



Autonomous Aerial Ice Observation for Ice Defense

Joakim Haugen Lars Imsland

Department of Engineering Cybernetics, Norwegian University of Science and Technology, N-7491 Trondheim, Norway. E-mail: {joakim.haugen, lars.imsland}@itk.ntnu.no

Abstract

One of the tasks in *ice defense* is to gather information about the surrounding ice environment using various sensor platforms. In this manuscript we identify two monitoring tasks known in literature, namely *dynamic coverage* and *target tracking*, and motivate how these tasks are relevant in ice defense using remotely piloted aircraft systems (RPASs). An optimization-based path planning concept is outlined for solving these tasks. A path planner for the target tracking problem is elaborated in more detail and a hybrid experiment, which consists of both a real fixed-wing aircraft and simulated objects, is included to show the applicability of the proposed framework.

Keywords: Automated guided vehicles; dynamic coverage control, ice observation; multitarget tracking.

1 Introduction

The recent retreat of polar ice may make exploration and development of offshore petroleum in cold regions a reality in the future. An increased human intervention in cold regions will involve the presence of vessels carrying out various types of tasks such as: lifting, installation, crew change, evacuation, maintenance, and drilling (Gürtner et al., 2012). These tasks often require the relevant vessels to perform stationkeeping, that is, remain at a fixed location, or more generally, to be dynamically positioned (DP), for instance slowly maneuvering close to an offshore installation (Fossen, 2011, Ch. 6). Pioneering full-scale experiments of DP in ice (see e.g. Moran et al. (2006)), together with similar operations, such as iceberg detection and tracking (Eik, 2008), have learned that a wide range of supporting activities are essential for responsible and safe offshore operations in cold regions.

In this paper, we will make a case for the use of remotely piloted aircraft systems (RPASs) for ice monitoring for ice defense as one such supporting activity. We start by describing ice defense, with emphasis on

the required information gathering and the tasks that can be solved by mobile aerial sensors, namely *target tracking* and *dynamic coverage*. We continue to discuss the use of RPAS in cold regions and how they can be used to solve these tasks with main attention to target tracking. The remainder of the paper is a case study in autonomous path planning for aerial sensors for the target tracking problem, with a hybrid experiment, which consists of both a real fixed-wing aircraft and simulated objects, that illustrates the approach.

1.1 Notation

An n -dimensional column vector of ones is denoted $\mathbf{1}_{n \times 1}$. I_n is the $n \times n$ identity matrix. A countable finite index set of positive natural numbers is defined as $\mathbb{I}_n := \{i \in \mathbb{N}^+ : i \leq n\}$. A block diagonal matrix of other matrices $X_{i \in \mathbb{I}_s} \in \mathbb{R}^{m_i \times n_i}$ is defined as $\text{bdiag}_{i \in \mathbb{I}_s}(X_i) := \bigoplus_{i \in \mathbb{I}_s} X_i$, where \bigoplus is the direct sum. The vertically stacked matrix of other matrices $X_{i \in \mathbb{I}_s} \in \mathbb{R}^{m_i \times n}$ is denoted $\text{col}_{i \in \mathbb{I}_s}(X_i) := \text{bdiag}_{i \in \mathbb{I}_s}(X_i) \cdot (\mathbf{1}_{s \times 1} \otimes I_n)$, where \otimes is the Kronecker product. The diagonal of a column vector $x \in \mathbb{R}^n$ is the block diagonal of its scalar ele-

ments x_i , $\text{diag}(x) := \text{bdiag}_{i \in \mathbb{1}_n}(x_i)$. The space of non-zero n -dimensional real vectors is denoted $\mathbb{R}_{\neq 0}^n$. Define the set of positive definite real matrices as $\Pi_n := \{A \in \mathbb{R}^{n \times n} : \forall x \in \mathbb{R}_{\neq 0}^n, x^\top A x > 0\}$. The orientation space is defined by $\mathbb{S} := [-\pi, \pi)$. The first moment of a random vector x is denoted by the expectation operator $\mathbb{E}(x)$. The covariance matrix of two random vectors x and y is defined as $\text{cov}(x, y) := \mathbb{E}[(x - \mathbb{E}(x))(y - \mathbb{E}(y))^\top]$. Non-negative real numbers are defined by the set $\mathbb{R}_{\geq 0}$. A zero-mean continuous-time white noise process $w(t)$ of dimension n has the properties $\mathbb{E}(w) = 0$ and $\text{cov}(w(t), w(\tau)) = Q(t)\delta(t - \tau)$, where $Q : \mathbb{R}_{\geq 0} \rightarrow \Pi_n$ is the deterministic spectral density and $\delta(t)$ is the dirac delta function. The above mentioned properties of $w(t)$ are written compactly as $w(t) \sim (0, Q(t))$.

2 Ice Defense

[Arctic Marine Solutions \(2014\)](#) uses *ice defense* as the aggregate term for supporting activities involved in cold regions marine operations such as Arctic dynamic positioning. Important activities in ice defending include ([Eik, 2008](#); [Keinonen, 2008](#)):

- Protecting the DP vessel or structure from hazardous ice through physical ice management (e.g. ice breaking and/or iceberg towing).
- Gathering and processing information for decision support.
- Decision-making such as operational threat assessment and strategies.

A utopian objective of ice defense operations includes creating a complete ice features awareness map of a vast spatial region and maintaining it continuously over time. This is neither economically nor practically feasible, so prioritization of the information gathering is needed. The region surrounding the DP operation is often divided into different zones depending on the estimated time of arrival (ETA) of the drifting ice ([Sheykin, 2010](#); [Edmond et al., 2011](#)). In the various zones, distinct monitoring objectives apply with different required level of urgency and detail. We divide the region into three conceptual surveillance zones:

Far-field zone. Here, we execute regional surveillance/coverage for detection and classification of hazardous ice features such as icebergs, ice ridges, and ice cover type. This information is crucial for threat assessment and operation planning. 1–7 days upstream.

Mid-field zone. This is the area within which ice features may reach the region where sea ice physical management is finding place. Ice identification (of for instance ice drift dynamics, ice concentration, and ice thickness) is important for operational efficiency, such as the choice of both ice breaking

strategy and tactics ([Hamilton et al., 2011](#)). 6–24 hours upstream.

Close-field zone. The region of ice that most likely will reach the DP vessel. Detailed information about the ice feature geometry, ice thickness, ice concentration, ice drift velocity, and more may all be important for good DP performance ([Metrikin et al., 2013](#)). Up to a few hours upstream.

The zones have different extents and no single sensor platform is able to perform all the monitoring tasks by itself. [Eik and Løset \(2009\)](#) motivates unmanned underwater vehicles (UUVs) as a tool in collecting ice intelligence for ice defense, whereas [Haugen et al. \(2011\)](#) motivates unmanned aerial vehicles (UAVs) to do the same together several other sensor platforms in a collaborative effort, see Figure 1.



Figure 1: Illustration of a possible future Arctic dynamic positioning operation that consists of many important components, including icebreakers, unmanned underwater vehicles, and unmanned aerial vehicles. Picture courtesy of Bjarne Stenberg.

The sensor platforms constitute an ice observation system, which is an integral part of both the ice defense and the dynamic positioning. Since the resources are constrained with respect to cost, physical, and practical considerations, some kind of high-level task allocation procedures need to choose the appropriate sensor platforms for the required monitoring tasks. This task allocation happens both before and during the operation execution:

In the planning phase, decisions such as the choice of immobile sensors, type and number of mobile sensor platforms are made. These choices are made based on the required level of information and redundancy for the particular operation in question.

During the operation *dynamic mission planning* occurs, which involves allocating needed tasks to the

fleet of singular or teams of mobile sensor networks.

The dynamic planning allows for low response times in deployment and may facilitate real-time acquisition of information, so that important operational decisions can be made. There is a wide range of monitoring missions that may be needed in ice defending. With respect to the mid-field and far-field zones, possible missions for remotely piloted aircraft systems (RPASs) are:

Mission 1. *Iceberg and ice ridge detection, identification, and tracking.*

Mission 2. *Sea ice identification and dynamic coverage.*

2.1 Remotely Piloted Aircraft Systems in Cold Regions

A remotely piloted aircraft system (RPAS) is a subgroup of the more general category of unmanned aircraft systems (UASs). The RPAS consists of all the components needed to operate such a system: one or several UAVs, a ground station including the pilot station, launch and recovery systems, communication equipment, and more (CAA–Norway, 2014). An RPAS may be autonomous in the sense that the system can make its own decisions during the course of operation execution, but with the restriction that an operator can intervene and remotely pilot the UAVs.

The inclusion of RPASs in cold regions comes with a whole range of challenges that need to be addressed. Common keywords when talking about operations in Arctic regions are remoteness, darkness, and low temperatures. Robustness against these attributes is very important and may include sophisticated launch and recovery systems from ships (Crowe et al., 2012), robust communication systems (Frew and Brown, 2009), and fault-tolerant guidance, navigation, and control (GNC) systems. Other aspects that need to be addressed include (Haugen et al., 2011; Crowe et al., 2012): icing problems, vibration issues, water intrusion, and airspace access. Norwegian research communities working with problems connected to RPAS in cold regions include AMOS (2014); Norut (2014); Simicon (2011).

Apart from GNC, one aspect of an autonomous aircraft system is its ability to create paths that help in solving some monitoring task. In a cold regions operation setting, one can think of many different tasks that need a specialized system that can perceive its environment and make intelligent decisions based on the observations. In this manuscript, we briefly discuss the following ice observation sub-tasks: *target tracking*, re-

lated to Mission 1, and *dynamic coverage control*, related to Mission 2. We provide an overview of common components that have been used to solve the mentioned sub-tasks in Haugen and Imsland (2014b) and Haugen and Imsland (2014a), respectively. We will finally display results from a hybrid field experiment of the *target tracking* task, which was first reported in Haugen and Imsland (2013b).

2.1.1 Target Tracking

Suppose a set of possibly hazardous icebergs and ice ridges has been detected in the far-field zone with the use of satellites. The acquired satellite data are of too coarse resolution to provide conclusive answers (Eik, 2008; Haugen et al., 2011). To confirm/refute the possible hazards, current practices involve manned aircraft (Eik, 2008) and reconnaissance vessels (Sheykin, 2010). We motivate the use of UAVs as a tool which may reduce costs and environmental footprint when solving this monitoring task. The task is formulated as a *target tracking* objective and approached by assuming that a small number of UAVs is dispatched to remotely gather more information before further actions are taken.

The target tracking problem is the task of monitoring mobile or immobile objects using (usually mobile) sensor agents. Many problems can be cast as a target tracking problem, so the literature is rich on various approaches, see Haugen and Imsland (2014b) and references therein. Haugen and Imsland (2014b) classify contributions in literature as “ n_m to n_o ” tracking, where n_m is the number of sensors and n_o is the number of objects. The above defined problem is a multi-target tracking objective, where the number of mobile sensor agents are possibly more than one.

In solving the target tracking problem, contributions usually choose between two main methodologies: resource allocation and information-driven approaches. In the resource allocation problem, the targets are prescribed to be visited a predefined number of times. It is formulated as a modified traveling salesperson problem (Rathinam et al., 2007; Looker, 2008; Savla et al., 2008), often taking the limited turning radius of the mobile sensor into account. Unlike resource allocation, information-driven methods define the visitations of the targets according to some information reward. Information gradients are usually utilized in the formulation of optimization problems, which are seeking to minimize measures related to the information level, either minimize time between target measurements (Tang and Özgüner, 2005), maximize observation time (Parker, 1999), or minimizing the targets’ estimation error covariance (Haugen and Imsland, 2013a,b, 2014b; Morbidi and Mariottini, 2013).

2.1.2 Dynamic Coverage Control

Imagine you want to get an awareness map of a bounded region in the mid-field zone. When creating this awareness map, one task may be to get more detailed information about relevant ice conditions, for instance the ice concentration, which is the area fraction of ice versus open water. Current approaches include using satellites, reconnaissance vessels, and marine radars (Sheykin, 2010). We propose to use UAVs to cover the region of interest. Since the ice has a drifting velocity, the task can be formulated as a *dynamic coverage problem*.

Wang and Hussein (2012) describes the dynamic coverage problem as the problem of covering a given region using mobile sensor networks. The desired information to be gathered changes in both time and space, so a non-dynamic coverage algorithm may not be sufficient to capture the information with the required level of accuracy. The monitoring task is therefore to perform state estimation of some distributed parameter system (DPS), usually described as a partial differential equation (PDE).

Previous work on state estimation dynamic coverage falls under two main approaches: optimal control formulations (Burns et al., 2009; Choi and How, 2010; Haugen and Imsland, 2014a), and gradient-based guidance algorithms (Demetriou and Hussein, 2009; Demetriou, 2010; Haugen et al., 2012). In the optimal control problem (OCP) formulation, one seeks to minimize some kind of objective functions that quantify the information reward of visiting a particular region. The formulations often use a measure based on the estimation error covariance dynamics (Burns et al., 2009; Choi and How, 2010), but to facilitate computational speed, simplified information dynamics has also been proposed (Haugen and Imsland, 2014a). Gradient-based guidance algorithms are more computationally efficient, but often rely on locally available estimation error to guide the vehicles (Demetriou and Hussein, 2009; Demetriou, 2010; Haugen et al., 2012).

2.2 Optimization-Based Path Planning

We approach the two tasks identified as target tracking and dynamic coverage control by considering two separate monitoring systems. Both systems have three common components:

RPAS: Remotely piloted aircraft system that acts as a mobile sensor network. The mobile sensors provide measurements of the environment. Depending on the which task is being solved, they provide measurements of objects or a distributed process.

Estimator: Prepares collected measurements and other inputs to return the most likely model state

and parameters of the mobile sensors and the observed environment.

Path Planner: Generates guidance inputs of when and where we want the mobile sensor to obtain measurements of the environment.

Figure 2 displays an overview of the system components. The *Supervision* component has the role as a dynamic mission planner. It is responsible for deciding how many mobile sensors to deploy, which region or objects to cover, and other high-level decisions.

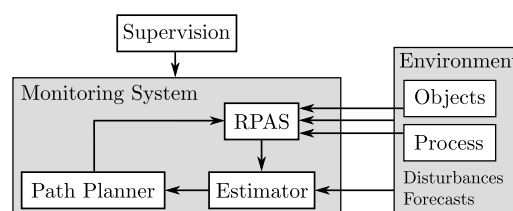


Figure 2: Key components of the monitoring systems used to solve the target tracking and dynamic coverage control problems.

The *Path Planner* component solves a dynamic optimization problem motivated by the information-driven approach. It contains mathematical descriptions of various modules that are needed to create meaningful guidance inputs (cf. Figure 3). The mobile sensor agents are described using simple ordinary differential equations (ODEs) with maneuverability constraints. The mobile sensors need to satisfy collision avoidance requirements and stay within permitted regions of operation. The objective of the path planner is to manipulate the commanded input of the mobile sensors such that some reward function is maximized. The reward function depends on a description of the monitored environment’s uncertainty. The environment uncertainty can be controlled using the mobile sensor agents, which act as uncertainty manipulators. For instance, in the case of target tracking, the environment uncertainty can be described as the sum of all the object’s state estimation error covariances. The uncertainty can then be manipulated by the moving agent’s sensing capability, which reduces the uncertainty close to the agent’s location.

In this manuscript we will focus on the *Path Planner*, and in particular describe more formally the conceptual modules needed for solving the “1 to n_o ” target tracking problem. The following sections are to a large extent based on Haugen and Imsland (2013b, 2014b).

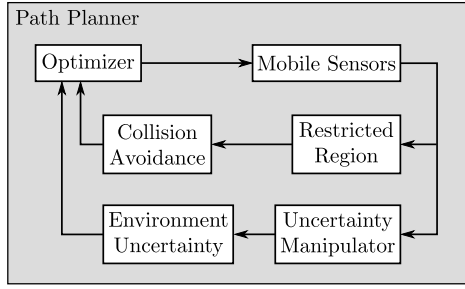


Figure 3: Conceptual modules in the path planner.

3 Path Planning for Target Tracking

Suppose that there are some objects moving with almost constant velocity that we want to keep track of. We assume that we have estimated initial positions of the objects at time t_0 , but that the objects' states are uncertain. An unmanned aerial vehicle is able to sense the objects using for instance a camera with a limited field of view (FOV). Since the FOV is limited, the objects cannot necessarily be observed simultaneously. The objective is thus to steer the mobile sensor such that all the objects' states can be estimated.

Let $\{\text{ned}\}$ denote a right-handed stationary reference frame whose axes denote north, east, and down coordinates, respectively. Define for each object in $\mathbb{I}_{n_o} =: \mathbb{O}$ the state vector $\chi_o := \text{col}(N_o, E_o)$, which are north and east coordinates. Let the initial position of object o be

$$\chi_o(t_0) = \chi_{o,0}, \quad (1a)$$

and the corresponding dynamics

$$\dot{N}_o(t) = v_{N,o} + w_{N,o}(t), \quad (1b)$$

$$\dot{E}_o(t) = v_{E,o} + w_{E,o}(t), \quad (1c)$$

where $p_o := \text{col}(v_{N,o}, v_{E,o})$ is the known velocity parameter vector, and $\text{col}(w_{N,o}(t), w_{E,o}(t)) =: w_o^\chi(t) \sim (0, Q_o(t))$ is process noise.

The states of the objects are random variables and are equipped with uncertainty measures. More specifically, $\forall o \in \mathbb{O}$ we define $\mathbb{E}(\chi_o(t)) = \hat{\chi}_o(t)$ as the state estimate of $\chi_o(t)$. For each $o \in \mathbb{O}$, we define the state estimation errors as

$$\tilde{\chi}_o(t) = \chi_o(t) - \hat{\chi}_o(t), \quad (2a)$$

which gives the estimation error covariance

$$P_o(t) = \text{cov}(\tilde{\chi}_o(t), \tilde{\chi}_o(t)). \quad (2b)$$

This matrix quantifies the uncertainty of object o .

To accommodate feasible trajectories for the mobile sensor, we consider a low-fidelity constant-altitude

kinematic vehicle model with the bank angle $u_\theta(t)$ as control input. Let $x_N(t)$ and $x_E(t)$ denote the north and east position of the sensor, and $\psi(t) \in \mathbb{S}$ the right-hand screw z-axis rotation of a body-fixed reference frame $\{\mathbf{b}\}$ relative to $\{\text{ned}\}$. Define $x(t) := \text{col}(x_N(t), x_E(t), \psi(t))$ with the initial condition

$$x(t_0) = x_0. \quad (3a)$$

The dynamics is

$$\dot{x}_N(t) = V_a \cos(\psi(t)), \quad (3b)$$

$$\dot{x}_E(t) = V_a \sin(\psi(t)), \quad (3c)$$

$$\dot{\psi}(t) = \frac{g}{V_a} \tan(u_\theta(t)), \quad (3d)$$

where the constant parameters V_a and g are the positive airspeed and the standard gravity, respectively. We constrain the commanded bank angle so that the resulting state trajectories are sufficiently conservative and feasible. For all $t \in \mathbb{R}_{\geq 0}$ and $u_{\theta,L}, u_{\theta,H} \in \mathbb{S}$ let

$$u_{\theta,L} \leq u_\theta(t) \leq u_{\theta,H}. \quad (3e)$$

We also restrict the planar position of the mobile sensor. In particular, we define a closed convex polygon $\mathcal{K} := \{y \in \mathbb{R}^2 : Ay \leq b\}$, where A and b have appropriate dimensions, so that $x(t)$ is constrained for all $t \in \mathbb{R}_{\geq 0}$ by

$$x(t) \in \mathcal{K} \times \mathbb{S} =: \mathbb{X}. \quad (3f)$$

Problem 1. Perform state estimation of the objects $o \in \mathbb{I}_{n_o}$ of (1) for all $t \in [t_0, T]$, where T is the final time of interest. This should be accomplished by determining a feasible input $u_\theta(t)$ for the mobile sensor (3) to obtain measurement intervals of all the objects.

We need to develop differential equations based on (2b) where the benefit of the mobile sensor is incorporated. The objective is to minimize the uncertainties of the objects' states to allow probable estimates of the objects future state trajectories. The task includes generating feasible trajectories for the mobile sensor. The problem will be approached by formulating and efficiently solving a receding horizon optimization problem that mathematically describes the objective. A sub-task is to find a plausible parametrization of the mobile sensor's influence on the objects' covariance dynamics.

3.1 Measurement Models

Write the two-dimensional Cartesian coordinates of the mobile sensor as $\varrho(t) := \text{col}(x_N(t), x_E(t))$. Further, $\forall o \in \mathbb{O}$ we have the object positions $q_o(t) = \chi_o(t)$. We assume that the objects are located in such a way that the mobile sensor cannot necessarily measure all

the objects simultaneously. We use sampling functions (Tricaud and Chen, 2012) that depend on the coordinates of the mobile sensor to reflect how the output vector is sampled by the sensor. In this context, the output vector is a vector function that depends on the planar position of an object being monitored.

Define the family of scalar sampling functions as \mathbb{W} and let \mathbb{B} be the codomain of this family. Let $w : \mathbb{R}_{\geq 0} \times \mathbb{R}^2 \times \mathbb{R}^2 \times \mathbb{S} \rightarrow \mathbb{B}$ and define a diagonal matrix function $W(t, \varrho(t), q(t), \psi(t)) = w(t, \varrho(t), q(t), \psi(t)) \times I_2$ with codomain $\in \mathbb{B}^{2 \times 2}$. The shaped measurement vector is therefore defined as

$$y_w(t) = W(t, \varrho(t), q(t), \psi(t)) y(t). \quad (4)$$

In our particular application we assume that the mobile sensor is capable of measuring an object's position, so the noise-free output function is for each $o \in \mathbb{O}$

$$y_o(t) = C_o q_o(t), \quad (5)$$

where $C_o = I_2$. The role of the sampling function is to *shape* the output function connected to an object. The output is switched off (set to zero) whenever the object is outside the measurement reach of the mobile sensor. A shaped measurement vector therefore captures the case where a measuring device has a limited field of view, for instance an image obtained from an optical device.

3.1.1 Non-Smooth Sampling Function

The purpose of this model is to simulate that the measuring device has a field of view in which it is able to obtain measurements. This includes for instance the cases of roll and pitch stabilized downward-looking optical devices and spectrometers.

Let $\Delta_{z_i} > 0$, $i \in \{1, 2\}$ and define $\Delta_z := \text{col}(\Delta_{z_1}, \Delta_{z_2})$, with $z = \text{col}(z_1, z_2)$. We define the two-dimensional FOV metric as a weighted infinity norm

$$\|x\|_{\text{FOV}, \Delta_z} := \max\left(\frac{|z_1|}{\Delta_{z_1}}, \frac{|z_2|}{\Delta_{z_2}}\right). \quad (6)$$

Suppose the position $\varrho(t)$ of a sensor is the origin of a body-fixed Cartesian coordinate system $\{b\}$. Further suppose the orientation $\psi(t)$ of $\{b\}$ is defined relative some stationary reference frame $\{i\}$ following the right-hand rule. Let $\mathbb{B}_{C^{-1}} := \{0, 1\}$ be the codomain of a binary sampling function $w_{C^{-1}} : \mathbb{R}_{\geq 0} \times \mathbb{R}^2 \times \mathbb{R}^2 \times \mathbb{S} \rightarrow \mathbb{B}_{C^{-1}}$ such that the codomain is nonzero only if a coordinate point $q(t) \in \mathbb{R}^2$ of an object is within the convex set formed by a FOV metric. The two-dimensional rotation matrix is

$$R(\psi) = \begin{bmatrix} \cos \psi & -\sin \psi \\ \sin \psi & \cos \psi \end{bmatrix}. \quad (7)$$

We can write the binary sampling function as

$$w_{C^{-1}}(t, \varrho, q, \psi) := \begin{cases} 1, & \|R^T(\psi)(q - \varrho)\|_{\text{FOV}, \Delta_z} < 1 \\ 0, & \text{otherwise.} \end{cases} \quad (8)$$

Figure 4 graphically illustrates the behavior of the binary sampling function. We see that Δ_{z_1} and Δ_{z_2} quantify respectively the field of view in x_b and y_b direction of the body-fixed reference frame.

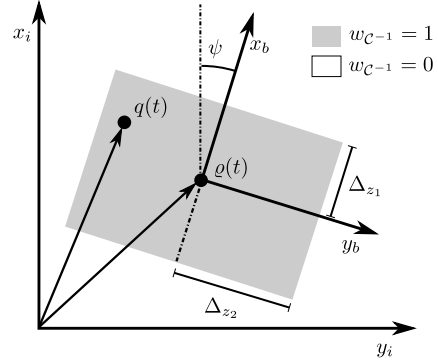


Figure 4: The non-smooth sampling function is one for $q(t)$ inside the box and zero otherwise. x_i and y_i denote the axes of the stationary reference frame.

3.1.2 Smooth Sampling Function

In some cases, for instance in an optimization problem, a continuously defined sampling function with positive codomain may be preferred as an approximation to some non-smooth sampling function. Let $\mathbb{B}_{C^\infty} := \{w \in \mathbb{R} : 0 \leq w \leq 1\}$. Define a smooth sampling function $w_{C^\infty}(t, \varrho, q, \psi) : \mathbb{R}_{\geq 0} \times \mathbb{R}^2 \times \mathbb{R}^2 \times \mathbb{S} \rightarrow \mathbb{B}_{C^\infty}$, which is 1 if $\varrho = q$ and less than 1 otherwise.

Example 1. Let $K_{i \in \mathbb{I}_n} \in \Pi_2$, and $\tilde{q} = q - \varrho$. A possible smooth sampling function is the linear combination of n two-dimensional Gaussian functions, for instance

$$w_{C^\infty}(t, \varrho, q, \psi) := \sum_{i \in \mathbb{I}_n} \lambda_i e^{-\tilde{q}^T R(\psi) K_i R^T(\psi) \tilde{q}}, \quad (9)$$

where $\sum_{i \in \mathbb{I}_n} \lambda_i = 1$, $\lambda_i \geq 0$. The purpose of having a combination of exponential functions is that the sampling surface can be shaped in such way that it approximates a non-smooth sampling surface while still having a nonzero image. Figure 5 displays an example of such a sampling surface.

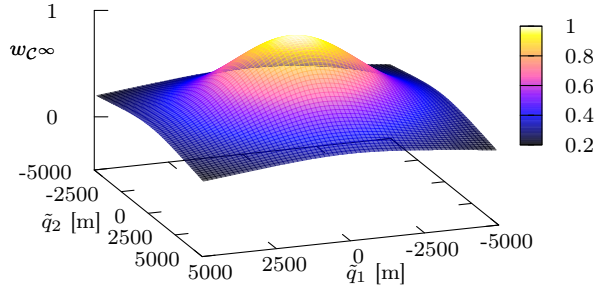


Figure 5: A smooth sampling function. The axes represent $\text{col}(\hat{q}_1, \hat{q}_2) = \tilde{q} = (q - \varrho)$, which is the relative planar coordinate between an object and a sensor.

3.2 Path Planner

3.2.1 Adapted Covariance Dynamics

The estimation error covariance (2b) of the objects quantify the uncertainties of the state vector. We want to reduce these uncertainties by measuring the objects using the mobile sensor. The covariance response of the objects can be described by the corresponding equation in the continuous-time extended Kalman filter (Simon, 2006). We present in the following a version with both time-varying process noise and measurement matrices.

To model the mobile sensor’s influence on the object covariance dynamics we make use of the measurement models presented in Section 3.1. More specifically, we use the smooth sampling function to define a diagonal matrix function W_o for the sensor’s influence on each object. We get $\forall o \in \mathbb{O}$

$$W_o(t, \varrho, q_o, \psi) = w_{C^\infty, o}(t, \varrho, q_o, \psi) \times I_2. \quad (10)$$

The motivation for using smooth sampling functions is that the chosen solver needs smoothness and curvature to find a solution to the optimization problem. Hence, non-smooth sampling functions need to be approximated using smooth sampling functions. The proposed shaping can be tuned in such way that the codomain approximates the behavior of a measuring device with a limited field of view such as a camera. The mobile sensor will significantly affect the covariance dynamics of the object if it is sufficiently close to it. The closer the mobile sensor is to the object, the bigger stabilizing impact it will have on the object’s covariance.

For each object $o \in \mathbb{O}$ let $\hat{\chi}_o(t)$ be the solution to the initial value problem (IVP) (1) with $\chi_o(t_0) = \hat{\chi}_{o,0}$ which is the best estimate at time t_0 . Then, the predicted trajectory for object o is $\hat{\chi}_o(t)$. We use the above matrix sampling functions to produce vehicle-dependent shaped versions of the measurement operators C_o of the objects along the predicted trajectories

$\hat{\chi}_o(t) = \hat{q}_o(t)$. The measurement operators of the objects are the identity matrices, so by using (10), we simply get for each object $o \in \mathbb{O}$

$$C_o^w(t) = w_{C^\infty, o}(t, \varrho, \hat{q}_o, \psi) \times I_2. \quad (11)$$

When the mobile sensor is sufficiently far away from the object, the shaping of the measurement should be so little that it in practice does not affect the covariance: the mobile sensor is not able to measure the object. For the sensor to still be attracted to distant objects, we propose to manipulate the process noise matrices $Q_o(t)$. We use a non-vanishing sampling function to reduce the process noise when a sensor is close to an object. In this way, the mobile sensor’s movement will always affect the covariance of the objects, but only slightly. Define for all objects $o \in \mathbb{O}$

$$Q_o^w(t, \varrho, \hat{q}_o, \psi) = Q_o(t) (1 - w_{C^\infty}(\varrho, \hat{q}_o, \psi)). \quad (12)$$

We use the vehicle-dependent time-varying expressions defined above to define the covariance dynamics of each object $o \in \mathbb{O}$. The differential Riccati equation of object o is

$$\dot{P}_o(t) = Q_o^w - P_o C_o^w{}^\top R_o^{-1} C_o^w P_o, \quad (13a)$$

$$P_o(t_0) = P_{o,0}, \quad (13b)$$

where we have omitted the arguments of the expressions and used that the Jacobian of the object dynamics (1) is the zero matrix.

3.2.2 Dynamic Optimization Problem

The object monitoring can be formulated as a Bolza-type OCP. Let $t_0, t_f \in \mathbb{R}_{\geq 0}$ respectively denote the start and the end of the optimization horizon. The decision variable is the control input $u_\theta(t)$. Let $P(t) := \text{bdiag}_{o \in \mathbb{O}}(P_o(t))$.

We define the Lagrange term as

$$\Phi_L(t, u_\theta) = \int_{t_0}^{t_f} \text{tr}(P(t) \text{diag}(v^L(t))) + \frac{du_\theta}{dt}{}^\top \Gamma(t) \frac{du_\theta}{dt} + u_\theta^\top \Xi(t) u_\theta dt, \quad (14a)$$

where for each $o \in \mathbb{O}$ we have $v^L(t) := \text{col}_{o \in \mathbb{O}}(v_o^L(t))$ with $v_o^L(t) \in \mathbb{R}_{\geq 0}^2$ and the scalar functions $\Xi(t) > 0, \Gamma_m(t) > 0$, which all are time-varying design variables. Notice that in the presented application u_θ is a scalar. The Mayer term is

$$\Phi_M(t_f) = \text{tr}(P(t_f) \text{diag}(v^M)), \quad (14b)$$

where $v^M := \text{col}_{o \in \mathbb{O}}(v_o^M) \in \mathbb{R}_{\geq 0}^{2n_o}$ also is a design vector.

The resulting optimization problem is to minimize (14) constrained by the mobile sensor dynamics and

the objects' covariance dynamics, that is, $\forall t \in [t_0, t_f]$, $\forall o \in \mathcal{O}$:

$$\min_{u_\theta} \Phi_L(t, u_\theta) + \Phi_M(t_f) \quad (15a)$$

$$\text{s. t. } (3), (13). \quad (15b)$$

The solution to (15) provides us with the input vector $u_\theta^*(t) \in \mathbb{U}$ in the interval $t \in [t_0, t_f]$. Given the variables u_θ^* , \hat{x}_0 , we can solve the IVP formed by (3) over the optimization horizon. This results in an optimal mobile sensor state trajectory, denoted as

$$x^*(t) \in \mathbb{X}, t \in [t_0, t_f]. \quad (16)$$

Equation (16) serves as guidance input to the aircraft.

3.2.3 Receding Horizon

Suppose we want to monitor the objects in the time interval $\mathbb{T} := [t_0, T]$. If T is sufficiently large, the optimization problem (15) becomes very difficult to solve in real time. To overcome computation time issues, we solve the optimization problem in a receding horizon fashion. This involves successively solving optimization problems with shorter time horizons. There are several factors that motivate this design decision:

- (i) Too long time horizons may make the optimization problem computationally intractable due to the problem size.
- (ii) There are modeling inaccuracies, so the predicted object trajectories may drift away from the true trajectories.
- (iii) Some conditions change, for instance a new object should be monitored.

By solving the optimization problem with receding horizons, the formulation can take into consideration updated information to improve the monitoring performance. Each time horizon overlaps with neighboring horizons. We utilize only a sub-interval of an optimized control input interval. Let $k_t \in \mathbb{N}_1$ be the number of optimization intervals and $\mathbb{T}_k := [t_{0,k}, t_{f,k}]$ be the optimization horizon for the k th iteration. For horizon k we have $t_{0,k} < t_{e,k} < t_{f,k}$, where $t_{e,k}$ is the final time for which we use the k th iteration's control input. The switch to the next iteration's control input is therefore equal to the utilization time of the preceding iteration: $t_{0,k+1} \equiv t_{e,k}$.

Consider the k th iteration of the monitoring process (see Figure 6). We divide it into a three-step procedure of collecting, optimizing, and utilizing. The first step, which is performed by the *Estimator*, involves collecting measurements of the objects' and sensor's states. At time $t_{0,k-1}$ the collected information so far is used to perform state and parameter estimation. This involves predicting the future state of the objects and sensor

at time $t_{0,k}$. The next step is to optimize by solving (15) to obtain the desired path (16). This step is accomplished by the *Path Planner*. The optimized paths should be readily available by the time $t_{0,k}$, since they at this time instant should be utilized by the RPAS, which is the final step.

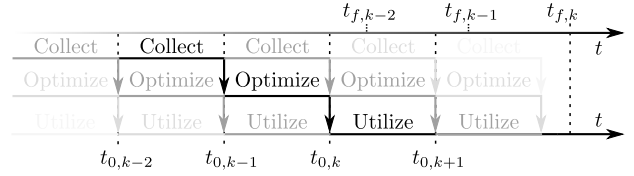


Figure 6: The monitoring system consists of a three-step procedure of collecting, optimizing, and utilizing.

The three steps of the procedure execute concurrently with earlier and later time steps: when the monitoring system is optimizing for iteration k , it is collecting for iteration $k + 1$, and utilizing iteration $k - 1$. Figure 6 illustrates the three-step procedure.

4 Case Study

4.1 Implementation

To efficiently solve the OCP (15), we choose a direct transcription approach where both the state and control variables are discretized into a finite-dimensional nonlinear programming (NLP) problem. The simultaneous collocation of finite elements is used to obtain Lagrange interpolation polynomial descriptions of the state variables. The control input is piecewise constant, whereas the states are described using K-point Radau collocation, for details consult Biegler (2010).

The resulting large-scale NLP formulation benefits from being sparse and having structure. These properties can be exploited using an efficient NLP solver. We formulate the problem in the symbolic framework CasADi (Andersson et al., 2012), which provides the necessary derivative information required by both the extended Kalman filter and the NLP solver. The CasADi library contains an interface for the primal-dual interior-point NLP solver IPOPT (Wächter and Biegler, 2006). IPOPT is compiled with OpenBLAS (Xianyi et al., 2012) and the linear algebra sparse direct solver MA27 (HSL, 2011).

When solving initial value problems, for instance when finding predict trajectories of the objects or expected closed-loop behavior of the mobile sensor, we use the ODE solver CVODES of the SUNDIALS suite (Hindmarsh et al., 2005).

An initial desired sensor path is provided a priori because it needs to be available when the first optimization is running. Once an OCP has finished, solving the mobile sensor dynamics with the provided initial condition and control inputs, results in a new optimal path. The performance of the discretized optimization problem benefits from good initial conditions. We initialize the object state and covariance variables by solving the matching IVPs with expected closed-loop behavior of the mobile sensor given its respective predicted initial condition. Since a new optimization horizon goes beyond the previous, we use the previous iterations final control input as extrapolation.

4.2 Experiment Setup

We are going to solve Problem 1, which was presented in Section 3 with $n_o = 3$. We numerically simulate the objects under surveillance and use an UAV as the mobile sensor. We assume that the UAV is capable of observing the objects with a pitch and roll stabilized downward-looking optical camera. Since no real objects are going to be observed, we simulate this behavior by providing object measurements at a sampling frequency of 1 Hz whenever the simulated object is within the field of view of $\Delta_{z_1} = \Delta_{z_2} = 300$ m using (8).

The RPAS is provided by [Maritime Robotics \(2013\)](#). It consists of a Penguin UAV B (Figure 7) from [UAV Factory \(2013\)](#) with a Piccolo autopilot and the flight management software Piccolo Command Center from [Cloud Cap Technology \(2013\)](#). This system is used with waypoint-tracking capability, so the continuous state trajectories provided by the path planner are sampled at a frequency of 1/8 Hz. The sampled north and east coordinates are combined with a constant altitude of 600 m and transformed into latitude and longitude decimal degrees (WGS84). This information is written to compatible waypoint files and manually uploaded to the aircraft autopilot by a flight operator in a timely manner. The path-planning algorithm itself is run on a laptop computer and receives aircraft telemetry data at 1 Hz from the Piccolo Command Center through a TCP/IP connection. From the telemetry data the GPS coordinates and orientation are utilized by the path planner.

The experiment was executed at Eggemoen Aviation and Technology Park, Ringerike, Norway. An extended visual line-of-sight (EVLOS) airspace was allocated, with additional field observers on a lookout for unscheduled air traffic. A nominal circular path within the confined airspace is uploaded a priori to make it easier to initialize the path planning algorithm. Once started, the path-planning framework received a mobile sensor initial condition of $\hat{x}(t_0) =$



Figure 7: Maritime Robotics’ Penguin B from UAV Factory that was used during the experiments.

$\text{col}(1621.63, -640.87, 6.00)$ at $t_0 = 0.74$ s. The bank angle is constrained to be within $[u_{\theta,L}, u_{\theta,H}] = [-\frac{\pi}{9}, \frac{\pi}{9}]$. This is more conservative than what the autopilot actually can manage, but software-in-the-loop simulations suggest better path-following performance under this constraint. The standard gravity g is set to 9.81 m s^{-2} and the airspeed V_a is 28 m s^{-1} .

The initial conditions of the objects are $\hat{\chi}_1(t_0) = \text{col}(1360, 700)$, $\hat{\chi}_2(t_0) = \text{col}(2946, -539)$, and $\hat{\chi}_3(t_0) = \text{col}(2400, 440)$, where $t_0 = 1.00$ s. The velocity parameters are $p_1 = \text{col}(-1.15, -0.96)$, $p_2 = \text{col}(-0.82, 0.57)$, and $p_3 = \text{col}(0, 0)$. The estimation error covariance matrices are $P_{o \in \mathbb{I}_3}(t_0) = I_2$. The process noise spectral densities are $Q_{o \in \mathbb{I}_3} = 0.1 I_2$ and measurement noise spectral densities are $R_{o \in \mathbb{I}_3} = 10 I_2$.

The smooth weighting functions for the measurement matrices are $W_{o \in \mathbb{I}_3} = w_{C^\infty}(\cdot) I_2$, using the weighting function of Example 1 with $n = 1$, $K_1 = 3.3 \times 10^{-5} I_2$. The weighting function of (12) is the same for all objects, also defined as in Example 1 with $n = 1$ and $K_1 = 5.2 \times 10^{-7} I_2$.

The optimization horizon is 120 s and the sampling interval 60 s. A 2-point Radau collocation is used for discretization of the state variables with a total for 40 finite elements at each optimization horizon. The control input is piecewise constant with 20 finite elements over the horizon. Meters are scaled by 1/100 in the optimization problem and the following variables are used in the scaled OCP: $v_{o \in \mathbb{I}_3}^L(t) = 25 \text{ col}(1, 1)$, $\Gamma(t) = 5$, $\Xi(t) = 0$, and $v_{o \in \mathbb{I}_3}^M = 40 \text{ col}(1, 1)$.

4.3 Results

All the planned paths were readily available at least 30 s before they were supposed to be used, see Figure 8. This allowed the flight operator to successfully upload the waypoints to the aircraft in due time, so that the switch between old and new waypoints could be made every 60 s. The graphical interface of the flight management system is not designed for rapid changes of

waypoint files, so it was stressful for the flight operator to execute this task. At several occasions the operator failed to switch to the new waypoint file in due time, resulting in the autopilot to perform a re-run of the currently tracked waypoint. This basically means that the aircraft did not fly sufficiently close to the waypoint and it was not properly visited. This behavior does not coincide well with the path planner, so the experiment had to be reset. The results in the following capture a run where this problem did not occur. Another run is illustrated in [Haugen and Imsland \(2014b\)](#).

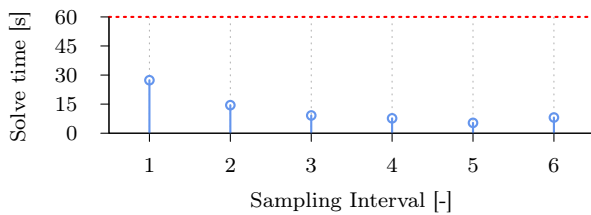


Figure 8: The computation time of each optimization problem is always less than 30 s. This gives the operator enough time to upload the new waypoints within the sampling period of 60 s.

As can be seen in Figure 9, the planned paths were within the constraints of the vehicle model. Nevertheless, no wind speeds were included in the path planning, so the aircraft still had to struggle to properly follow the planned path. The autopilot is set with a constant airspeed. With nonzero wind velocity, the aircraft did not move along the path, but the temporal execution was incorrect. As a result, the aircraft might come too late or too early to a specific location. This was not a huge problem in our case, but by including estimates of the wind velocity in the vehicle model of the path planner and/or allowing some leeway in the desired airspeed, this error can perhaps be reduced. We did try the former approach with an extended Kalman filter, but likely due to poor wind velocity estimates, we did not succeed to obtain satisfactory results.

A plot showing the experiment with north and east coordinates are displayed in Figure 10. The shaded polygon is the admissible operation region. Dashed lines represent optimized/predicted trajectories, and solid lines are filtered values. An object marker indicates that the object has been observed by the aircraft (there are also markers at t_0). An aircraft marker indicates the instant of switching from one sampling interval to the next, which occurs every 60 s. We see that the aircraft is able to follow the planned paths fairly well and does observe the objects under surveillance without violating the admissible region.

The uncertainty of an object's position is significantly reduced every time it is being observed. Once

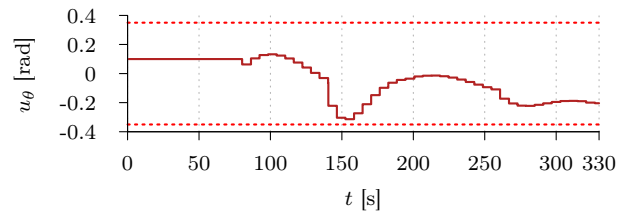


Figure 9: The optimized control input that are used together with the low-fidelity vehicle model to construct the desired paths for the mobile sensor. The input remains within the upper and lower constraint, which are indicated by dashed lines.

an object leaves the FOV of the mobile sensor, the uncertainty increases. Figure 11 illustrates this behavior during the experiment. It should be pointed out that the objects were simulated without noise, so the path planner perfectly predicted their future locations. Even if noise had been included in the object simulations, the mobile sensor still might have been able to find the objects, since they were observed fairly often, and that the FOV was relatively large. In the opposite case, an unobserved object leads to greater uncertainty and higher priority at the next optimization horizon. This does not remedy the fact that an object was not where it was predicted to be, and the object might still remain unobserved. The current framework does not handle the event of missed observations, and this should be considered as a relevant extension. Moreover, if an object leaves the region of interest, or if a new one arrives, this can easily be added to the path planner at the next sampling interval.

5 Concluding Remarks

The optimization-based path planner for object surveillance devised in [Haugen and Imsland \(2013b\)](#) has been successfully demonstrated in a full-scale experiment. Although the planned paths can be followed successfully, including wind velocity estimates in the vehicle model of the path planner, or manipulating the desired airspeed of the autopilot is advised to improve path-tracking performance.

The optimization framework in its current form does not scale well with increasing number of objects. The reason for this is that the computation time of the sampling intervals rapidly increases due to the curse of dimensionality. At some point the computation time becomes larger than the horizon length and the approach is no longer realizable. To overcome this issue, future work may include clever partitioning of the monitor-

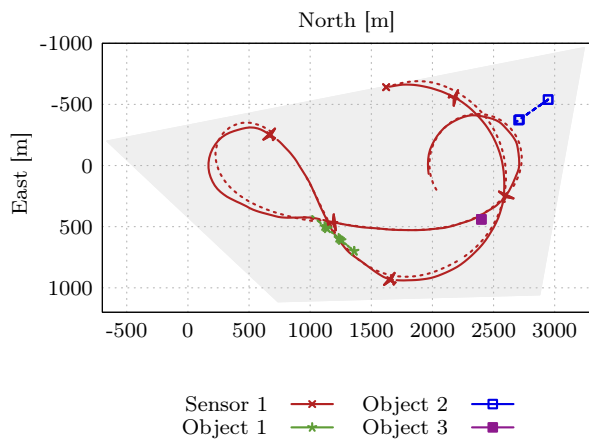


Figure 10: The mobile sensor performs remote sensing of the three objects in question. The dashed line represents optimized/predicted trajectories, whereas solid lines are filtered values. Every aircraft marker represents the instant when a new sampling interval is utilized. An object observation is represented with a line marker. The shaded polygon is the admissible region.

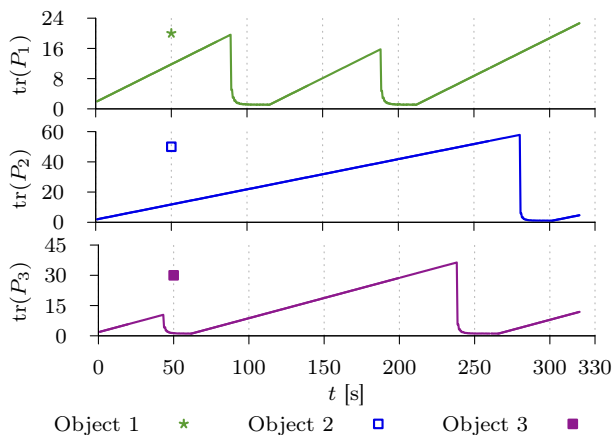


Figure 11: The trace of each object’s covariance matrix. An object observation is indicated by a significant reduction in the trace magnitude.

ing task, where an instance of the monitoring systems presented herein executes within each partition. The independent instances are connected to a higher-level *Supervision* component, which is responsible for ensuring that sub-tasks are properly delegated.

Other further work include extending the framework to handle missed observations, allowing simultaneous detection and tracking, and executing a more real-life experiment with real objects. It would also be interesting to see an execution without human-in-the-loop, both allowing more objects to be observed and displaying truly autonomous surveillance.

Acknowledgments

Research partly funded by Research Council of Norway (RCN) KMB project no. 199567: “Arctic DP”, with partners Kongsberg Maritime, Statoil, and DNV GL, and partly by RCN project no. 223254: CoE AMOS. The authors would like to thank Carl Erik Stephansen, Vegard Hovstein, and other involved parties at Maritime Robotics for their professional and flawless execution of the practical sides of the experiments. The authors also appreciate the assistance provided by Lars Semb, Anders Willersrud, and Anne Mai Ersdal during the field experiment period. Additionally, the authors would like to thank Francesco Scibilia and Esten Ingar Grøtli for constructive feedback on the manuscript.

References

- AMOS. Center for Autonomous Marine Operations and Systems. 2014. URL <http://www.ntnu.edu/amos/centre-for-autonomous-marine-operations-and-systems>.
- Andersson, J., Åkesson, J., and Diehl, M. CasADi: A symbolic package for automatic differentiation and optimal control. In S. Forth, P. Hovland, E. Phipps, J. Utke, and A. Walther, editors, *Recent Advances in Algorithmic Differentiation*, volume 87 of *Lecture Notes in Computational Sci. and Eng.*, pages 297–307. Springer-Verlag, Berlin Heidelberg, Germany, 2012. doi:10.1007/978-3-642-30023-3_27.
- Arctic Marine Solutions. 2014. URL www.arcticmarinesolutions.se.
- Biegler, L. T. *Nonlinear Programming: Concepts, Algorithms & Applications to Chemical Processes*. SIAM, Philadelphia, PA, 2010. doi:10.1137/1.9780898719383.
- Burns, J. A., Cliff, E. M., and Rautenberg, C. A distributed parameter control approach to optimal fil-

- tering and smoothing with mobile sensor networks. In *Proc. 17th Mediterranean Conf. Control & Automation, MED '09*. Thessaloniki, Greece, pages 181–186, 2009. doi:[10.1109/MED.2009.5164536](https://doi.org/10.1109/MED.2009.5164536).
- CAA–Norway. Civil Aviation Authority – Norway, RPAS-FAQ. 2014. URL www.luftfartstilsynet.no/selvbetjening/allmennfly/RPAS-FAQ/.
- Choi, H. and How, J. P. Continuous trajectory planning of mobile sensors for informative forecasting. *Automatica*, 2010. 46(8):1266–1275. doi:[10.1016/j.automatica.2010.05.004](https://doi.org/10.1016/j.automatica.2010.05.004).
- Cloud Cap Technology. 2013. URL <http://www.cloudcaptech.com>.
- Crowe, W., Davis, K. D., la Cour-Harbo, A., Vihma, T., Lesenkov, S., Eppi, R., Weatherhead, E. C., Liu, P., Raustein, M., Abrahamsson, M., Johansen, K.-S., and Marshall, D. Enabling science use of unmanned aircraft systems for arctic environmental monitoring. Technical Report 6, Arctic Monitoring and Assessment Programme (AMAP), Oslo, Norway, 2012. URL <http://amap.no/documents/download/938>.
- Demetriou, M. A. Guidance of mobile actuator-plus-sensor networks for improved control and estimation of distributed parameter systems. *IEEE Trans. Autom. Control*, 2010. 55(7):1570–1584. doi:[10.1109/TAC.2010.2042229](https://doi.org/10.1109/TAC.2010.2042229).
- Demetriou, M. A. and Hussein, I. I. Estimation of spatially distributed processes using mobile spatially distributed sensor network. *SIAM J. Control Optim.*, 2009. 48(1):266–291. doi:[10.1137/060677884](https://doi.org/10.1137/060677884).
- Edmond, C., Liferov, P., and Metge, M. Ice and iceberg management plans for shtokman field. In *Proc. OTC Arctic Technol. Conf. 2011*. Houston, TX, pages 1–9, 2011. doi:[10.4043/22103-MS](https://doi.org/10.4043/22103-MS).
- Eik, K. Review of experiences within ice and iceberg management. *J. Navig.*, 2008. 61(4):557–572. doi:[10.1017/S0373463308004839](https://doi.org/10.1017/S0373463308004839).
- Eik, K. and Løset, S. Specification for a sub-surface ice intelligence system. In *Proc. ASME 28th Int. Conf. Ocean, Offshore and Arctic Eng., OMAE2009*. Honolulu, HI, pages 103–109, 2009. doi:[10.1115/OMAE2009-79606](https://doi.org/10.1115/OMAE2009-79606).
- Fossen, T. I. *Handbook of Marine Craft Hydrodynamics and Motion Control*. John Wiley & Sons Inc., Hoboken, NJ, 2011. doi:[10.1002/9781119994138](https://doi.org/10.1002/9781119994138).
- Frew, E. W. and Brown, T. X. Networking issues for small unmanned aircraft systems. *J. Intell. Robot. Syst.*, 2009. 54(1–3):21–37. doi:[10.1007/s10846-008-9253-2](https://doi.org/10.1007/s10846-008-9253-2).
- Gürtner, A., Baardson, B. H. H., Kaasa, G.-O., and Lundin, E. Aspects of importance related to arctic dp operations. In *Proc. ASME 31st Int. Conf. Ocean, Offshore and Arctic Eng., OMAE2012*. Rio de Janeiro, Brazil, pages 617–623, 2012. doi:[10.1115/OMAE2012-84226](https://doi.org/10.1115/OMAE2012-84226).
- Hamilton, J., Holub, C., Blunt, J., Mitchell, D., and Kokkinis, T. Ice management for support of arctic floating operations. In *Proc. OTC Arctic Technol. Conf. 2011*. Houston, TX, pages 1–12, 2011. doi:[10.4043/22105-MS](https://doi.org/10.4043/22105-MS).
- Haugen, J., Grøtli, E. I., and Imsland, L. State estimation of ice thickness distribution using mobile sensors. In *Proc. IEEE Multi-Conf. Syst. Control*. Dubrovnik, Croatia, pages 336–343, 2012. doi:[10.1109/CCA.2012.6402649](https://doi.org/10.1109/CCA.2012.6402649).
- Haugen, J. and Imsland, L. Optimization-based autonomous remote sensing of surface objects using an unmanned aerial vehicle. In *Proc. European Control Conf. (ECC)*. Zurich, Switzerland, pages 1242–1249, 2013a. URL <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6669610>.
- Haugen, J. and Imsland, L. UAV Path Planning for Multitarget Tracking with Experiments. In *Proc. 2nd IFAC Workshop Res., Develop. and Educ. Unmanned Aerial Syst.* Compiègne, France, pages 316–323, 2013b. doi:[10.3182/20131120-3-FR-4045.00061](https://doi.org/10.3182/20131120-3-FR-4045.00061).
- Haugen, J. and Imsland, L. Monitoring an advection-diffusion process using aerial mobile sensors. *Unmanned Systems*, 2014a. Submitted.
- Haugen, J. and Imsland, L. Monitoring moving objects using aerial mobile sensors. *IEEE Trans. Control Syst. Technol.*, 2014b. Accepted.
- Haugen, J., Imsland, L., Løset, S., and Skjetne, R. Ice observer system for ice management operations. In *Proc. 21st Int. Offshore and Polar Eng. Conf.* Maui, HI, pages 1120–1127, 2011.
- Hindmarsh, A. C., Brown, P. N., Grant, K. E., Lee, S. L., Serban, R., Shumaker, D. E., and Woodward, C. S. SUNDIALS : Suite of nonlinear and differential / algebraic equation solvers. *ACM T. Math. Software*, 2005. 31(3):363–396. doi:[10.1145/1089014.1089020](https://doi.org/10.1145/1089014.1089020).

- HSL. A collection of Fortran codes for large scale scientific computation. 2011. URL <http://www.hs1.rl.ac.uk>.
- Keinonen, A. J. Ice management for ice offshore operations. In *Proc. OTC Arctic Technol. Conf. 2008*. Houston, TX, pages 1–15, 2008. doi:10.4043/19275-MS.
- Looker, J. R. Minimum paths to interception of a moving target when constrained by turning radius. Technical Report DSTO-TR-2227, Defence Sci. & Technol. Organisation, Canberra, Australia, 2008. URL <http://dSPACE.dsto.defence.gov.au/dSPACE/bitstream/1947/9741/1/DSTO-TR-2227PR.pdf>.
- Maritime Robotics. 2013. URL <http://www.maritimerobotics.com>.
- Metrikin, I., Løset, S., Jenssen, N. A., and Kerkeni, S. Numerical simulation of dynamic positioning in ice. *Marine Technol. Soc. J.*, 2013. 47(2):14–30. doi:10.4031/MTSJ.47.2.2.
- Moran, K., Backman, J., and Farrell, J. W. Deep-water drilling in the arctic ocean’s permanent sea ice. In J. Backman, K. Moran, D. B. McInroy, L. A. Mayer, and the Expedition 302 Scientists, editors, *Proc. IODP, 302*. Integrated Ocean Drilling Program Management Int., Inc., Edinburgh, UK, pages 1–13, 2006. doi:10.2204/iodp.proc.302.106.2006.
- Morbidi, F. and Mariottini, G. L. Active target tracking and cooperative localization for teams of aerial vehicles. *IEEE Trans. Control Syst. Technol.*, 2013. 21(5):1694–1707. doi:10.1109/TCST.2012.2221092.
- Norut. Norut UAV remote sensing. 2014. URL <http://norut.no/en/satelitter-fjernmaling-og-ubemannede-fly>.
- Parker, L. E. Cooperative robotics for multi-target observation. *Intell. Autom. Soft Comput.*, 1999. 5(1):5–19. doi:10.1080/10798587.1999.10750747.
- Rathinam, S., Sengupta, R., and Darbha, S. A resource allocation algorithm for multivehicle systems with nonholonomic constraints. *IEEE Trans. Autom. Sci. Eng.*, 2007. 4(1):98–104. doi:10.1109/TASE.2006.872110.
- Savla, K., Frazzoli, E., and Bullo, F. Traveling salesperson problems for the Dubins vehicle. *IEEE Trans. Autom. Control*, 2008. 53(6):1378–1391. doi:10.1109/TAC.2008.925814.
- Sheykin, I. B. Icebreaker reconnaissance for ice management: offshore experience. In *Proc. 20th IAHR Int. Symp. Ice*. Lahti, Finland, pages 525–536, 2010.
- Simicon. Simicon Arctic UAS. 2011. URL <http://simicon.no/>.
- Simon, D. *Optimal State Estimation*. John Wiley & Sons Inc., Hoboken, NJ, 2006.
- Tang, Z. and Özgüner, U. Motion planning for multitarget surveillance with mobile sensor agents. *IEEE Trans. Robot.*, 2005. 21(5):898–908. doi:10.1109/TRO.2005.847567.
- Tricaud, C. and Chen, Y. *Optimal Mobile Sensing and Actuation Policies in Cyber-physical Systems*. Springer-Verlag, London, UK, 2012. doi:10.1007/978-1-4471-2262-3_1.
- UAV Factory. 2013. URL <http://www.uavfactory.com>.
- Wächter, A. and Biegler, L. T. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Math. Programming*, 2006. 106:25–57. doi:10.1007/s10107-004-0559-y.
- Wang, Y. and Hussein, I. I. *Search and Classification Using Multiple Autonomous Vehicles*. Springer-Verlag, London, UK, 2012. doi:10.1007/978-1-4471-2957-8_1.
- Xianyi, Z., Qian, W., and Yunquan, Z. Model-driven level 3 BLAS performance optimization on Loongson 3A processor. In *Proc. IEEE 18th Int. Conf. Parallel and Distributed Syst. (ICPADS)*. Singapore, pages 684–691, 2012. doi:10.1109/ICPADS.2012.97.