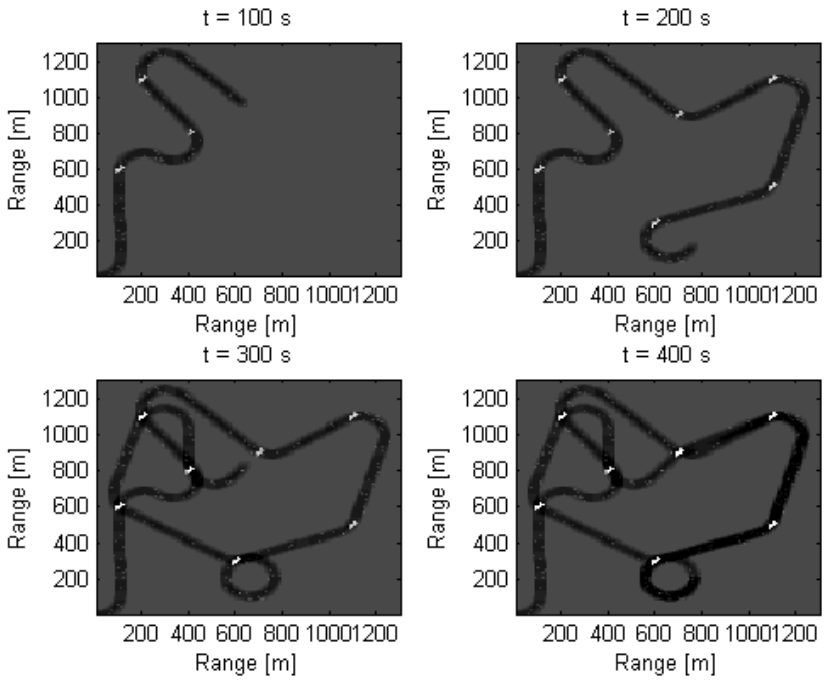


**Figure 4.2:** Case A: Fly to last iceberg location. Estimated location of icebergs with flight path calculated by the path planner at simulation time $t = [100, 200, 300, 400]$ seconds. The white and light grey grids are occupied, while the black and dark grey are empty. The grey grid represents unexplored grids. The UAV is visiting each node twice.

As explained in Section 4.1.1, the position uncertainty $\sigma_{nodes}$ is updated while the simulation is running. The UAV path planner calculates the optimal path based on the distance between the nodes, but also the positions uncertainty $\sigma_{nodes}$. The positions uncertainty $\sigma_{nodes}$ is in the beginning 5000 for all the nodes, and when the UAV reaches a node, the new positions uncertainty $\sigma_{nodes}$ is calculated as Section 4.1.3 proposes. The UAV path planner then calculates the optimal path, and the node is not prioritized in the path due to the low positions uncertainty $\sigma_{nodes}$ compared to the other nodes. The path planner chooses the node with the highest positions uncertainty $\sigma_{nodes}$ and the shortest distance from the current position. The positions uncertainty $\sigma_{nodes}$ for the nodes calculated during the simulation are presented in Table 4.2.

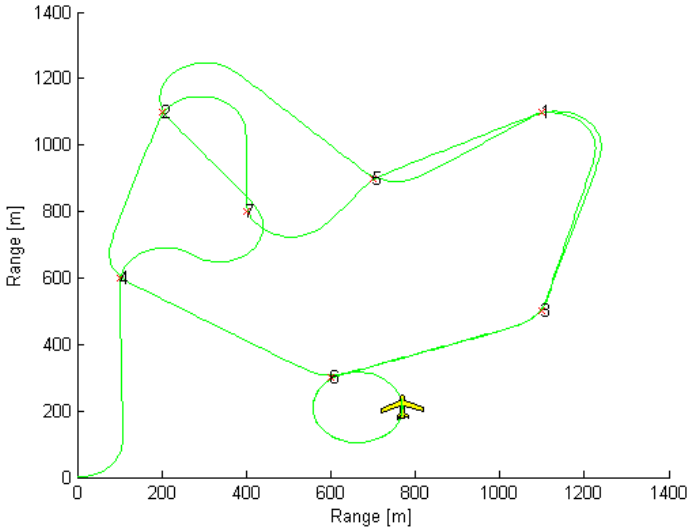


**Figure 4.3:** Case A: Fly to last iceberg location. Illustrated UAV flight path with iceberg numbering at simulation time $t = 400$. The UAV is visiting each node twice.

| Iceberg no. | $t = 0$ | $t = 100$ | $t = 200$ | $t = 300$ | $t = 400$ |
|---|---|---|---|---|---|
| 1 | 5000 | 5000 | 220 | 220 | -80 |
| 2 | 5000 | 270 | 270 | -20 | -20 |
| 3 | 5000 | 5000 | 190 | 190 | -110 |
| 4 | 5000 | 310 | 310 | 10 | 10 |
| 5 | 5000 | 5000 | 240 | 240 | -60 |
| 6 | 5000 | 5000 | 160 | 130 | -140 |
| 7 | 5000 | 290 | 290 | 60 | 60 |

**Table 4.2:** Uncertainty value after the given simulation time t.

From the results in Figure 4.2 and Table 4.2, there are two topics to discuss. The first is the behavior of OGM for this case, and the second is the behavior of the path planner with OGM for this case.

For this case, the UAV is flying over the icebergs, i.e. the predefined nodes, twice. The OGM estimates, therefore, a probability of occupancy multiple times for each iceberg. At simulation time $t = 200$ second, see Figure 4.2 are all the icebergs found, and there are seven white areas in the map. Each of these icebergs has the probability of occupancy close to 100% due to overlapping images. As mentioned in Section 4.2.2, the ”TimeStep” is 0.5 seconds, which means that there is two images taken per second. The second time the UAV is flying over the icebergs, the probability of occupancy is calculated based on the previous estimations, which results in a increased probability of occupancy. The icebergs

get, therefore, a probability of occupancy closer to 100%, and the grids becomes more white. The opposite applies to the empty grids; closer to 0% and darker grey grids. In Figure 4.2 the map for simulation time $t = 400$ seconds have whiter grids for the icebergs, than for the map at simulation time $t = 200$. The output of OGM is thus behaving as expected, and the icebergs can be located in this map.

The path changes based on the output of OGM. The path planner bases the optimal path calculations on both shortest distance and the positions uncertainty $\sigma_{nodes}$. In Figure 4.2 at simulation time $t = 200$ seconds, all the nodes have new calculated positions uncertainty $\sigma_{nodes}$ values. The path planner continues to prioritize the nodes with the highest positions uncertainty $\sigma_{nodes}$. After the simulation time $t = 200$, the path planner thus is prioritizing iceberg no. 4, see Table 4.2. The next iceberg after no.4 is no. 2 instead of no. 7, due to the turning radius, even though no.7 has a higher positions uncertainty $\sigma_{nodes}$ than no. 2. The next nodes are the same sequence as the first round. Every time the iceberg is observed, the probability of occupancy increases and the positions uncertainty $\sigma_{nodes}$ decreases. The positions uncertainty $\sigma_{nodes}$ is, therefore, decreasing from the values at $t = 200$ to $t = 300$ and $t = 400$, see Table 4.2. Some of the new estimated position uncertainty values at $t = 300$ and $t = 400$ are negative. The path planner can use negative values, but if the negative values are undesirable, Equation 4.4 in Section 4.1.3 can be changed. The probability of occupancy for the icebergs are all between 95% and 100%, the "TimeValue" thus have a greater impact than the "ProbabilityValue" when calculating the positions uncertainty $\sigma_{nodes}$ in Equation 4.2 in Section 4.1.3.

### 4.2.4 Case B: Fly to New Estimated Location

In this case, the icebergs move. The simulation for this case is be divided into two separate simulations.

**Part 1** The first simulation, simulates and estimates the positions given from the satellite information and calculates the position uncertainty $\sigma_{nodes}$ of the nodes to be used in the next part.

**Part 2** The next simulation, simulates the UAV fly to the positions where the icebergs are estimated to have moved after a given time based on the position uncertainty $\sigma_{nodes}$ from the first simulation.

In part 1 the positions of the nodes are given by the following matrix:

$$\text{points} = \begin{pmatrix} 1100 & 1100 & 5000 \\ 200 & 1100 & 5000 \\ 1100 & 500 & 5000 \\ 100 & 600 & 5000 \\ 700 & 900 & 5000 \\ 600 & 300 & 5000 \\ 400 & 800 & 5000 \end{pmatrix} \tag{4.15}$$

which is the same as for Case A in Section 4.2.3, see Figure 4.1. The estimated position map of the 200 first seconds is shown in Figure 4.4 and the UAV path with the iceberg

numeration is illustrated in Figure 4.5. The calculations and execution is the same as for the first 200 second of Case A, which is described in Section 4.2.3.

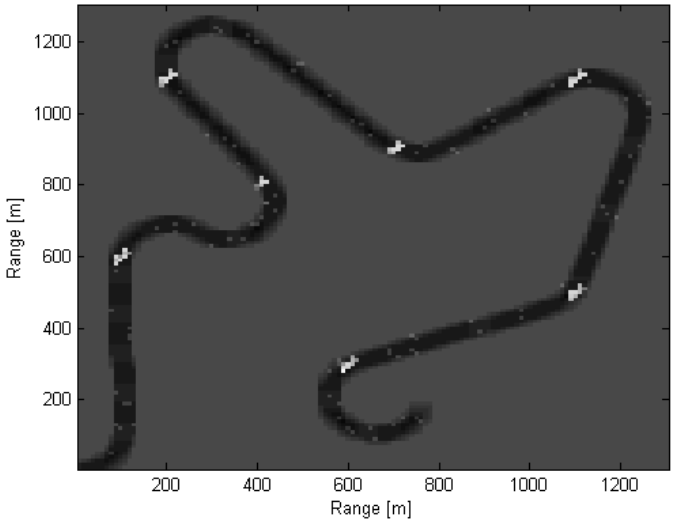


**Figure 4.4:** Case B: Fly to new estimated location (Part 1). Estimated location of icebergs with flight path calculated by the path planner at simulation time $t = 200$ seconds. The white and light grey grids are occupied, while the black and dark grey are empty. The grey grid represents unexplored grids.

After the simulation based on the matrix "points" in Equation 4.16, the calculated new matrix "points" is:

$$\text{points} = \begin{pmatrix} 1100 & 1100 & 220 \\ 200 & 1100 & 270 \\ 1100 & 500 & 190 \\ 100 & 600 & 310 \\ 700 & 900 & 240 \\ 600 & 300 & 160 \\ 400 & 800 & 290 \end{pmatrix} \tag{4.16}$$

The next simulation, i.e. part 2, use the position uncertainties in this matrix to calculate a new optimal path, only for the next simulation, the position of icebergs have moved. Based on current, the icebergs moves 100 meters straight north. The south edge of the area, therefore, need to be investigated for new and incoming icebergs. Nodes along this edge are therefore added to the matrix "points" with a high position uncertainty $\sigma_{nodes}$ to make the path planner prioritize the edge.

The simulation is therefore starting with the matrix "points" as following:

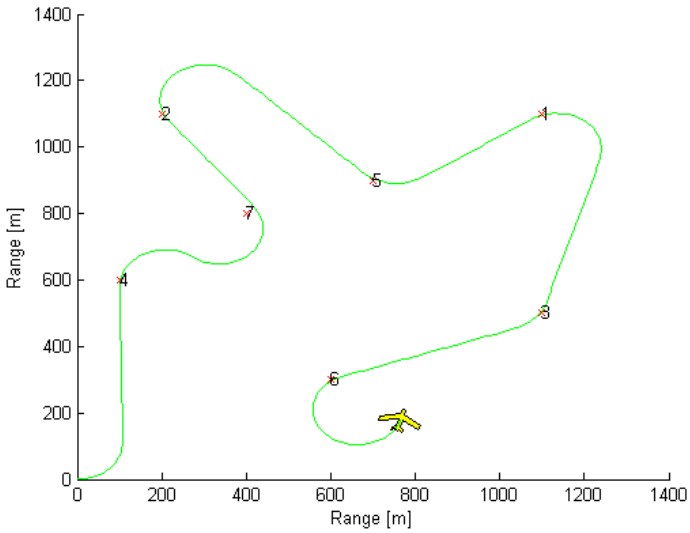


**Figure 4.5:** Case B: Fly to new estimated location (Part 1). Illustrated UAV flight path with iceberg numbering at simulation time $t = 200$.

$$\text{points} = \begin{pmatrix} 1100 & 1200 & 220 \\ 200 & 1200 & 270 \\ 1100 & 600 & 190 \\ 100 & 700 & 310 \\ 700 & 1000 & 240 \\ 600 & 400 & 160 \\ 400 & 900 & 290 \\ 200 & 50 & 5000 \\ 350 & 50 & 5000 \\ 500 & 50 & 5000 \\ 650 & 50 & 5000 \\ 800 & 50 & 5000 \\ 950 & 50 & 5000 \\ 1100 & 50 & 5000 \\ 1250 & 50 & 5000 \end{pmatrix} \tag{4.17}$$

where the seven first rows are the nodes from the part 1 and the eight next are nodes on the south edge of the area. The ”SimulationTime” for this simulation is 240 seconds. The output of the simulation, based on these values, is shown in Figure 4.6, while the UAV path with the iceberg numbering is illustrated in Figure 4.7.

The simulation shows the iceberg found in part 1 have moved 100 meter north, and there are no new icebergs on the south edge. The output of OGM show white grids where the

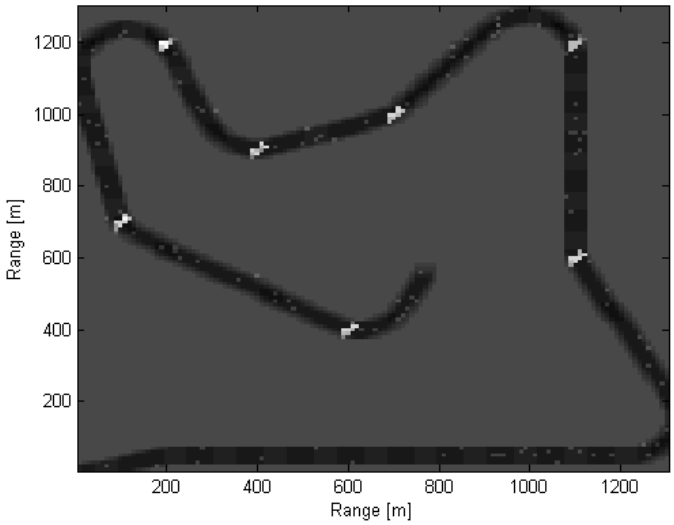


**Figure 4.6:** Case B: Fly to new estimated location (Part 2). Estimated location of icebergs with flight path calculated by the path planner at simulation time $t = 240$ seconds. The white and light grey grids are occupied, while the black and dark grey are empty. The grey grid represents unexplored grids.

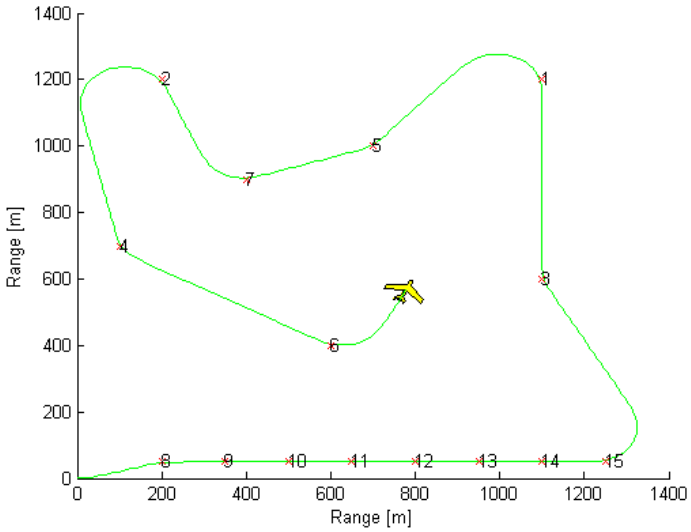


**Figure 4.7:** Case B: Fly to new estimated location (Part 2). Illustrated UAV flight path with iceberg numbering at simulation time $t = 200$.

icebergs are located, and all the icebergs have a probability of occupancy near 100%, due to overlapping as explained for Case A. In this case, the UAV flight path is different in part 2 than in part 1. The UAV is here flying the opposite order than in the simulation of part 1. This is due to where the UAV is starting. Here, the UAV examines the edge first and, therefore, starts exploring the icebergs in another corner. Due to the distance between the icebergs and the current position, the path planner chooses iceberg no.3 over iceberg no.6, see Figure 4.7. The position uncertainty from Equation (4.16), does therefore not have as big impact on the path planning as expected.

### 4.2.5 Case C: Exploring Unexpected Icebergs

In this case, unexpected icebergs are explored and double checked to see if they are errors from OGM or not. The simulation for is be divided into two parts.

**Part 1** The first simulation is based on the information about the area from satellite pictures. The results of this simulation show all grids with the probability of occupancy over a certain value.

**Part 2** The next simulation uses the results from part 1 to check if the grids are icebergs or errors produced by the OGM.

The first simulation simulates the path planning with the positions and the position uncertainty $\sigma_{nodes}$ in the following matrix "points":

$$\text{points} = \begin{pmatrix} 1100 & 1100 & 5000 \\ 200 & 1100 & 5000 \\ 1100 & 500 & 5000 \\ 100 & 600 & 5000 \\ 700 & 900 & 5000 \\ 600 & 300 & 5000 \\ 400 & 800 & 5000 \end{pmatrix} \tag{4.18}$$

The estimated position map from OGM is shown in Figure 4.8, where all grids with probability of occupied higher than approximately 27%, i.e. log odds -1, are marked. As shown in the figure, some single grids and the icebergs from Equation (4.18) are marked. The reason for only marking grids with probability of occupancy over approximately 27%, is due to computational complexity for the simulation of part 2. The computational complexity is mentioned in Section 2.2.3. In Figure 4.8 are no grids with probability of occupancy between 25% and 63%. The limit is therefore sat to be 27% to exclude the grids with probability of occupancy under 25%. If the path planner would tolerate over 15 nodes, it would be interesting to include grids with probability of occupancy down to 15-20%.

The purpose of marking the grids with probability of occupancy over 27%, is to check whether the grids with a low probability of occupancy are errors, due to the measurement errors $\epsilon_{\text{occ}}$ and $\epsilon_{\text{free}}$, or if the grids actually are occupied. When a grid has probability of occupancy at 27%, there is a 27% chance to find the grid occupied, and the grid should thus be double checked. If double checking is not possible, all grids over this probability should

be assumed to be icebergs. In this case, false positives are better than false negatives, i.e. assuming the grid occupied when it is empty is better than assuming the grid is free when it is occupied.

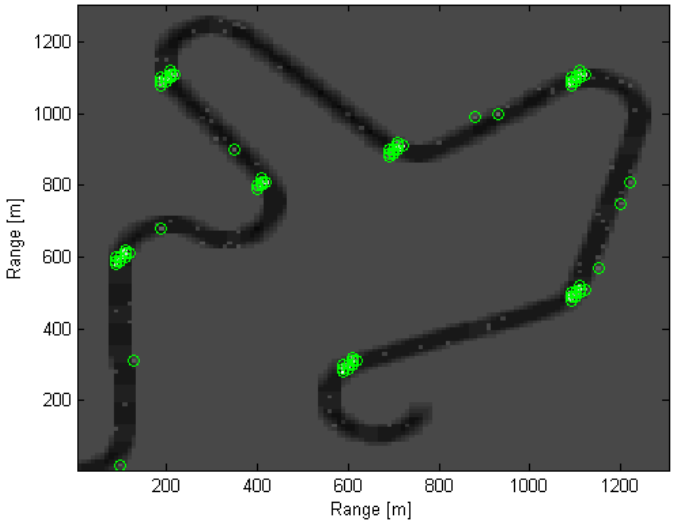


**Figure 4.8:** Case C: Exploring unexpected icebergs (Part 1). Estimated location of icebergs with flight path calculated by the path planner at simulation time $t = 200$ seconds. The white and light grey grids are occupied, while the black and dark grey are empty. The grey grid represents unexplored grids. The green circles mark the grids with probability of occupancy over 27%.

The new and unexpected grids with probability of occupancy at 27% marked in Figure 4.8 need to be explored to be checked. The new nodes with the position uncertainty $\sigma_{nodes}$ calculated in part 1 of this case, are given by the matrix "points":

$$\text{points} = \begin{pmatrix} 100 & 20 & 1050 \\ 130 & 310 & 1050 \\ 1150 & 570 & 380 \\ 190 & 680 & 430 \\ 1200 & 750 & 980 \\ 1220 & 810 & 980 \\ 350 & 900 & 440 \\ 880 & 990 & 390 \\ 930 & 1000 & 290 \end{pmatrix} \tag{4.19}$$

In part 2 of this case, only the single grids are examined, and not the icebergs given in Equation 4.18, which is due to the computational complexity of the path planner. If the icebergs was included, the position uncertainty $\sigma_{nodes}$ of the icebergs would be higher due to higher probability of occupancy calculated from OGM compared to the nodes in Equation (4.19). Figure 4.9 show the output of OGM when the path planner is based on

only the matrix ”points” in Equation (4.19). The grids with probability of occupancy over 27% are also marked in this figure. The UAV flight path with the numberation of the nodes in Equation (4.19) is illustrated in Figure 4.10.

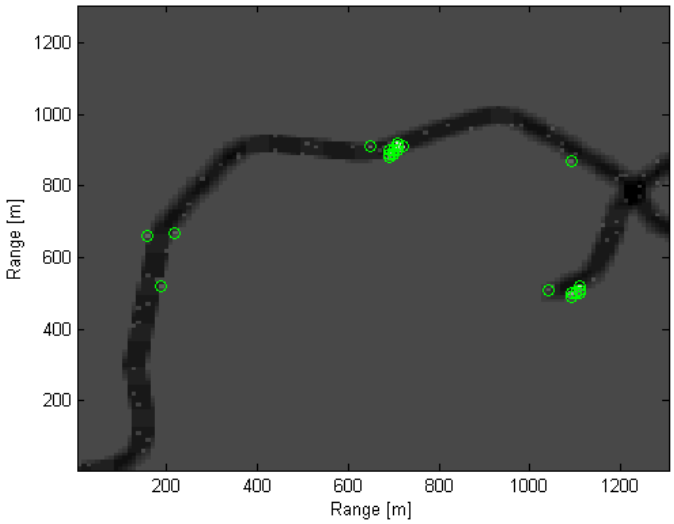


**Figure 4.9:** Case C: Exploring unexpected icebergs (Part 2). Estimated location of icebergs with flight path calculated by the path planner at simulation time $t = 160$ seconds. The white and light grey grids are occupied, while the black and dark grey are empty. The grey grid represents unexplored grids. The green circles mark the grids with probability of occupancy over 27%.

The probability of occupancy for the unexpected nodes for the two parts of the simulation are presented in Table 4.3. The node numbering refers to the node order in the matrix ”points” in Equation (4.19), which is illustrated in Figure 4.10. The probability of occupancy for the unexpected icebergs observed in part 1 is decreasing the probability of occupancy from over 27% in part 1 to approximately 0% in part 2. This means the grids with probability of occupancy over 27% found in Figure 4.8, are wrong observed grids. The output of the OGM, i.e. the probability of occupancy for each grid, have been affected by the measurement errors $\epsilon_{\text{occ}}$ and $\epsilon_{\text{free}}$, see Section 3.2.2. These grids are therefore not occupied with icebergs and are thus empty. The next case will show how to reduce the effect of the measurement errors $\epsilon_{\text{occ}}$ and $\epsilon_{\text{free}}$.

Here, the OGM is doing what is expected, finding icebergs and double checks unexpected single icebergs. Due to the start position of the UAV and the positions uncertainties of the nodes in Equation (4.19), the UAV first visits node 1 and 2. After visiting these nodes, the next node is, naturally, the closest.

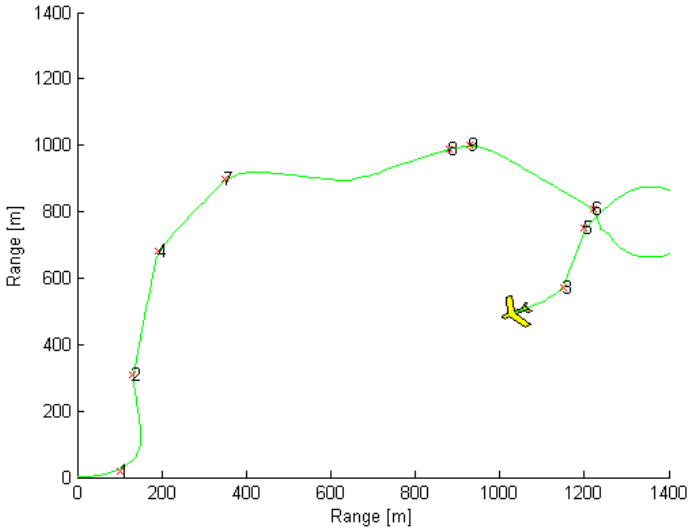


**Figure 4.10:** Case C: Exploring unexpected icebergs (Part 2). Illustrated UAV flight path with node numbering at simulation time $t = 160$.

| Node no. | Probability Part 1 (%) | Probability Part 2 (%) |
|---|---|---|
| 1 | 63.3 | 0.003 |
| 2 | 63.3 | 0.001 |
| 3 | 90.0 | 0.001 |
| 4 | 91.6 | 0.001 |
| 5 | 63.3 | 0.003 |
| 6 | 63.3 | $5.0 \cdot 10^{-5}$ |
| 7 | 91.6 | 0.003 |
| 8 | 90.0 | 0.001 |
| 9 | 98.3 | 0.007 |

**Table 4.3:** Probability of occupancy of the unexpected icebergs of Part 1 and Part 2 in Case C.

### 4.2.6 Case D: Reduce the Effect of Measurement Errors

In Case C, the effect of the measurement errors $\epsilon_{\text{occ}}$ and $\epsilon_{\text{free}}$ was presented and discussed. The purpose of the simulation in Case D is to show how the effect of the measurement errors $\epsilon_{\text{occ}}$ and $\epsilon_{\text{free}}$ can be reduced when simulating.

The measurement errors $\epsilon_{\text{occ}}$ and $\epsilon_{\text{free}}$ are added errors to the simulation to represent image processing errors and other errors connected to the measurements. The explanation of the measurement errors is presented in Section 3.2.2. For this case, the measurement errors have the same values as for the simulations in Chapter 3. 10% of the occupied grids are observed as empty, and 1% of the empty grids are observed as occupied. Due to multiple overlapping images over the same area and grids, the measurement errors in most cases do not have a significant impact. If the image taking frequency is low, the images overlap less, and the OGM gets less data to calculate the probability of occupancy from. The measurement errors $\epsilon_{\text{occ}}$ and $\epsilon_{\text{free}}$ then affect the output of OGM more.

For the simulation in this case, the image taking frequency is increased to reduce the effect of the measurement errors and avoid the unexpected icebergs discussed for Case C in Section 4.2.5. The "SimulationTime" and "UpdateTime" will be unchanged, 200 and 10 seconds, respectively. The variable to change during this simulation is "TimeStep". This variable is decreased, such that the OGM receives measurements more frequently than in the simulations of the previous cases. The previous simulations had a "TimeStep" value at 0.5, i.e. two images per second. "Timestep" is for this simulation be 0.1, i.e. ten images per second. The initial positions are the same as for the simulations in the previous cases:

$$\text{points} = \begin{pmatrix} 1100 & 1100 & 5000 \\ 200 & 1100 & 5000 \\ 1100 & 500 & 5000 \\ 100 & 600 & 5000 \\ 700 & 900 & 5000 \\ 600 & 300 & 5000 \\ 400 & 800 & 5000 \end{pmatrix} \tag{4.20}$$

The estimated map after the simulation time $t = 200$ seconds are shown in Figure 4.11. Note, the grey tones in this map are different compared to the grey tones in the previous simulations in Case A-C. In this simulation, where the "TimeStep" value is decreased, the path of the UAV is constant black, with the exception of the edges of the path. However, in the other simulations, different grids have different grey tones in the path. The same applies for the icebergs, i.e. the white grids. In this simulation, the grids are constant white instead of different shades of light grey, see Figure 4.11. As explained before, the black grids refer to the probability of occupancy being 0%, the white grids means 100%, and the grey tones are between 0-100%. This means when the "TimeStep" value is decreased, and the OGM estimates the probability of occupancy and positions based on a multiple number of images and measurement, the resulting map produced from the OGM is more accurate and gives a better probability of occupancy results. A high frequency of image

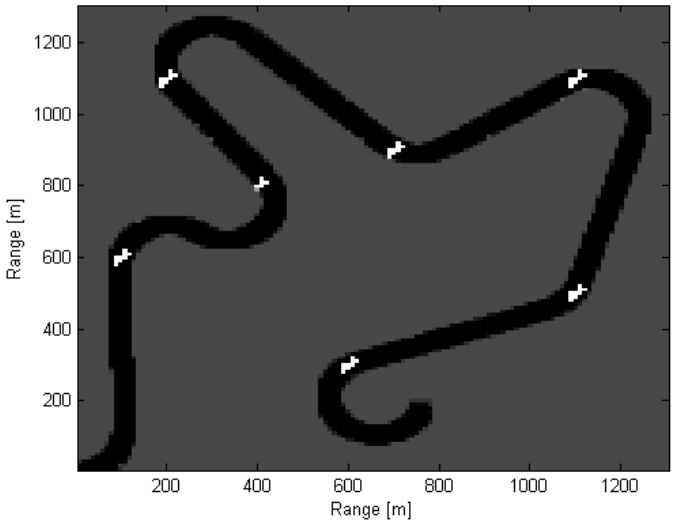


**Figure 4.11:** Case D: Reduce the effect of measurement errors. Estimated location of icebergs with flight path calculated by the path planner at simulation time $t = 200$ seconds. The white and light grey grids are occupied, while the black and dark grey are empty. The grey grid represents unexplored grids. "TimeStep" value reduced to 0.1.

taking reduces the effect of the measurement errors $\epsilon_{\text{occ}}$ and $\epsilon_{\text{free}}$, and is thus desirable when simulating.

The main disadvantage of increasing the image taking frequency is the calculation time. When the OGM get information from ten images instead of two images per second, the number of calculations of the probability of occupancy increase. This is the reason for why the simulations in this project use the "TimeStep" value 0.5 instead of 0.1. As a comparison, when timestep $= 0.1$ the OGM and the measurement calculations use around 28 seconds for the 10 seconds information, while for timestep $= 0.5$ the calculations use 5 seconds for the 10 seconds information. The OGM program uses approximately the same for each iteration, but the total calculation time is essential for the UAV due to time and range limitations. Note, the calculation time for this project is depending on the implementation, computer processing and MATLAB, and can differ if this is tested for experiments in real life.

In Figure 4.12, the grids with probability over 27% have been marked. As the figure show, there are almost none grids marked except the icebergs with 100% probability. For this simulations, only four new grids are marked, but these are at the edge of the UAV path. The edges are explored less times than in the middle of the path, and the measurement errors affect the probability of occupancy more at the edge. This is intuitive, because the image is squared and the UAV is not flying in a straight line. The OGM is thus for this case estimating the probability of occupancy and finding the icebergs as expected.

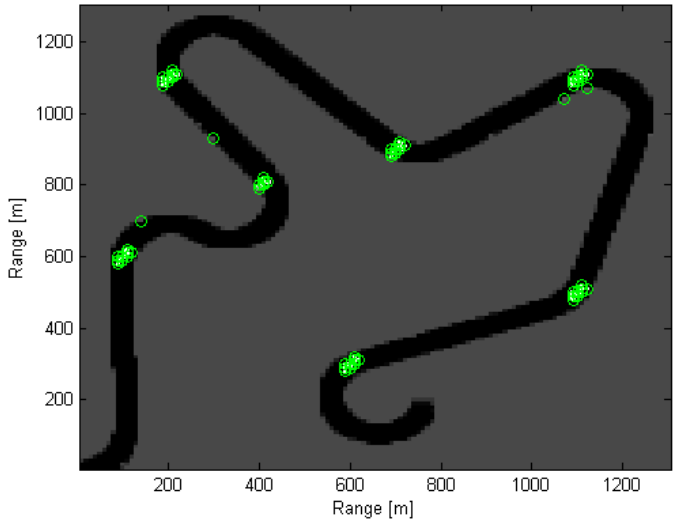


**Figure 4.12:** Case D: Reduce the effect of measurement errors. Estimated location of icebergs with flight path calculated by the path planner at simulation time $t = 200$ seconds. The white and light grey grids are occupied, while the black and dark grey are empty. The grey grid represents unexplored grids. "TimeStep" value reduced to 0.1. The green circles mark the grids with probability of occupancy over 27%.

Chapter 5

# Discussion

## 5.1 Occupancy Grid Mapping for Iceberg Localization

All the cases in Chapter 3, show that icebergs can be located by using Occupancy Grid Mapping. The following discussion will compare the results of each case in Chapter 3.

The simulation of Case 1 in Section 3.2.3 show that if the UAV measurements are done often enough, the estimation of the probability of occupancy will be very accurate. This is regardless if the measurement errors $\epsilon_{\text{occ}}$ and $\epsilon_{\text{free}}$ are added to the simulation or not. The disadvantage of this case is the large simulation time.

The simulation results of Case 2, clearly show that the probability of occupancy depends on how many times each grid is observed. When the UAV observe grids on the diagonal, the grids will only be observed a few times. In the corner, when the UAV is flying back and fourth, the grids will be observed more times than on the diagonal. The color difference thus represents the difference in the probability of occupancy in log odds. As discussed in Section 3.2.6, the measurement errors $\epsilon_{\text{occ}}$ and $\epsilon_{\text{free}}$ will affect the probability of occupancy more when the grids are observed few times.

Even though the measurement errors $\epsilon_{\text{occ}}$ and $\epsilon_{\text{free}}$ affects the probability of occupancy, the output of OGM will show where the icebergs are located in the area. Section 3.2.6 discusses the limit of the probability of occupancy between occupied grids and free grids. The maps, build in Section 3.2.6, had the limit of log odds -1, which is a probability of occupancy at 27%. As discussed, it is better to assume a grid being occupied when the grid is empty, rather than assuming a free grid when it is occupied. Assuming an occupied grid to be free can be dangerous and possibly damaging for operations and vessels.

Case 1 and Case 2 use a lot of simulation time to calculate the probability of occupancy for the grids when exploring every grid multiple times. The back and forth method are

thus unacceptable when the UAV have simulation time limits and the calculations should happen real time. Another back and fourth method can be acceptable if the flight path is more realistic. If the image is covering $100 \times 100$ meter, the UAV can, for instance, fly back and forth every 100 meters such that the area will be covered, but not as accurate as in Case 1 and Case 2. This approach will probability reduce the simulation time. However, the OGM should be used with a path planner to avoid unnecessary flight time. Case 3 show an example of a flight path when the UAV is not flying back and forth. This reduces the simulation time, and the icebergs can clearly be located from the map. This flight path is not realistic due to range and turning radius of the UAV. In Case A-D in Chapter 4, the OGM was simulated with the path planner with realistic turning radius and speed of the UAV.

## 5.2 Occupancy Grid Mapping with UAV Path Planning

The simulations of Occupancy Grid Mapping with the path planner in Case A-D, show a more realistic path of the UAV based on real values of speed and turning radius.

The simulation of Case A shows that the results of the probability of occupancy will get more accurate when the UAV is flying over the icebergs twice. The average probability of occupancy for each iceberg will increase, and the probability of occupancy for the empty grids will decrease. The map created by OGM show where the icebergs are located in the area.

The simulation results of Case B show that the path planner can use the OGM to estimate the position of icebergs at new locations and along the edge of the area. The map produced by OGM clearly show where the icebergs are located relative to the starting point. The path planner algorithm can maximum have 15 nodes to make the calculation time acceptable. This case has 15 nodes, but if other areas should be investigated at the same flight, some nodes could be merged to get the number of nodes to maximum 15.

In Case C was the grids with the probability of occupancy over 27% investigated to see if the grids were results of the measurement error effect or actually occupied grids. The results from the simulation part 2 show that the unexpected nodes were empty, and, therefore, results of the measurement errors $\epsilon_{\text{occ}}$ and $\epsilon_{\text{free}}$.

In Case D was the ”TimeStep” value reduced to increase the frequency of the image taking from the UAV. The results show the effect of measurement error was reduced, and only the grids on the edge of the path were affected by the measurement errors. The probability of occupancy for the grids were more accurate in this case than for the other cases, and the variations in probability of occupancy between the grids were reduced.

For all the cases, the position uncertainty $\sigma_{nodes}$ was calculated from the output of OGM. The proposed calculations of the position uncertainty were based on the probability of occupancy from OGM and time since the current grid was visited. The simulations of the cases in Chapter 4 showed that placement of the iceberg in the area and time since visiting the iceberg had a greater impact on the position uncertainty than the probability of

occupancy. This is due to the variations of the probability of occupancy between the grids. All the icebergs had an average probability of occupancy close to 100%. The probability of occupancy has a greater impact when the variations are bigger. In Case C, the grids with probability over 27% were explored. The grids with a lower probability than 100% will get a higher position uncertainty $\sigma_{nodes}$, than the position uncertainty of the predefined icebergs. The proposed calculations of the position uncertainty work, therefore, best for nodes with probability of occupancy spread out on the probability scale from 0% to 100%, and not when all icebergs have probability of occupancy close to 100%.

The simulations could be made more realistic with a few changes. The image area could be increased, and the measurements could be based on real observations and actual image processing. The area used in this project is large, and the ideally can the area be smaller to fit the image size. The frequency of image taking should be high enough to avoid major effects by the measurement errors. But when increasing frequency of image taking, the probability of occupancy for the grids will have little to no variations. The position uncertainty $\sigma_{nodes}$ calculations proposed in this project, depends then only on time and position of the nodes relative to each other and the position of the UAV.

The main disadvantage of using the path planner is the path planner needs predefined nodes to calculate a path. The OGM, therefore, needs to search the area to find icebergs and give the nodes to the path planner, for then again locate the icebergs with the flight path from the path planner. This requires an extra flight for the UAV if this position of icebergs unknown. The best usage of the OGM with the path planner is therefore for tracking and updating the position of known icebergs in a given area.

The main advantage of using Occupancy Grid Mapping for iceberg localization is the simplicity. The Occupancy Grid Mapping algorithm is easy to understand and implement. Additionally, based on the theory, the computational time is lower for Occupancy Grid Mapping than for other grid based estimation methods due to independent grids.

Chapter 6

# Conclusion

In this project, Occupancy Grid Mapping was implemented and simulated for building a grid-based map showing iceberg location based on UAV measurements. This was done by dividing a rectangular area into grids, and for each grid, a probability of occupancy was calculated. High probability of occupancy represented the current grid being occupied by ice, while a low probability of occupancy means that the grid was empty and free from ice. The Occupancy Grid Mapping algorithm was most accurate when the frequency of image taking was high, such that the calculations of the probability of occupancy can be based on multiple images. The grid-based map, build by Occupancy Grid Mapping, illustrated the position of icebergs based on the probability of occupancy, and the Occupancy Grid Mapping can, therefore, be used for iceberg localization. The strength of Occupancy Grid Mapping is the simplicity, while the weakness is the computational time when the estimation is based on a high number of overlapping images.

A procedure for using the information in Occupancy Grid Mapping for UAV path planning was proposed. The proposed calculations were also implemented and tested. The proposed procedure was based on using the probability of occupancy from Occupancy Grid Mapping to calculate the path planner optimization variable, the position uncertainty. This position uncertainty value was calculated from the probability of occupancy from Occupancy Grid Mapping and time since the observations. This procedure worked best when the variations of probability of occupancy between grids were significant. Based on the simulations results, the conclusion is that the information from Occupancy Grid Mapping can be used for UAV path planning.

## 6.1 Further Work

Further work of this project may be to use the Occupancy Grid Mapping with the path planner to explore unknown areas. The path planner may be extended with the possibility

to explore areas and not only specific icebergs. With this extension of the path planner, the Occupancy Grid Mapping can locate icebergs in unexplored areas.

Extensions of Occupancy Grid Mapping is also possible. The Occupancy Grid Mapping method may be extended to be used for estimation of ice concentration where ice covers major parts of the area. A suggestion of this case is to divide the area into grids and use Occupancy Grid Mapping to find the border between ice and sea.

If the work, done in this project, is to be used for other experiments, some adjustments may be necessary. The frequency of image taking depends on the camera, but can be higher than in this project. The searching area can be made smaller, and the image area can be made bigger, to make the ratio more realistic. The calculations of the position uncertainty proposed in this project should be further developed to fit the experiments and to make the positions uncertainty count more.

# Bibliography


Albert, A., Imsland, L., 2015. *Mobile Sensor Path Planning for Iceberg Monitoring Using a MILP Framework*.

Andersson, L. E., Scibilia, F., Imsland, L., 2015. *The Moving Horizon Estimator Used in Iceberg Drift Estimation and Forecast*.

Arulampalam, M., Maskell, S., Gordon, N., Clapp, T., 2002. *A tutorial on Partcle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking.* IEEE Transactions on singal processing 50 (2), 174–188.

Bruteig, C., 2015. *Estimation for Mobile Sensor Path Planning Using Grid-Based Bayesian Filters (Project Thesis)*.

Cormen, T., Leiserson, C., Rivest, R., Stein, C., 2009. *Introduction to Algorithms*. The MIT Press.

Fossen, T. I., 2011. *Handbook of marine craft hydrodynamics and motion control*. John Wiley and Sons.

Leira, F. S., Johansen, T. A., Fossen, T. I., 2015. *Automatic Detection , Classification and Tracking of Objects in Ocean Surface from UAVs Using a Thermal Camera*.

Särkkä, S., 2013. *Bayesian Filtering and Smoothing*. Cambridge University Press.

Thrun, S., Burgard, W., Fox, D., 2006. *Probabilistic Robotics*. The MIT Press.