ABSTRACT
Detailed information about the subsurface section of an iceberg is of high value to researchers and offshore operators, and can be used to create hydrodynamic models of icebergs, calculate impact loads, and to forecast iceberg drift. However, since the iceberg will translate and rotate when affected by water currents, mapping the underwater geometry of an iceberg is non-trivial. This paper will describe an algorithm for automatic detection and mapping of drifting icebergs using autonomous underwater vehicles (AUVs). Using an upwards-looking multibeam echosounder (MBE), the AUV monitors a given area in detection mode. When exceeding a specified ice draft threshold, the AUV will enter mapping mode. The AUV will search for the edge of the iceberg, and follow the edge once detected. Through this paper, the detection procedure based on measurements from a MBE is presented, and the algorithm for mapping the iceberg while the iceberg is drifting is outlined. The results will be presented in a simulation study, which will show the effectiveness of the mapping procedure for a simulated iceberg drifting with constant, linear speed. The AUV will perform a complete circumnavigation of the iceberg, before completing the mapping procedure, and continuing its pre-defined mission.

INTRODUCTION
Obtaining an accurate description of the keel geometry of an iceberg is of increasing interest for researchers and offshore operators. Information about the iceberg keel geometry is needed to incorporate detailed water current information into iceberg drift modeling to calculate the water drag forces at different depths (Kubat et al., 2005). An operational model of iceberg drift is presented by Kubat et al. (2005), where an empirical model of the keel cross-sectional area depending of waterline length is used (Barker et al., 2004), and development of such empirical models would greatly benefit from detailed iceberg surveys. Similarly, repeated surveys of the keel geometry of a given iceberg is needed to develop and verify iceberg deterioration models, as stated by Murphy and Carrieres (2010).

In Arctic offshore activities, where there may exist a threat of sea-ice, and/or icebergs, ice management (IM) must be employed. Eik (2008) defines IM as the sum of all activities where the objective is to reduce, or avoid, actions from any kind of ice features. This includes detection, tracking, and forecasting of sea-ice, ridges, and icebergs. As mentioned above, and accurate description of the iceberg shape is needed to precisely forecast the iceberg drift. Empirical models can predict iceberg drift with a limited accuracy, but Hughes et al. (2014) argue that the submerged part of the iceberg is one of the two main sources of uncertainty when determining
iceberg properties. Thus, *in situ* measurements of the keel geometry is beneficial for iceberg forecasting.

Autonomous underwater vehicles (AUVs) are a class of underwater vehicles, characterized by having relatively high spatial and temporal coverage, that are unaffected by the potentially harsh surface conditions in the Arctic. Norgren and Skjetne (2014) discusses the use of AUVs as sensor platforms for ice monitoring applications, and concludes that AUVs are suitable for ice-monitoring, under the requirements of available infrastructure, and further research within the field of autonomy. By mounting suitable payload sensors on the AUV, e.g. multibeam echosounder (MBE), the AUV can acquire high accuracy data of the underwater geometry of iceberg. For real-time monitoring applications, the collected data must be compressed to allow feasible transmit over a hydroacoustic transmission channel. This way, key parameters of the iceberg can be estimated real-time, for further use in online decision support systems in an IM operation.

The contributions made through this paper is an edge-detection algorithm for an iceberg with steep sides, as well as an outline of a guidance system for detecting and following the edge of the iceberg. A discussion of different iceberg mapping strategies will also be presented.

**ICEBERG MAPPING USING AUVS**

*Previous work on iceberg mapping using AUVs*

A previous experiment aiming to map icebergs by means of an AUV is presented by Forrest et al. (2012). The authors present four AUV missions conducted under a fragment of the Petermann Ice Island, at a depth of 60 m. During these missions, draft measurements from the underside of the iceberg was captured using an interferometric sidescan sonar. Multibeam measurements of the side of the iceberg keel was also captured, from a ship-mounted MBE, seen in Figure 1. Forrest et al. (2012) states that one of the biggest challenges encountered was the ability to plan missions for a drifting reference frame, indicating the need for an autonomous mapping scheme.

Kimball and Rock (2014) presents a method to estimate the AUV trajectory in a iceberg-fixed reference frame (non-inertial frame, translating and rotating with the iceberg), and mapping of
MBE measurements into this reference frame to construct a 3D map of the keel of an iceberg. The presented algorithm utilizes relative velocity information between the iceberg and the AUV, captured from a side-mounted Doppler velocity logger (DVL), and are able to estimate the position and heading of the iceberg, via post-processing. An extension of the algorithm presented by Kimball and Rock (2014) is detailed by Hammond and Rock (2014), where the inertial position of the vehicle is removed from the algorithm. This way, the map is constructed solely in iceberg-relative terms, removing potential AUV position errors from the geometry estimation. The position of the iceberg is, however, not estimated, which is valuable information for IM purposes. Another limitation of the above work is the requirement for self-intersecting paths – it is not possible to map parts of the iceberg to estimate the translation and rotation. Zhou et al. (2014) presents an algorithm for profiling an iceberg by performing a downward spiral movement around the iceberg, using an AUV with a side-mounted scanning sonar. The algorithm assumes that the waterline profile of the iceberg is known at the commencement of the survey, and a new segment of the spiral is generated at each circumnavigation based on the iceberg profile from the last round.

**Challenges**

As mentioned above, the main challenge with mapping of icebergs with AUVs are operating with a drifting reference frame. While it is possible to measure the translation of the iceberg with DVL, the rotation of the iceberg remains unknown. Mean iceberg translational velocities of 0.2 m/s and maximum velocities of 1 m/s, as well as rotational velocities as high as 90°/h is reported by Yulmetov and Løset (2014), which will cause considerable warping in the constructed 3D map, if not corrected.

Another challenge with mapping of icebergs comes from navigating AUVs at Arctic latitudes. At these latitudes, the magnetic inclination is close to vertical, and the horizontal force from the earth’s rotation is close to zero – rendering magnetic compasses and gyrocompasses unstable (McEwen et al., 2005). Gyroscopes can still be used as heading reference, but these instruments will drift with time without external corrections. In addition, the deep waters in Arctic will render DVL for bottom-tracking purposes useless – severely limiting the dead-reckon capabilities of the AUVs navigation system.

**STRATEGIES FOR ICEBERG MAPPING**

To be able to develop an autonomous guidance algorithm for monitoring of icebergs, we must first determine an appropriate mapping scheme. Through this section we will discuss three different mapping schemes, which can be divided into two main categories: mapping from the side of the iceberg, and mapping from below.

**Scanning of the iceberg keel side**

An illustration of an AUV mapping the side of an iceberg keel can be seen in Figure 2. A requirement for this mapping scheme is that the AUV performing the mapping has a side-mounted MBE, and preferably a side-mounted DVL to be able to measure relative velocity between the AUV and the iceberg. This could be a drawback, since most AUVs are designed with the DVL looking down (and many also have an upwards-looking DVL), and changing the configuration of the DVL could be a time-consuming process, since the DVL is often integrated with the
inertial navigation system (INS).

An advantage with the mapping scheme shown in Figure 2 is that designing a guidance algorithm for circumnavigating the iceberg can be done using existing control algorithms, once the iceberg has been detected. Since the iceberg can be seen as a surface, and the AUV is flying at an “altitude” above this surface (the cross-track distance between the iceberg and the AUV), an altitude controller can be used for maintaining a constant cross-track distance between the iceberg and the AUV. The altitude controller can be implemented by utilizing sonar ranges, or by using the DVL range measurements, as presented by Dukan and Sørensen (2014).

It is possible that the AUV might still hit the iceberg using this “altitude” control scheme if the iceberg shape is non-convex, and has a sharp edge. This will be similar to ordinary altitude control where the AUV runs into near vertical wall – it does not have time to react to the change in altitude before hitting the bottom. To avoid this, a forward-looking obstacle avoidance sonar could be utilized. A forward-looking sonar would also be beneficial when the location of the iceberg is unknown and the iceberg must be detected.

**Mapping of the underside of the iceberg**

Mapping of the underside of the iceberg is preferable for an IM operation, since we are able to utilize the same AUV sensor suite for mapping the underside of the sea-ice, and other ice features. Also, several AUVs comes with upwards-looking DVL (e.g. the Gavia, Iver2, and REMUS AUVs), which will complement the data-set with iceberg-AUV relative velocity – useful for post-processing of data. Through this section, two different mapping schemes will be proposed.

**Edge-to-edge mapping:** Figure 3 illustrates a strategy for mapping the underside of an iceberg. The AUV detects it has located an iceberg (by analyzing the MBE measurements), and keeps the same heading until it cannot detect the iceberg any longer. Now the AUV changes its heading by some angle (e.g. 120 degrees) towards the iceberg. The AUV detects the iceberg when passing the edge, and continues until the next edge is located, and so on. Following this strategy, a boundary box of the iceberg can be calculated (by storing the locations of the edges, as seen in Figure 3). To achieve full coverage, a lawn-mower pattern could be performed after a sufficient number of edge-points have been detected.

It must be noted that for this strategy to work, the translation and rotation of the iceberg must be estimated online. The detected edge-points must be fixed to the moving iceberg frame, any planned paths (e.g. lawn-mower pattern) must also be fixed to this frame. In the work conducted...
by Kimball and Rock (2014) and Hammond and Rock (2014), the translations and rotations of the iceberg is estimated offline, and requires the paths to be self-intersecting. Thus, estimating these parameters online is not a trivial task. Furthermore, this scheme also requires using the AUVs estimated position and orientation – information with potentially degraded accuracy at Arctic latitudes.

**Edge-following:** An Arctic iceberg is a piece of a glacier, which has calved into the ocean, and come in many sizes and shapes. For the mapping strategy presented here we will assume that the edge of the iceberg have a relatively steep edge (as the iceberg seen in Figure 1). The proposed strategy is to use MBE measurements to detect the large gradient in the range measurements present when measuring along the edge. Since we want to map as much of the iceberg as possible, and since we want to measure the relative velocity between the iceberg and the AUV with the DVL, the AUV should travel parallel to the edge, but offset a distance towards the center of the iceberg.

Since the AUV is detecting and following the edge of the iceberg, it is not necessary to estimate the iceberg translation and rotation for guidance purposes, and the strategy will not be affected by iceberg drift. A complete coverage of the iceberg can not be guaranteed with this mapping, since the size of the iceberg may be larger than the MBE swath width can cover with one circumnavigation, but the main particulars of the iceberg may be extracted from this data. Additionally, the guidance algorithm will not be affected by a degradation of the AUVs navigation system, since the objective is to follow a path constructed online from MBE measurements.

Before an iceberg is detected, the AUV runs a risk of hitting a deep-drafted iceberg. The AUV must therefore either run at a safe depth, dependent on the expected iceberg size in the operational area, or an obstacle avoidance algorithm must be employed. Obstacle avoidance may be the safest approach, since we cannot guarantee that there are no icebergs with keels running deeper than the estimated maximum draft. On the other hand, obstacle avoidance introduces
additional complexity to the guidance system, and is outside the scope of this paper.

**ICEBERG EDGE-PROFILE DETECTION AND FOLLOWING**

This paper focuses on the edge-following mapping scheme presented in the previous section. Two different guidance schemes must be employed for autonomous detection and mapping of icebergs – one for the case where no icebergs has been detected, and one for the case where the iceberg has been detected. Similarly, the case where the iceberg has been detected can be divided into three sub-cases – one where the edge has been found, one for locating the edge before the edge following algorithm commences, and one for relocating a lost edge. The overall guidance algorithm can thus be described as a state machine, with two overall states, and three sub-states. The state machine for the guidance algorithm is illustrated in Figure 4.

The AUV follows a predefined search plan when no iceberg has been located. This could for
Figure 5. AUV with MBE running under an iceberg edge.

Figure 6. MBE measurements from running under an iceberg edge (noise free). The green line illustrates a piecewise linear regression function fitted to the measurements.
example be a lawn-mover pattern in the area we are interested in. An iceberg is detected when the ice draft is above a specified threshold. Since acoustic measurements tend to be noisy, at least $N \geq 2$ beams should show an ice draft above the threshold value before signaling that an iceberg has been detected. This is to avoid false detections, hence to avoid unnecessary searches for the iceberg edge.

**Edge detection**

By assuming a step gradient in the range measurements at the edge of the iceberg (i.e. the walls of the iceberg are vertical, as seen in Figure 1), the edge of the iceberg can be detected as the point where the large gradient starts. As can be seen from Figure 5 and Figure 6, the edge of the iceberg is clearly identifiable from the MBE measurements.

To extract the edge-point from the (potentially noisy) measurements, piecewise linear regression is used on the data (see e.g. Malash and El-Khaiary (2010)). We assume that the points extracted from the range measurements from the MBE can be fitted to 3 lines. That is:

$$F(x, \theta_1) = \begin{cases} f_1(x) = a_1 + b_1 x, & x \leq c_1, \\ f_2(x) = a_2 + b_2 x, & c_1 \leq x \leq c_2, \\ f_3(x) = a_3 + b_3 x, & x \geq c_2, \end{cases}$$  \tag{1}

where $a_k$ and $b_k$ are the intersect and slope of the $k^{th}$ linear segment, respectively. $c_k$ is the breakpoint between the $k^{th}$ segment and $k + 1^{th}$ segment, and $\theta_1$ is a column vector containing all the parameters of (1). Figure 6 illustrates piecewise linear regression on MBE measurements, and the parameters $a_1, a_3, c_1, \text{ and } c_3$ are shown in this figure. Equation 1 can be rewritten as:

$$F(x, \theta) = \begin{cases} f_1(x) = a_1, & x \leq c_1, \\ f_2(x) = a_1 - c_1 b_2 + b_2 x, & c_1 \leq x \leq c_2, \\ f_3(x) = a_1 - c_1 b_2 + c_2 b_2, & x \geq c_2, \end{cases}$$  \tag{2}

by solving $f_k(c_k) = f_{k+1}(c_k)$ for $a_{k+1}$ for the two breakpoints. We have also assumed that the surface and the bottom of the iceberg is flat, resulting in $b_1 = b_3 = 0$. The new parameter vector is $\theta = [a_1 \ b_2 \ c_1 \ c_2]^T$. We can use nonlinear least squares optimization to fit the MBE measurements to the model in Equation 2. That is:

$$\min_{\theta} J(x, \theta) = |Z(x) - F(x, \theta)|^2, \ s.t \ c_2 > c_1,$$  \tag{3}

where $\theta$ contains the variables to be estimated by the least squares optimization, and $x, Z(x) \in \mathbb{R}^k, k = 1, \ldots, N_{\text{beams}}$ is the cross-track distance and the depth of each beam, respectively. The notation $| \cdot |$ represents the Euclidean vector-norm. The line between the breakpoints will mark
the transition between the iceberg and the surface, and by inserting $x = c_1$ and $x = c_2$ into the second equation in (2), and choosing the maximum draft ($\max(f_2(c_1), f_2(c_2))$), we find the cross-track error to the edge:

$$e = -(x_{ct} + s\epsilon)$$

where $x_{ct}$ is the chosen breakpoint, and $\epsilon$ is used to offset the AUVs trajectory towards the center of the iceberg, and $s \in [-1, 1]$ shifts the offset depending on if the iceberg is to the left or to the right of the AUV. This is to make sure we map as much as possible of the iceberg, and to always have measurements of the AUV - iceberg relative speed from DVL measurements. Since the swath-width of the MBE will vary with the distance from the AUV to the iceberg, the offset should ideally have been calculated from this distance, but this is considered outside the scope of this paper.

The procedure described above has been tested on noisy range measurements from a simulated MBE. A figure showing the calculated line segments, as well as the noisy measurements, can be seen in Figure 7. From the figure it is clear that the algorithm is able to detect the edge of the iceberg, even with very noisy measurements. The distance from the estimated edge and the actual edge is less than 2 meters.

A search algorithm is necessary if the edge is not detected once the edge-search is started. A reason for the edge not being detected, may be that the AUV has moved too far away from the edge towards the center of the iceberg before the edge-search algorithm was invoked. In this paper, the search algorithm is an outward spiral, due to its simplicity.
It should be noted that the edge-detection algorithm presented in this section will only work under the assumption that the walls of the iceberg are near-vertical. If the AUV encounters a rounded iceberg, which is not uncommon, this edge-detection algorithm will not work. The edge-detection algorithm was chosen to withstand noisy range measurements, and to demonstrate a line-of-sight approach to edge-following, which is presented in the following section.

**Edge following**

The edge detection algorithm relies on nonlinear least squares optimization, which is an iterative search method. This method is vulnerable to local minima, and may not converge to a solution. To increase the robustness to noisy edge detections and potential non-convergence in the edge detection algorithm, the edge angle is calculated by performing linear regression over the last \( M \geq 2 \) number of edge-detections (thus the edge-following algorithm will not be invoked before \( M \) successful edge detections has been performed). The parameters of the general line equation \( y = a + bx \) can be calculated by linear least-square regression, see Weisberg (2014, Chapter 2) for details. From the line equation we can find two points along the line, \((x_1, y_1)\) and \((x_2, y_2)\), and the line angle can now be calculated from:

\[
\alpha = \text{atan2}(y_2 - y_1, x_2 - x_1)
\]  

(5)

Having the edge line angle we can employ the integral line-of-sight guidance scheme for calculating the desired heading of the vehicle (Fossen, 2011):

\[
\chi_d(e) = \alpha + \arctan \left( -K_p e - K_i \int_0^t e(\tau) d\tau \right),
\]  

(6)

where \( e \) is the cross track error to the edge (see section on edge detection), and \( K_p \) and \( K_i \) are the proportional and integral gain of the guidance controller, respectively. The integral term is necessary to guarantee convergence to the desired path in the presence of ocean currents.

If the edge detection algorithm fails to detect the edge for a certain time, the guidance scheme will revert to the relocate edge algorithm. The AUV will lose the edge if the iceberg edge changes with 90 degrees or more, and by assuming that the iceberg is convex, we relocate the edge by turning in the direction of the iceberg by some angle (e.g. 120 degrees) and keeping this heading until the edge is relocated. If the iceberg is not detected for an extended period of time the guidance algorithm should abort the search and continue with its pre-programmed mission.

**SIMULATIONS**

A six degree-of-freedom AUV simulator is implemented in MATLAB® and Simulink®, containing a beam range simulator implemented in C++ and compiled as a Simulink S-function for efficiency. Similarly, the guidance system presented through this paper is also implemented as a C++ S-function. The guidance algorithm state machine is implemented using the Boost Statechart library (Boost, 2015), and the Eigen C++ library is used for linear algebra, and matrix and
Figure 8. AUV trajectory during mapping of an iceberg (stationary iceberg, ideal range measurements).

Figure 9. AUV trajectory during mapping of an iceberg (linear drift, ideal range measurements).

vector representation (Guennebaud, 2015). For solving the nonlinear least square optimization problem in the edge-detection algorithm, shown in (3), the ALGLIB library is used (Bochkanov, 2015).

A simulation of the complete guidance system can be seen in Figure 8. The simulation shown in this figure serves as a base, since the iceberg is assumed stationary and ideal range measurements are used. The shape of the iceberg used in the simulations presented in this paper is a rectangular shape, with slightly sloped walls (a section of the iceberg can be seen in Figure 5).

As can be seen from Figure 8, the AUV are able to detect and completely circumnavigate the iceberg, but overshoots at the corners. At the corners, the AUV loses the edge, and the relocate edge algorithm is invoked. This causes a small section of the edge of the iceberg to remain undetected. It could be argued that a more sophisticated commencement of the edge-following algorithm could be performed once the edge has been detected. This is outside the scope of this paper.

Figure 9 shows a simulation of a mapping of the same iceberg, but in this case the iceberg is affected by ocean currents. The iceberg drift direction is set to 45 degrees, and the drift velocity is set to 1 m/s. From Figure 9 we see that the AUV circumnavigates the iceberg, and the warping of the data if not corrected for drift is clear. In this case the iceberg is simulated with a constant...
linear drift, and no rotation is assumed. This is not a valid assumption for real icebergs, since
icebergs may rotate significantly (Yulmetov and Løset, 2014). However, as the range and map
simulator, in its current implementation, does not account for rotation of the map, the objective
here was only to verify the performance to linear drift. Extending the simulator to account for a
rotating iceberg will be considered in the next future step.

CONCLUSION
This paper has dealt with autonomous detection and edge-following of icebergs. Different strate-
gies for mapping of a drifting iceberg has been discussed, and a guidance algorithm has been
developed. The proposed edge-detection algorithm utilizes MBE measurements for locating
the edge of the iceberg – allowing the AUV to circumnavigate the iceberg using a line-of-sight
guidance approach. A simulation study demonstrates the guidance algorithms performance, and
ability to handle a constant linear drift of the iceberg.

Further work include calculating the offset distance based on the height from the AUV to the
iceberg, and implementing a more sophisticated edge search algorithm. The algorithms perform-
ance when mapping rotating icebergs must also be investigated. The edge-detection algorithm
used in this paper is based on the assumption that the walls of the iceberg are near vertical, and
will fail in the case where the iceberg is rounded. The edge detection algorithm was chosen
in an attempt to achieve some robustness to noisy range measurements, but the nonlinear least
squares solver encountered problems with relatively low noise levels. Due to these problems,
the edge detection algorithm should be redesigned in future extensions of this work.

ACKNOWLEDGEMENTS
This work was supported by the Research Council of Norway (RCN) through the Centre of Ex-
cellence AMOS (RCN-project 223254), and the Centre for Research-based Innovation SAM-
CoT (RCN-project 203471). The authors would also like to acknowledge Sigurd Holsen for his
contributions to the range and beam simulator through his master-project.

REFERENCES
Barker, A., Sayed, M., and Carrieres, T. 2004. Determination of iceberg draft, mass and cross-
sectional areas. In Proceedings of 14th International Offshore and Polar Engineering Con-
ference. Toulon, France, pages 899–904.
libs/statechart/.
Dukan, F. and Sørensen, A. J. 2014. Sea floor geometry approximation and altitude control of
006.
Eik, K. 2008. Review of Experiences within Ice and Iceberg Management. Journal of Naviga-
Forrest, A. L., Hamilton, A. K., Schmidt, V. E., Laval, B. E., Mueller, D., Crawford, A. J.,
in the Canadian High Arctic. In 21st IAHR International Symposium on Ice. Dalian, China,
pages 710–721.


