Optimization-based planning and control of AUVs applied to adaptive sampling under ice *

Jens E. Bremnes[1] Alex Devonport[2] He Yin[2]
Murat Arcak[2] Asgeir J. Sørensen[1] Ingrid B. Utne[1]

[1]Department of Marine Technology (IMT), Norwegian University of Science and Technology (NTNU), Norway Centre for Autonomous Marine Operations and Systems (AMOS)
Emails: {jens.e.bremnes, asgeir.sorensen, ingrid.b.utne}@ntnu.no
[2]Deptartment of Electrical Engineering and Computer Sciences (EECS), University of California, Berkeley (UCB), USA
Emails: {alex_devonport, he_yin, arcak}@berkeley.edu

Abstract

This paper presents a framework for optimization-based informative planning and control with applications to adaptive sampling with AUVs under sea ice. A spatial model of the information of interest is approximated as a Gaussian process (GP), which is learned online from in-situ sensor data. The planner uses a two-layer model predictive control (MPC) scheme on a low-fidelity model of the vehicle for exploration and exploitation of the GP, subject to safety constraints. The planner trajectories are then tracked using a constant bearing based guidance law, aligning the desired orientation of the AUV toward the planned trajectory. The proposed framework enables the vehicle to plan and replan its mission as new data is obtained, while ensuring tracking of the planned trajectories and safety constraint satisfaction. Simulation results of a case study are presented for demonstrating the performance of the proposed method. An AUV is tasked with finding and tracking concentrations of marine biomass in 3D under sea ice while avoiding collisions.

1 Introduction

There has recently been an increasing interest in conducting ocean research in polar regions and under sea ice. Autonomous underwater vehicles (AUVs)

*This work was supported by the Research Council of Norway through the Centre of Excellence funding scheme, NTNU AMOS, project number 223254, and the UNLOCK project, through the Research Council of Norway FRINATEK funding scheme, project number 274441.

provide unique mapping capabilities for spatial coverage and resolution, and are able to operate untethered with limited human intervention over relatively large time scales. Operating under sea ice imposes demanding requirements to the system due to reduced capabilities of the navigation sensors, limited communication and harsh and unstructured environments [1]. The vehicle might encounter drifting submerged ice features, such as ice ridges, ice keels and icebergs, which impose a risk of collision. Collisions with ice may damage the thrusters, antenna and/or the hull of the vehicle, and in the worst case, the vehicle might get stuck under the ice.

Also, the biological and oceanographic phenomena of interest in these oceans are highly dependent on location and time, that is, spatial and temporal scales. Due to the size of the domain and the limited time of operations, it is not feasible to sample everything in detail. Relying purely on pre-programmed missions based on prior data may lead to both unsafe and ineffective data acquisition.

Adaptive sampling for mobile robotics is an active research area. [2] use a chance-constrained Markov decision process formulation for calculating an optimal trajectory for maximizing predictions from a GP for AUVs. [3] and [4] use a GP discretized on a grid for adaptive sampling for AUVs, using a greedy approach for choosing which grid point to sample next.

A commonly used method in planning is MPC [5], where a model of the system is used to predict the future states over a finite time horizon, and an optimal sequence of control inputs is computed by solving a constrained finite time optimization problem by minimizing a cost function. Advantages of such online optimization-based methods are that they use updated information in each iteration, and allow for formulating safety and input constraints, and checking if they are satisfied for future predictions. [6] presents an MPC approach to the information filter for informative path planning.

For complex planning and control problems, it might be too computationally demanding to solve the whole problem within one optimization program while satisfying sampling time requirements. To alleviate this issue, a common approach is to use a two-layer approach, as done in [7] for motion planning, and in [8] for planner-tracker design.

This paper proposes a framework for optimization-based informative planning and control with applications to adaptive sampling of AUVs under sea ice. The planner learns a spatial model of the information of interest (IOI) online using a GP from in-situ payload sensor data. The planner then calculates an optimal planner trajectory using two-layer kinematic MPC based on a cost function expressed in terms of exploitation and exploration of the GP. Safety constraints may be included in both layers. The planner trajectories are then tracked using a constant bearing guidance law. This will increase safety and efficiency of adaptive AUV operations.

The paper is organized as follows: In Section 2, preliminaries on mathematical modeling of AUVs and GPs are given. Section 3 presents the proposed framework. Simulation results are given in Section 4. Section 5 concludes the paper.

2 Preliminaries

2.1 Mathematical modeling an underwater vehicle

The generalized position of an AUV in the Earth-fixed North-East-Down (NED) reference frame $\{n\}$ is given by $\boldsymbol{\eta} = [x, y, z, \phi, \theta, \psi]^\top$, and the velocity of an AUV in the body-fixed reference frame $\{b\}$ is given by $\boldsymbol{\nu} = [u, v, w, p, q, r]^\top$. The kinematic and kinetic equations of motion for the vehicle may be described by [9]

$$\dot{\boldsymbol{\eta}} = \boldsymbol{J}(\boldsymbol{\eta})\boldsymbol{\nu}, \tag{1a}$$

$$\boldsymbol{M}\dot{\boldsymbol{\nu}} + \boldsymbol{C}(\boldsymbol{\nu})\boldsymbol{\nu} + \boldsymbol{D}(\boldsymbol{\nu})\boldsymbol{\nu} + \boldsymbol{g}(\boldsymbol{\eta}) = \boldsymbol{\tau}, \tag{1b}$$

where $\boldsymbol{J}(\boldsymbol{\eta})$ is a matrix transforming the velocities from $\{b\}$ to $\{n\}$, $\boldsymbol{M} = \boldsymbol{M}_{RB} + \boldsymbol{M}_A$ is the rigid-body and added mass inertia matrix, $\boldsymbol{C}(\boldsymbol{\nu}) = \boldsymbol{C}_{RB}(\boldsymbol{\nu}) + \boldsymbol{C}_A(\boldsymbol{\nu})$ is the rigid-body and added mass Coriolis and centripetal matrix, $\boldsymbol{D}(\boldsymbol{\nu})$ is the damping matrix, $\boldsymbol{g}(\boldsymbol{\eta})$ is the hydrostatic restoring force vector, and $\boldsymbol{\tau}$ is the body-fixed propulsion force/torque vector generated by the vehicle's actuators. For more information, the reader is referred to [9].

2.2 Gaussian process

A GP is a random variable over functions with the property that any collection of point evaluations is a random variable with a multivariate normal distribution [10]. They are fully specified by the mean value at every point in the domain, denoted as the *mean function* $m(\boldsymbol{x})$, and the covariance between each pair of points in the domain, denoted as the *covariance function* $c(\boldsymbol{x}, \boldsymbol{x'})$, for two points $\boldsymbol{x}$ and $\boldsymbol{x}'$.

Formally, $g(\boldsymbol{x}) \sim \mathcal{GP}(m(\boldsymbol{x}), c(\boldsymbol{x}, \boldsymbol{x'}))$ denotes that $g(\boldsymbol{x})$ is a GP with mean function $m(\boldsymbol{x})$ and covariance function $c(\boldsymbol{x}, \boldsymbol{x'})$, so that $m(x) = \mathbb{E}[g(\boldsymbol{x})]$ and $c(\boldsymbol{x}, \boldsymbol{x'}) = \mathbb{E}[(g(\boldsymbol{x}) - m(\boldsymbol{x}))(g(\boldsymbol{x'}) - m(\boldsymbol{x'}))]$.

GPs are used as a Bayesian, nonparametric regression model by choosing a GP as a *prior distribution* over the unknown function and conditioning the GP on the data. Under a Gaussian noise assumption, the *posterior distribution* over the unknown function (that is, the prior GP conditioned on the data) is also a GP, whose mean and covariance are known analytically. Since GPs provide an approach to modeling spatially and temporally correlated data with probabilistic uncertainties, GP regression is a popular model for environmental sampling applications [11]. [12] evaluates and compares the performance of different 3D spatial modeling and interpolation methods for the marine pelagic environment, and conclude that GP regression is the preferred method.

GP regression begins with a dataset $(\boldsymbol{x}^{(i)}, y^{(i)})$ with n datapoints, for $i = 1, ...n$, where $y^{(i)}$ is the i^{th} measurement in $\boldsymbol{y}$ of the underlying function g at the point $\boldsymbol{x}^{(i)}$ corrupted by Gaussian noise with variance σ_n^2, and the selection of a *kernel function* $c(\boldsymbol{x}, \boldsymbol{x'})$ which is used as the covariance function of the GP prior. The GP model's prediction of the underlying process g at a test point $\boldsymbol{x}_*$ is the evaluation of the posterior GP at that point. This yields a *predictive*

distribution, which is a Gaussian distribution with mean $\bar{g}_*$ and variance $\mathbb{V}[g_*]$ given by [10]

$$\bar{g}_* = \boldsymbol{c}_*^\top \left(\boldsymbol{\Sigma} + \sigma_n^2 \boldsymbol{I}\right)^{-1} \boldsymbol{y}, \tag{2}$$

$$\mathbb{V}[g_*] = c(\boldsymbol{x}_*, \boldsymbol{x}_*) - \boldsymbol{c}_*^\top \left(\boldsymbol{\Sigma} + \sigma_n^2 \boldsymbol{I}\right) \boldsymbol{c}_*, \tag{3}$$

where $\boldsymbol{\Sigma}$ is the matrix with elements $(\boldsymbol{\Sigma})_{ij} = c(\boldsymbol{x}^{(i)}, \boldsymbol{x}^{(j)})$, and $\boldsymbol{c}_*$ is the vector with elements $(\boldsymbol{c}_*)_i = c(\boldsymbol{x}_*, \boldsymbol{x}^{(i)})$. The predictive mean corresponds to the "predicted value" of the GP regression model, while the predictive variance provides a quantified uncertainty about the accuracy of the predictive mean.

The quality of the regression is determined to a great extent by the kernel function. To see this, consider that the predictive mean is simply a weighted sum of kernel functions evaluated at the data points. If we let $\boldsymbol{\alpha} = \left(\boldsymbol{\Sigma} + \sigma_n^2 \boldsymbol{I}\right)^{-1} \boldsymbol{y}$, then

$$\bar{g}_* = \sum_{i=1}^{n} \alpha_i c(\boldsymbol{x}_*, \boldsymbol{x}^{(i)}),$$

where α_i is the i^{th} element of $\boldsymbol{\alpha}$. A similar expression exists for the predictive variance. From this expression, it is clear that the kernel function determines most of the mathematical properties of the regression function, such as continuity, stationarity, periodicity, and correlation lengths in each dimension. Choosing a kernel function that is suitable for the underlying process is therefore a critical part of the model selection.

The most common choice of kernel function for GP regression modeling is the emphisometric squared exponential kernel

$$c(\boldsymbol{x}, \boldsymbol{x'}) = \sigma^2 \exp\left(-\frac{\|\boldsymbol{x} - \boldsymbol{x'}\|^2}{2\ell^2}\right),$$

where $\|\cdot\|$ denotes the Euclidean norm. The scalars σ^2 and ℓ are *kernel hyperparameters*: σ^2 determines the magnitude of the predictive variance, and ℓ determines the *lengthscale*, or correlation length, of the predictions. This kernel function forces the lengthscale in each dimension to be identical, which is the reason for the name "isometric".

3 Optimization-based informative planning and control

We propose a framework for optimization-based informative planning and control. The framework may be separated into the following components:

A) Spatial modeling layer

B) Long-term planning layer

C) Short-term planning layer

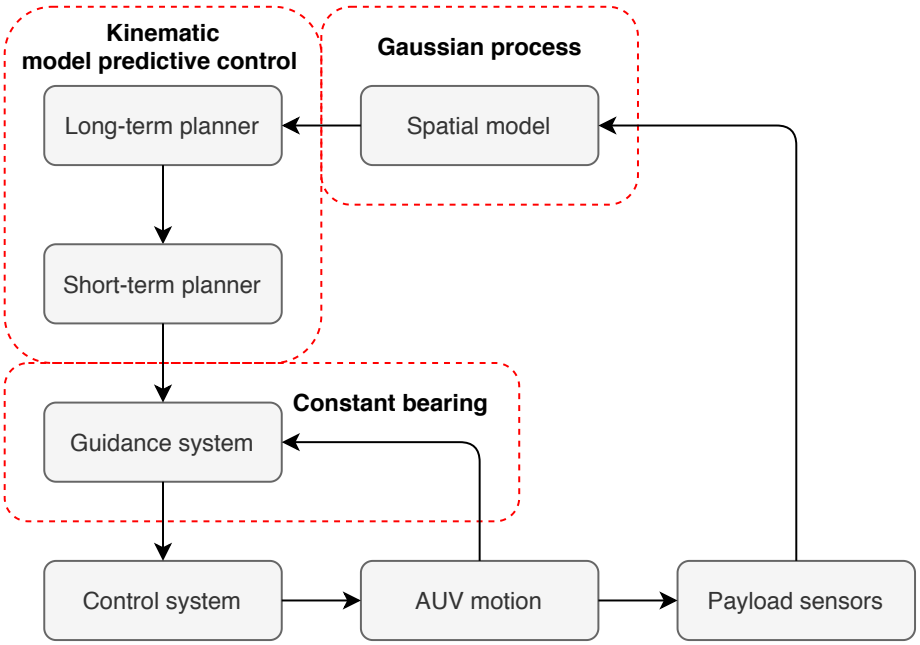

Figure 1: Block diagram of the proposed framework. The dotted red lines highlight the main focus of this paper.

D) Guidance and control layer

This framework should enable informative planning and control in terms of exploration and exploitation of a GP, while satisfying safety constraints. Figure 1 shows the structure of the proposed framework.

3.1 Spatial modeling layer

At this layer, the AUV continuously records new measurements, and updates its GP-based spatial model of the IOI. The restriction that the lengthscale shall be identical in each dimension is a serious limitation for modeling variables across the water column, as the correlation length in the horizontal directions are typically significantly different from the correlation length in the vertical direction, often differing by an order of magnitude or more. One solution, considered in [3], is to rescale the data so that the correlation length in each dimension is similar. Instead of doing this, we select a kernel function which allows each dimension to have its own lengthscale. Specifically, we use a squared-exponential with *automatic relevance determination* (ARD),

$$c(\boldsymbol{x}, \boldsymbol{x'}) = \sigma^2 \exp\left(-\sum_{i=1}^{d} \frac{(x_i - x_i')^2}{2\ell_i^2}\right),$$

where d is the number of input dimensions considered, and ℓ_i is the lengthscale hyperparameter for input dimension i. Differences in the lengthscales between input dimensions can be handled by a suitable choice for the ℓ_i, without needing to rescale the data.

While the kernel hyperparameters may be chosen using prior knowledge or heuristics, we instead use a data-driven approach, which learns suitable hyperparameter values by optimizing the *marginal likelihood*

$$p(y^{(1)}, \ldots, y^{(n)} | \sigma^2, \ell_1, \ldots, \ell_n, \boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(n)}). \tag{4}$$

That is, the probability of observing the specific values of the $y^{(i)}$ at the input points $\boldsymbol{x}^{(i)}$ with respect to the hyperparameters for n datapoints. In addition to improving the quality of the GP regression, the hyperparameters learned in this fashion also provide some insight into the underlying process. For instance, the optimal lengthscale parameters in the North, East and depth directions give an estimate of how frequently the levels of marine biomass in the water fluctuate in each of those directions.

3.2 Long-term planning layer

The long-term planner (LTP) updates the AUV's plan as new information is obtained in order to maximize the efficiency of the mission objective. Safety constraints with low requirements to response time and sampling rate, such as prohibited areas and limited time and energy, may be included in this layer.

An MPC scheme is used for optimizing the planner trajectories balancing exploration and exploitation of the GP. An exploratory strategy is most efficient when little or no prior information about the environment is available. As more information is gathered, it may be advantageous to gradually shift over to an exploitation-based strategy. Exploration of a GP model is addressed by maximization of the predictive variance, see (3), while exploitation is addressed by utilizing the predictive means, see (2), in the cost function.

Performing regressions on the GP over candidate trajectories is increasingly computationally expensive to do as the number of training points increase. In order to alleviate the issue of the *local optima* problem of optimization, a long planning horizon is needed, which further complicates the computations. Therefore, the LTP will update at a relatively slow sampling rate. In order to reduce the computational complexity of the planner, it is proposed that the planner uses a low-fidelity version of the vehicle model.

For the low-fidelity model used in the planner, we propose to use only the kinematics (1a) assuming self-stabilization in roll. Moreover, due to the underactuated nature of conventional AUVs, the sway and heave velocity v and w are not considered controllable in the planner. This means that it only has direct control over the surge speed, pitch rate and yaw rate. We define the planner states as $\check{\boldsymbol{\eta}} \triangleq [\check{x}, \check{y}, \check{z}, \check{\theta}, \check{\psi}]^\top$ with its corresponding planning input vector $\check{\boldsymbol{\nu}} \triangleq [\check{u}, \check{q}, \check{r}]^\top$, where the planning variables are defined similarly as in Section II.A. Please note that the *check* notation $\check{(\cdot)}$ denote planner variables, describing the reference trajectory to be tracked by AUV.

The proposed GP-based LTP is formulated as

$$\begin{aligned}
\min_{\check{\boldsymbol{\nu}}^{ltp}} \quad & \sum_{k=1}^{N^{ltp}} -\boldsymbol{r}_{k|t}^{\top}\boldsymbol{Q}\boldsymbol{r}_{k|t} + (\check{\boldsymbol{\nu}}_{k|t}^{ltp})^{\top}\boldsymbol{R}\check{\boldsymbol{\nu}}_{k|t}^{ltp} \\
\text{s.t.} \quad & \check{\boldsymbol{\eta}}_{k+1|t}^{ltp} = \check{\boldsymbol{f}}(\check{\boldsymbol{\eta}}_{k|t}^{ltp}, \check{\boldsymbol{\nu}}_{k|t}^{ltp}, T^{ltp}), \\
& \check{\boldsymbol{\eta}}_{k|t}^{ltp} \in \check{\mathcal{X}}^{ltp}, \qquad\qquad k \in \{1, ..., N^{ltp}\} \\
& \check{\boldsymbol{\nu}}_{k|t}^{ltp} \in \check{\mathcal{U}}, \\
& \check{\boldsymbol{\eta}}_{0|t}^{ltp} = \check{\boldsymbol{\eta}}(t)
\end{aligned} \tag{5}$$

where $\check{\boldsymbol{\eta}}_{k|t}^{ltp}$ is the LTP state at step k from time t, $\check{\mathcal{X}}^{ltp}$ is the LTP state constraints, $\check{\boldsymbol{\nu}}_{k|t}^{ltp}$ is the LTP input, $\check{\mathcal{U}}$ is the planner input constraints, $\check{\boldsymbol{f}}$ is the discretized low-fidelity vehicle model, N^{ltp} is the number of prediction steps, T^{ltp} is the sampling time, and $\boldsymbol{Q} \succeq \boldsymbol{0}$ and $\boldsymbol{R} \succeq \boldsymbol{0}$ are weighting matrices. Moreover, $\boldsymbol{r}_{k|t} \triangleq [\mathbb{V}[g_*(\check{\boldsymbol{\eta}}_{k|t}^{ltp})], \bar{g}_*(\check{\boldsymbol{\eta}}_{k|t}^{ltp}),]^{\top}$ represents the reward (negative cost) characterized by the predictive variance (exploration) and the predictive mean (exploitation) in the GP at $\check{\boldsymbol{\eta}}_{k|t}^{ltp}$, respectively.

The first term in the MPC represents the reward characterized by the balance between maximization of variance and gradient seeking, while the last term penalizes the squared norm of the planner input, weighted by the matrices $\boldsymbol{Q}$ and $\boldsymbol{R}$, respectively. Thus, the MPC planner will balance exploration and exploitation of the GP while minimizing effort and adhering to state and input constraints.

3.3 Short-term planning layer

The short-term planner (STP) receives and monitors the plan suggested by the LTP. At a high sampling rate, the planner trajectory is updated locally in a minimally invasive way. We propose to use an MPC scheme with a short prediction horizon, minimizing the squared norm of the difference between the STP input and the LTP input, subject to the STP safety constraints.

The proposed STP is given by

$$\begin{aligned}
\min_{\check{\boldsymbol{\nu}}^{stp}} \quad & \sum_{k=1}^{N^{stp}} (\check{\boldsymbol{\nu}}_{k|t}^{stp} - \check{\boldsymbol{\nu}}_{p|t}^{ltp})^{\top}\boldsymbol{\Gamma}(\check{\boldsymbol{\nu}}_{k|t}^{stp} - \check{\boldsymbol{\nu}}_{p|t}^{ltp}) \\
\text{s.t.} \quad & \check{\boldsymbol{\eta}}_{k+1|t}^{stp} = \check{\boldsymbol{f}}(\check{\boldsymbol{\eta}}_{k|t}^{stp}, \check{\boldsymbol{\nu}}_{k|t}^{stp}, T^{stp}), \\
& \check{\boldsymbol{\eta}}_{k|t}^{stp} \in \check{\mathcal{X}}^{stp}, \qquad\qquad k \in \{1, ..., N^{stp}\} \\
& \check{\boldsymbol{\nu}}_{k|t}^{stp} \in \check{\mathcal{U}}, \\
& \check{\boldsymbol{\eta}}_{0|t}^{stp} = \check{\boldsymbol{\eta}}(t)
\end{aligned} \tag{6}$$

where $\check{\boldsymbol{\eta}}_{k|t}^{stp}$ is the STP state at step k from time t, $\check{\mathcal{X}}^{stp}$ is the STP state constraints, $\check{\boldsymbol{\nu}}_{k|t}^{stp}$ is the STP input, N^{stp} is the number of prediction steps in the

STP, T^{stp} is the STP sampling time, and $\mathbf{\Gamma} \succeq \mathbf{0}$ is a weighting matrix. Moreover, the index $p \triangleq \lceil k\frac{T^{stp}}{T^{ltp}} \rceil$, where $\lceil \cdot \rceil$ denotes the ceiling function operator, accounts for the different sampling times between the LTP and the STP, such that the the input velocities $\check{\boldsymbol{\nu}}^{stp}$ and $\check{\boldsymbol{\nu}}^{ltp}$ are compared at the same points in time.

Remark 1: We have omitted the superscript on the "real" planner state $\check{\boldsymbol{\eta}}(t)$, as this is a continuous signal without any predictions (whereas the LTP and STP states are discrete signals). This signal is calculated from $\dot{\check{\boldsymbol{\eta}}}(t) = \check{\boldsymbol{f}}_c(\check{\boldsymbol{\eta}}(t), \check{\boldsymbol{\nu}}^{stp}(t))$, where $\check{\boldsymbol{f}}_c$ is the continuous-time low-fidelity vehicle model, and $\check{\boldsymbol{\nu}}^{stp}(t)$ is the STP input at time t.

Remark 2: The state constraints of the LTP and the STP do not need to be equal to each other. For example, state constraints with low requirements to sampling rate, such as battery and time limitations, do not need to be included in the STP. Thus, we typically have that $\check{\mathcal{X}}^{stp} \subseteq \check{\mathcal{X}}^{ltp}$. This will reduce the computational complexity of the STP.

Remark 3: Both optimization programs (5) and (6) are non-convex due to the nonlinear nature of AUV dynamics. Nonlinear solvers, e.g. IPOPT [16] are required to solve (5) and (6).

3.4 Guidance and control layer

The low-level control objective of the AUV is to track the trajectory suggested by the planning layer, that is,

$$\lim_{t\to\infty} [\boldsymbol{p}(t) - \check{\boldsymbol{p}}(t)] = \mathbf{0}, \tag{7}$$

where $\boldsymbol{p} \triangleq [x, y, z]^\top$ is the position of the AUV and $\check{\boldsymbol{p}} \triangleq [\check{x}, \check{y}, \check{z}]^\top$ is the planner position. Motivated by interceptor-target guidance laws (commonly used for air-to-air missiles), we propose to use a constant bearing (CB) based guidance law for tracking the planner trajectory. We denote the AUV as the interceptor, and the planner position as the target.

The goal of CB guidance it to reduce the line-of-sight (LOS) vector between the interceptor and the target to zero, such that the interceptor closes on a collision course [9]. The desired linear velocity vector for the AUV in $\{n\}$ is given by [9]

$$\boldsymbol{v}_d^n = \check{\boldsymbol{v}}^n - U_{a,max}\frac{\tilde{\boldsymbol{p}}}{\sqrt{\tilde{\boldsymbol{p}}^\top \tilde{\boldsymbol{p}} + \Delta_{\tilde{p}}^2}}, \tag{8}$$

where $\tilde{\boldsymbol{p}} \triangleq \boldsymbol{p} - \check{\boldsymbol{p}}$ is the position error, $\check{\boldsymbol{v}}^n$ is the linear velocity vector for the planner in $\{n\}$, $U_{a,max}$ is the maximum approach speed toward the target, and $\Delta_{\tilde{p}}$ is a speed tuning parameter.

The desired speed, pitch and heading are then found according to $U_d = \|\boldsymbol{v}_d^n\|$, $\theta_d = -\arctan\left(\frac{\boldsymbol{v}_d^n(3)}{\|\boldsymbol{v}_d^n(1:2)\|}\right) + \kappa_i \int_0^t \tilde{\boldsymbol{p}}(3)d\tau$ and $\psi_d = \arctan\left(\frac{\boldsymbol{v}_d^n(2)}{\boldsymbol{v}_d^n(1)}\right)$, respectively, where κ_i is an integral gain. These reference signals are then sent as input to a low-level control system. It can be shown that this gives uniform convergence

of $\boldsymbol{p}$ to $\check{\boldsymbol{p}}$ [9]. Note that integral action is added to the desired pitch angle for counteracting the influence of non-zero slip angles. The same may be done for the desired heading angle.

4 Results and discussion

This Section presents the case study and simulation setup and discusses the results.

4.1 Case study

For demonstrating the performance of the proposed framework, we present a case study where an AUV is tasked with finding and tracking concentrations of marine biomass in 3D under sea ice. During its operation, it must avoid collisions with the ice. Chlorophyll *a* (Chla) is indicative of primary productivity and phytoplankton concentration [3], and is used as a proxy for marine biomass. We assume that the AUV is able to measure the Chla levels using onboard sensors.

4.2 Safety constraints

Without loss of generality, we consider the following safety constraints:

1. *Ice altitude constraint.* The AUV should not be closer to the ice surface than a certain threshold:

$$-\check{z}_{k|t} + t_{ice} + a_{ice,max} \leq 0,$$

   where $\check{z}_{k|t}$ is the planned depth, the ice thickness is found from $t_{ice} = z - a_{ice}$, a_{ice} is the measured ice altitude (defined as the vertical distance between the AUV and the ice) and $a_{ice,max}$ is the maximum allowed ice altitude.

2. *Collision avoidance constraint.* The AUV should not collide with ice ridges (modeled as ellipsoids):

$$(\check{\boldsymbol{p}}_{k|t} - \boldsymbol{p}_{o,i})^\top \boldsymbol{P}_{o,i} (\check{\boldsymbol{p}}_{k|t} - \boldsymbol{p}_{o,i}) \geq 1,$$

   where $\boldsymbol{p}_{o,i}$ and $\boldsymbol{P}_{o,i}$ are the position and shape matrix of the elliptic ice ridge i, respectively.

Both constraints are added in both the LTP and the STP, such that $\check{\mathcal{X}}^{ltp} \equiv \check{\mathcal{X}}^{stp}$. Ice altitude may be measured by an upwards-looking Doppler velocity log (DVL). Ice ridges may be detected by onboard sensors using sonars and optics.

Remark 4: Ellipsoids may be a poor representation of complex ice structures such as ice ice ridges. We argue that ellipsoids may be used to overapproximate

these structures, thus adding a safety margin to the collision avoidance constraint.

Remark 5: [17] develop dynamic safety envelopes for underwater vehicles, that is, a three-dimensional spatial area around the vehicle that that serves as a virtual protective barrier against collision, using information about the vehicle velocity, the probability of acoustic sensor failure and a time to collision risk indicator. Developing dynamic safety envelopes for use in this framework is of interest for further work, as this may increase safety.

Remark 6: There are other operational challenges which may be addressed through safety constraints, such as battery and time limitations, reduced navigational capabilities, surfacing limitations and rapidly-changing environmental conditions. This will be refined in further work.

4.3 Simulation setup

The implemented dynamics of the vehicle are based on the REMUS-100 AUV [13]. Proportional-integral-derivative (PID) controllers for speed, pitch and heading were implemented for tracking the signals produced by the guidance system. The MPC-based planners were implemented in MATLAB/Simulink using CasADi [15] with the IPOPT [16] solver.

The ice data was synthesized using the *Automatic Terrain Generator* toolbox [14] in MATLAB with initial depth, initial roughness and roughness roughness of $h_0 = 2$, $r_0 = 2$ and $r_r = 2$ meters, respectively. Two ice ridge are included and placed underneath the ice, modeled as ellipsoids with positions $\boldsymbol{p}_{o,1} = [100, 100, 0]$ and $\boldsymbol{p}_{o,2} = [150, 200, 0]$ meters, and shape matrices $\boldsymbol{P}_{o,1} = \boldsymbol{P}_{o,2} = diag([1/75^2, 1/7.5^2, 1/15^2])$ (75 meters long, 7.5 meters wide, 15 meters deep), respectively. The ice considered in this paper is stationary, that is, non-drifting and non-deforming.

The maximum ice altitude is set to $a_{ice,max} = 5$ meters. The guidance parameters are set to $U_{a,max} = 1\ m/s$, $\Delta_{\tilde{p}} = 5$ meters and $\kappa_i = 0.003$.

As navigation is not the main focus of this paper, it is assumed that all states of the system (1) are fully observable with no noise or bias in the simulations. Moreover, the position and shape of the elliptic ice ridges, as well as the altitude, are assumed fully observable.

4.4 Chla distribution and spatial model implementation

For simplicity and without loss of generality, the simulated spatial distribution of Chla in mg/m^3 is given by the sum of three exponential functions

$$Chla(x, y, z) = \sum_{i=1}^{3} e^{\frac{(x-x_{c,i})^2}{\ell_{c,x}^2} + \frac{(y-y_{c,i})^2}{\ell_{c,y}^2} + \frac{(z-z_{c,i})^2}{\ell_{c,z}^2}},$$

where $[x_{c,1}, y_{c,1}, z_{c,1}] = [250, 150, 12]$, $[x_{c,2}, y_{c,2}, z_{c,2}] = [150, 250, 8]$ and $[x_{c,3}, y_{c,3}, z_{c,3}] = [250, 250, 10]$ meters are the local maxima, and $[\ell_{c,x}, \ell_{c,y}, \ell_{c,z}] = [75, 75, 4]$ meters are the lengthscales. Gaussian zero-mean noise with standard deviation $\sigma_y = 0.001\ mg/m^3$ are added to the Chla measurements.

Table 1: State and input constraints in the planner. LB: lower bound. UP: upper bound.

	$\check{x}$	$\check{y}$	$\check{z}$	$\check{\theta}$	$\check{\psi}$	$\check{u}$	$\check{q}$	$\check{r}$
LB	0	0	0	-15	$-\infty$	2	-8.6	-8.6
UP	300	300	30	15	∞	2	8.6	8.6
	[m]	[m]	[m]	[°]	[°]	[m/s]	[°/s]	[°/s]

The hyperparameters for the squared-exponential ARD kernel were learned by optimizing the marginal likelihood as shown in (4) from random samples in the above distribution. The learned values for the hyperparameters are $\ell_x = \ell_y = 125$, $\ell_z = 6.3$ and $\sigma = 0.175$ meters.

A prior mean $Chla_0(x, y, z) = e^{\frac{(x-x_0)^2}{\ell_{0,x}^2} + \frac{(y-y_0)^2}{\ell_{0,y}^2} + \frac{(z-z_0)^2}{\ell_{0,z}^2}}$ with $[x_0, y_0, z_0] = [150, 150, 10]$ and $[\ell_{0,x}, \ell_{0,y}, \ell_{0,z}] = [125, 125, 6.3]$ was used in the GP predictions for guiding the vehicle's search towards the middle of the map.

4.5 Planner implementation

The planners were implemented with sampling times and predictions steps $T^{ltp} = 2$ seconds, $T^{stp} = 0.2$ seconds, $N^{ltp} = 20$ and $N^{stp} = 10$, respectively. The cost matrices in the LTP and STP are given as

$$\boldsymbol{Q} = \begin{bmatrix} 10^3 & 0 \\ 0 & 5n^{0.7} \end{bmatrix}, \boldsymbol{R} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \boldsymbol{\Gamma} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

where $n = n(t)$ is the number of Chla measurements. A new Chla measurement is taken every 5 seconds. The implemented state and input constraints in the planner are given in Table 1.

4.6 Simulation results

The simulations were performed using MATLAB/Simulink on a MacBook Pro with an Intel Core i7 CPU and 16GB RAM. Figure 2 shows the trajectory of the AUV under the ice. Figure 3 shows the North, East and depth trajectories of the AUV.

As seen in Figure 2, the AUV is able to find and track the Chla concentrations. The trajectory of the AUV is highly dependent of the exploration/exploitation trade-off reflected in the tuning of $\boldsymbol{Q}$. The AUV satisfies the ice altitude constraint, and does not collide with the ice ridges. The guidance system and low-level control system are able to track the trajectory suggested by the planner.

The computation time for the LTP ranges from 0.5 second to 5 seconds depending on the number of measurements used in the GP predictions. The computation time for the STP remains fairly constant at 0.1 seconds.

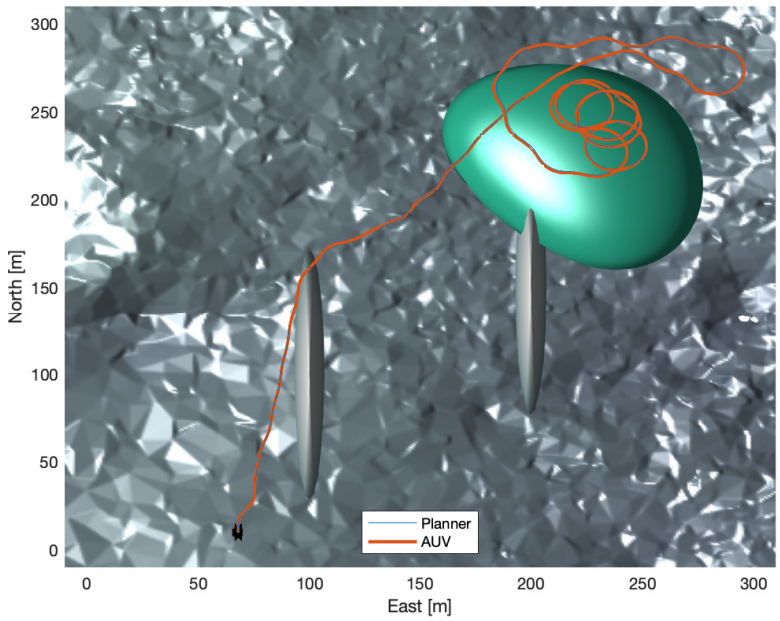

Figure 2: Trajectory of the AUV under the ice. The green structure represents the isosurface with value 1.6 mg/m^3 of the Chla distribution. The gray ellipsoids represent the ice ridges. The black star represents the initial position of the AUV. Please note that the AUV dives around the corner of the first obstacle.

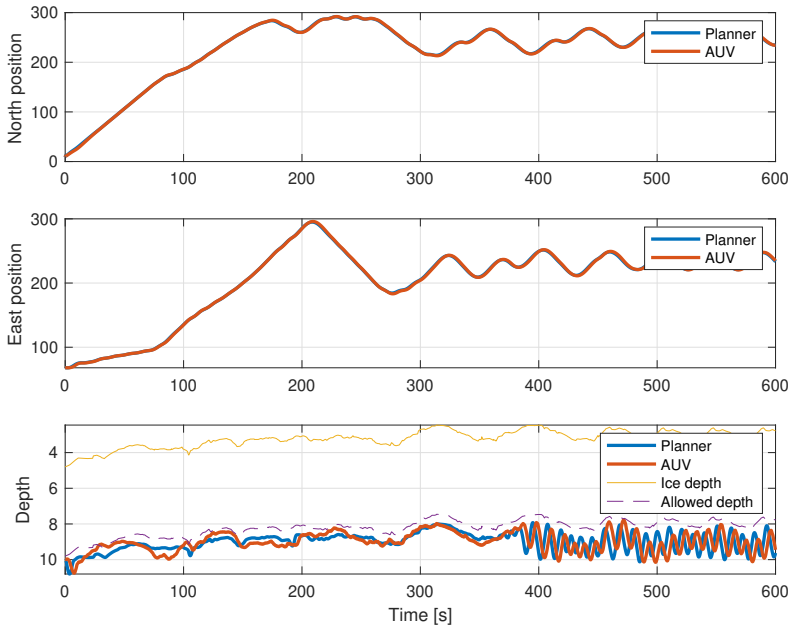

Figure 3: North, East and depth trajectories of the AUV.

Remark 7: An ice following altitude control behavior is achieved when the ice altitude constraint is active. This may happen when the spatial model predicts that there is a high marine biomass concentration close to the ice surface, and/or the area close to the ice surface is unexplored.

Remark 8: This framework is general and allows for modularity. For instance, one may add more layers of planners with different prediction horizons. At the higher levels, long prediction horizons help mitigate the local minima problem in optimization, while at the lower levels, short prediction horizons incorporates local information at a high rate. Moreover, additional safety constraints may be added. Using supervisory switched control to switch between different mission objectives is also of interest. The supervisor may monitor the process in the background, and suggest new mission objectives, e.g. by switching between different planners, or changing parameters in the planners, such as the cost functions and/or the constraints. This is topic for further work.

5 Conclusion

The proposed optimization-based framework for informative planning and control of AUVs integrates spatial modeling and planning using GPs and kinematic MPC at the higher level, with guidance and control at the lower level. New information is incorporated in each planning iteration, used for both GP regression and constraint satisfaction over a finite time horizon, thereby increasing safety and efficiency of adaptive AUV missions. The framework performed well in a simulated case study, where an AUV was tasked with finding and tracking concentrations of marine biomass in 3D under sea ice while satisfying safety constraints. This framework may be used for other AUV applications or other types of autonomous vehicles by changing the relevant parameters and dynamic models of the problem, and adding other safety constraints.

References

[1] M. Brito, and G. Griffiths, "A Bayesian approach for predicting risk of autonomous vehicle loss during their mission," Reliab. Eng. Syst. Safe., vol 146(C), pp. 55-67, 2016.

[2] B. Ayton, B. Williams, and R. Camilli, "Measurement maximizing adaptive sampling with risk bounding functions," in Thirty-Third AAAI Conference on Artificial Intelligence (AAAI-19), vol. 3, January 2019, pp. 7511-7519.

[3] T. O. Fossum, G. M. Glaucia, E. J. Davies, J. E. Ullgren, R. Mendes, G. Johnsen, I. Ellingsen, J. Eidsvik, M. Ludvigsen, and K. Rajan, "Toward adaptive robotic sampling of phytoplankton in the coastal ocean," Sci. Robot., vol. 4(27), pp. 1-11, 2019.

[4] G. E. Berget, T. O. Fossum, T. A. Johansen, J. Eidsvik, and K. Rajan, "Adaptive sampling of ocean processes using an AUV with a Gaussian

proxy model," in 11th IFAC Conference on Control Applications in Marine Systems, Robotics, and Vehicles (CAMS 2018), vol. 51(29), September 2018, pp. 238–243.

[5] J. B. Rawlings, and D. Q. Mayne, Model predictive control: Theory and design, Nob Hill Pub., 2009.

[6] P. Boström-Rost, D. Axehill, and G. Hendeby, "On global optimization for informative path planning," IEEE Control Syst. Lett., vol. 2(4), pp. 833-838, 2018.

[7] C. Liu, T. Tang, H. C. Lin, Y. Cheng, and M. Tomizuka, "SERoCS: Safe and efficient robot collaborative systems for next generation intelligent industrial co-robots," arXiv preprint arXiv:1809.08215, 2018.

[8] H. Yin, "Optimization Based Planner–Tracker Design for Safety Guarantees," arXiv preprint arXiv:1910.00782, 2019.

[9] T. I. Fossen, Handbook of Marine Craft Hydrodynamics and Motion Control, John Wiley Sons, Ltd, 2011.

[10] C. E. Rasmussen, and C. K. I. Williams, Gaussian Processes for Machine Learning, The MIT Press, 2005.

[11] J. Eidsvik, T. Mukerji, and D. Bhattacharjya, Value of Information in the Earth Sciences, Cambridge University Press, 2015.

[12] M. A. Sahlin, A. Forest, and M. Babin, "Assessment of 3d spatial interpolation methods for study of the marine pelagic environment," Mar. Geod. vol. 37(2), pp. 238–266, 2014.

[13] P. Norgren, "Autonomous underwater vehicles in Arctic marine operations - Arctic marine research and ice monitoring," PhD thesis, Norwegian University of Science and Technology, 135 pp, 2018.

[14] T. McClure, Automatic Terrain Generation, 2020 (https://www.mathworks.com/matlabcentral/fileexchange/39559-automatic-terrain-generation), MATLAB Central File Exchange, retrieved July 7, 2020.

[15] J. A. E. Andersson, J. Gillis, G. Horn, J. B. Rawlings, and M. Diehl, "CasADi - A software framework for nonlinear optimization and optimal control," Math. Program. Comput., vol. 11(3), pp. 1–36, 2018.

[16] A. Wächter, and L. T. Biegler, "On the Implementation of a Primal-Dual Interior Point Filter Line Search Algorithm for Large-Scale Nonlinear Programming," Math. Program., vol. 106(1), pp. 25–57, 2006.

[17] J. Hegde, E. H. Henriksen, I. B. Utne, and I. Schjølberg, "Development of dynamic safety envelopes for autonomous remotely operated underwater vehicles," in 28th European Safety and Reliability Conference (ESREL 2018), June 2018, pp. 1–8.