

# Adaptive sampling of algal blooms using an AUV and satellite imagery

Joana Fonseca, Alexandre Rocha, Miguel Aguiar, Karl H. Johansson

**Abstract**—In this paper we propose a method that uses satellite data to improve adaptive sampling missions. We find and track algal bloom fronts using an autonomous underwater vehicle (AUV) equipped with a sensor that measures the concentration of chlorophyll *a*. The proposed method learns the kernel parameters of a Gaussian process model using satellite images of chlorophyll *a* from previous days. Using locally collected data, the AUV estimates the chlorophyll *a* concentration online. The algal bloom front estimate is fed to our motion control algorithm. The performance of this method is evaluated through simulations using a real dataset of an algal bloom front in the Baltic. We consider a real-world scenario with sensor and localization noise and with a detailed AUV model.

## I. INTRODUCTION

We develop an adaptive sampling approach using an autonomous underwater vehicle (AUV) and satellite data to monitor algal blooms. Algal blooms can cause human illness and large scale mortality of fish [1]. The scenario is illustrated in Fig. 1, in which the AUV and available satellite data are highlighted. The algal blooms satellite data is obtained from the Copernicus Marine Environment Monitoring Service (CMEMS) [2]. Algal blooms occur naturally in the sea. However, when algae colonies experience abnormal growth, which can result in the production of harmful toxins [3], they are called harmful algal blooms. In the Baltic Sea, this is due to excessive nutrient input mainly attributed to human pollution. [4]. Motivation to monitor algal blooms is due to its recent increase in frequency, intensity, and geographical distribution in parallel with the increased utilization of coastal waters for aquaculture [5]. There is significant scientific and societal interest in developing systems for automated surveillance and monitoring of algal blooms. Traditional methods for observation, such as satellite imaging or ship-towed sensors, are generally unable to provide measurements at the spatial and temporal resolutions required to understand the detailed dynamic ocean phenomena [6]. While remote sensing with satellites can offer a first guess, such data is weather-dependent and prone to false positives in turbid coastal waters [7]. Therefore, we propose the use of AUVs that can sense water parameters at higher resolutions and coverage than research vessels and buoys, at a reduced risk and cost. In fact, AUVs can perform measurement runs over a long period of time at sea [8], which makes them a suitable choice for oceanographic data collection [9].

This work is supported by Knut and Alice Wallenberg Foundation, Swedish Research Council, and Swedish Foundation for Strategic Research.

J. Fonseca, A. Rocha, M. Aguiar, and K. H. Johansson are with the Division of Decision and Control Systems, School of Electrical Engineering and Computer Science, and Digital Futures, KTH Royal Institute of Technology, SE-100 44 Stockholm, Sweden. {jfgf, aguiar, kallej}@kth.se, rochaa@student.chalmers.se.

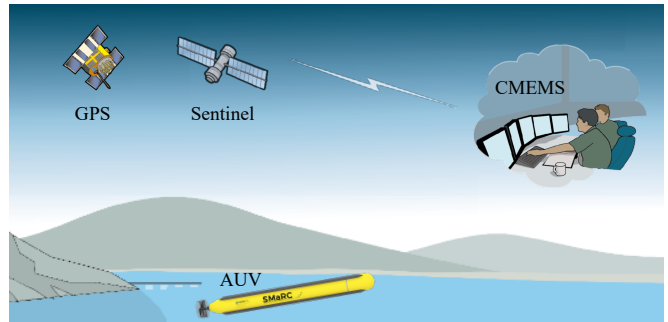


Fig. 1: System overview for algal blooms monitoring based on AUV and satellite data.

There has been a significant amount of work in developing solutions for autonomous ocean sampling. A common approach in the literature is open-loop techniques with a fixed sampling pattern. A widely used sampling pattern is the lawn-mower [10] which has been used for both single-agent [11] and multi-agent [12] cases. Other relevant methods include the spiral and circular patterns in [13]. While these open-loop strategies guarantee coverage of survey areas, they do not react or respond to changes in the environment or the features that are being observed. In such cases, one needs to close the loop. Adaptive sampling is such a closed-loop control in which an agent autonomously makes decisions during a mission in response to environmental changes [14], [15]. As reviewed in [16], adaptive sampling can be divided into three distinct objectives: source localization, front determination, and tracking and mapping. Different types of targets can be considered: thermoclines, algal blooms, oils spills, etc. In addition, using different vehicles can be deployed: single-agent, multi-agent with leader-follower, cooperative multi-agent, etc. Similar to this paper, an adaptive sampling algorithm that augments a standard Gaussian process (GP) with a nearest neighbors prior is proposed in [17]. Unlike our approach however, [17] does not use external data to aid the vehicle's decisions. One essential problem is how to aid ocean sampling missions with external data. An early approach uses predictive ocean models to assist in solving the motion planning problem of steering an AUV to a desired location [18]. Here, it is assumed that there is a prediction for the day of the mission, which is not available in this paper's scenario. Another related result uses knowledge from previous missions to create a model [19].

The main contribution of this paper is a system for algal bloom front tracking comprising a GP model estimator, a gradient estimator, and a motion controller. The GP model estimates the chlorophyll *a* concentration field using satellite data and is updated in real time by conditioning it on the measurements collected by the AUV. The gradient estimator derives the gradient of the chlorophyll *a* concentration field.

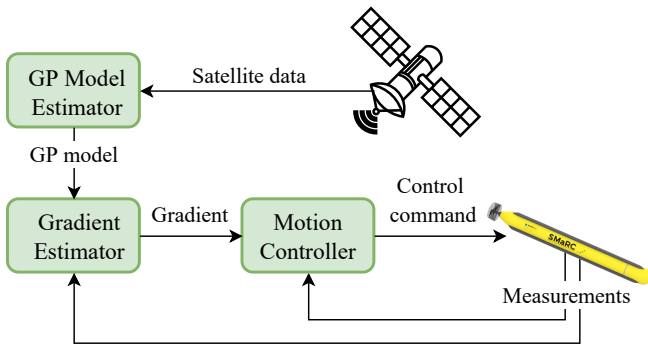


Fig. 2: System architecture with its main components: satellite data, GP model estimator, gradient estimator, motion controller, and AUV.

The motion controller implements a path-planning guidance law for adaptive sampling using the estimated chlorophyll  $a$  concentration and field gradient. We run an experimental evaluation in a simulated environment that highlights the performance of the estimators and controller, considering sensor noise and detailed AUV model.

The paper is organised as follows. In Section II, we introduce the proposed front tracking algorithm. This includes the system architecture, the satellite data, the GP model for the chlorophyll  $a$  concentration, the gradient estimator, the motion controller, and the AUV model. In Section III we analyse the performance of the algorithm with simulations considering disturbances and noise. Concluding remarks and future directions follow in Section IV.

## II. ALGAL BLOOM FRONT TRACKING

This paper considers algal bloom front tracking as an adaptive environmental sampling problem. The objective of front tracking is to find and track a front with no global information on the front’s location or shape, and only use local information collected by the AUV itself, as it moves to explore the map. Then, the AUV has to make the decision of where to explore next given the information it has collected so far. We approach this problem using an AUV with a chlorophyll  $a$  sensor and remote satellite data from CMEMS. Our solution consists of a novel system architecture containing three main components, as seen in the three green blocks of Fig. 2. They are a GP model estimator, a gradient estimator, and a motion controller. In the following subsections, we present the system architecture and its components.

### A. System Architecture

The system architecture is summarised in Fig. 2. Here, we illustrate the main components of the proposed system, from the AUV, to the motion controller, gradient estimator, GP model estimator, and satellite data.

The AUV has a chlorophyll  $a$  sensor that measures the chlorophyll  $a$  concentration at a set frequency, as it moves in the field. The AUV movement is dictated by the control command received from the motion controller. The motion controller uses the AUV’s past measurements and an estimation of the field gradient to calculate the control command, closing the adaptive sampling loop. The gradient

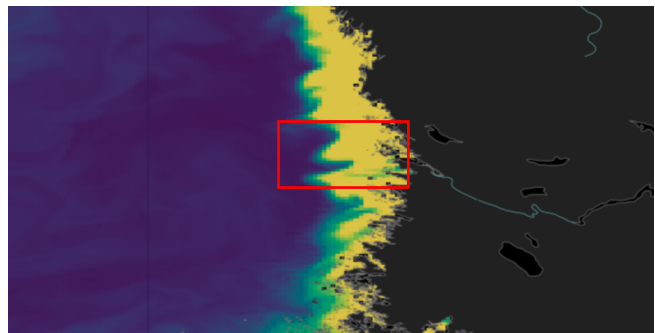


Fig. 3: CMEMS data of chlorophyll  $a$  in the Baltic Sea (blue - yellow), and land (dark grey).

estimator uses both the past measurements taken by the AUV and a model of the chlorophyll  $a$  concentration field to produce an estimate of the chlorophyll  $a$  concentration field gradient. The GP model estimator uses the previous days of satellite data to train kernel parameters of a GP model that represents the field we want to estimate. Finally, the satellite data consists of remote measurements of the chlorophyll  $a$  concentration field from a number of days preceding the mission and is used in the GP model estimator to generate the GP model estimate of the chlorophyll  $a$  concentration field.

### B. Satellite Data

The satellite data concerns chlorophyll  $a$  concentration for a given region. We denote this chlorophyll  $a$  concentration field by  $\delta(\mathbf{p})$ , where  $\delta$  denotes the chlorophyll  $a$  concentration at position  $\mathbf{p}$ . In this paper, we consider surface data only. Fig. 3, shows a plot of sample chlorophyll  $a$  concentration data, where high regions of high concentration are highlighted in yellow and regions of low concentration are highlighted in blue. The dark grey area represents the land. This data has a spatial resolution of 2 km by 2 km and is obtained from CMEMS [20]. The location is on the west coast of Finland, near the coastal city *Pori*. We chose this location because there’s a clear chlorophyll  $a$  bloom front which we hypothesise is due to the nutrients that the river *Kokemäenjoki* carries into the Baltic sea [21]. In this paper, we’ll focus on the region marked by the red square taken on the 17th of April 2021. We chose April because it is the spring season of algal blooms.

### C. GP Model Estimator

The GP model estimator models the chlorophyll  $a$  concentration field  $\delta(\mathbf{p})$  for the given region and time, exploiting prior information from satellite data of the previous days and measurements taken by the AUV in real time. Given this application’s slow time scale, we assume that the chlorophyll  $a$  concentration fields at different days have identical distributions, which simplifies model fitting and reduces the computational complexity.

In order to obtain the chlorophyll  $a$  concentration model, we must define the type of kernel that will accurately depict the process. The kernel represents *a priori* knowledge about

the process by specifying how the chlorophyll  $a$  concentration data is related to the corresponding spatial locations. Among the multitude of kernels described in the literature, only some fit the characteristics of the biogeochemical data we consider. We use the Matérn kernel [22], which proves to be capable of modelling different degrees of smoothness, across both vertical and horizontal length scales [23]. The covariance matrix  $K \in \mathbb{R}^{N \times N}$  is defined with respect to two points in the field map,  $x_i$  and  $x_j$ . Each element of the matrix is computed by the kernel function  $k(x_i, x_j)$  for which  $1 \leq i, j \leq N$ . The kernel is defined as

$$K_{i,j} = k(x_i, x_j) = \sigma_k^2(1+r)e^{-r}, \quad (1)$$

where  $r^2 = (x_i - x_j)^\top M(x_i - x_j)$ , with

$$M = \begin{bmatrix} \left(\frac{\sqrt{3}}{l_0}\right)^2 & 0 \\ 0 & \left(\frac{\sqrt{3}}{l_1}\right)^2 \end{bmatrix}. \quad (2)$$

The kernel hyper-parameters are  $(\sigma_k^2, l_0, l_1)$ , where  $\sigma_k^2$  is the variance of the chlorophyll  $a$  concentration process, and  $(l_0, l_1)$  define the length scale in each dimension.

The kernel hyper-parameters,  $(\sigma_k^2, l_0, l_1)$  are estimated by maximising the log marginal likelihood of the prior distribution - using only the available satellite data from previous days. This data is called the training set, and consists of a vector of size  $N$  containing positions in the chlorophyll  $a$  concentration field  $\mathbf{X} = [\mathbf{p}_1, \dots, \mathbf{p}_N]$ , and their respective chlorophyll  $a$  concentration value  $\mathbf{y} = [\delta_1, \dots, \delta_N]$ . The log marginal likelihood to maximise is

$$\begin{aligned} \log p(\mathbf{y}|\mathbf{X}) = & \quad (3) \\ & -\frac{1}{2}\mathbf{y}^\top (K + \sigma^2 I)^{-1}\mathbf{y} - \frac{1}{2} \log |K + \sigma^2 I| - \frac{N}{2} \log 2\pi \end{aligned}$$

where  $K$  is the  $N \times N$  covariance matrix in which each value is created as in (1), and  $\sigma$  is the noise variance of each data point.

Using the trained kernel, the GP model can be fit with the collected data, using the standard conditioning formulas [22] to obtain the model for the chlorophyll  $a$  concentration field, which we define as  $\bar{\delta}(\mathbf{p})$ . In order to do that, we consider the  $n$  most recent measurements taken by the AUV. It contains the AUV's positions  $\mathbf{P} = [\mathbf{p}_1, \dots, \mathbf{p}_n]$ , and its chlorophyll  $a$  concentration measurements  $\Delta = [\delta_1, \dots, \delta_n]$ . Then, the mean of the model  $\delta(\mathbf{p})$  is denoted by  $\bar{\delta}(\mathbf{p})$  and the covariance by  $\text{cov}(\delta(\mathbf{p}))$ . We can compute the mean and covariance at some point  $\mathbf{p}_*$  as

$$\begin{aligned} \bar{\delta}(\mathbf{p}_*) &= K_* (K + \sigma^2 I)^{-1} \Delta \quad (4) \\ \text{cov}(\delta(\mathbf{p}_*)) &= K_{**} - K_* [K + \sigma^2 I]^{-1} K_*^\top \quad (5) \end{aligned}$$

where  $K \in \mathbb{R}^{n \times n}$  corresponds to the covariance between the data in points  $\mathbf{P}$ ,  $K_* \in \mathbb{R}^{1 \times n}$  corresponds to the covariance between the data in points  $\mathbf{p}_*$  and  $\mathbf{P}$ ,  $K_{**} = \sigma^2$  corresponds to the variance at the point  $\mathbf{p}_*$ , and  $\sigma^2$  is the variance of the measurement noise. Since the set of the  $n$  most recent measurements is changing in time, then our estimate  $\bar{\delta}(\mathbf{p}_*)$  is also changing in time.

#### D. Gradient Estimator

The gradient estimator derives the previously obtained model of the chlorophyll  $a$  concentration to produce an estimate of the chlorophyll  $a$  concentration gradient field. From the equation (4), the gradient  $\nabla \bar{\delta}(\mathbf{p}_*)$  is obtained by computing the derivative of the predicted chlorophyll  $a$  concentration with respect to position  $\mathbf{p}_*$ ,

$$\nabla \bar{\delta}(\mathbf{p}_*) = \nabla_{\mathbf{p}_*} \left[ K_* (K + \sigma^2 I)^{-1} \Delta \right]. \quad (6)$$

Since the second and third terms inside the gradient in (6) are constant relative to  $\mathbf{p}_*$ , we only need to compute  $\nabla_{\mathbf{p}_*} K_*$ . Each element of the  $K_*$  matrix is given by (1), in which  $x_i$  corresponds to  $\mathbf{p}_*$  and  $x_j$  corresponds to  $\mathbf{p}_j \in \mathbf{P}$ . So we take the derivative of  $k(\mathbf{p}_*, \mathbf{p}_j)$  with respect to  $\mathbf{p}_*$ ,

$$\nabla_{\mathbf{p}_*} k(\mathbf{p}_*, \mathbf{p}_i) = -\sigma_k^2 e^{-r} M(\mathbf{p}_* - \mathbf{p}_j),$$

where  $M$  and  $r$  are as in subsection II-C. Note that the gradient of the kernel equation is not defined when the test point in  $\mathbf{P}$  is equal to the current position  $\mathbf{p}_*$ . To account for this, the current position  $\mathbf{p}_*$  is not included in  $\mathbf{P}$  when computing (6). Then the gradient estimate at position  $\mathbf{p}_*$  is

$$\nabla \bar{\delta}(\mathbf{p}_*) = \nabla_{\mathbf{p}_*} K_* (K + \sigma^2 I)^{-1} \Delta. \quad (7)$$

#### E. AUV

The AUV receives the control command  $\mathbf{u}$  from the motion controller, which is the reference for direction and velocity. Then, the AUV turns this reference  $\mathbf{u}$  into thrust commands to its thrusters  $\tau_C$ , using its internal lower-level controller. For this AUV, we consider the 6DOF model in which the state is the velocity vector given by  $\boldsymbol{\nu} = [u \ v \ w \ p \ q \ r]^\top$  containing the translational and rotational velocities. Note that these elements of the velocity vector are directly influencing the AUV's position  $\mathbf{p}$ .

The dynamics of the AUV can be formulated as a nonlinear system represented by a vectorial notation presented by Fossen [24] as follows:

$$\begin{aligned} (M_{RB} + M_A)\dot{\boldsymbol{\nu}} + (C_{RB}(\boldsymbol{\nu}) + C_A(\boldsymbol{\nu}))\boldsymbol{\nu} + \\ \mathbf{D}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{g}(\boldsymbol{\eta}) = \boldsymbol{\tau}_C, \quad (8) \end{aligned}$$

where  $M_{RB}$  is the rigid body mass and inertia matrix and  $C_{RB}$  is the matrix of Coriolis and centripetal terms on the left hand side.  $M_A$  and  $C_A(\boldsymbol{\nu})$  represent the effect of added mass,  $\mathbf{D}(\boldsymbol{\nu})$  represents the damping matrix and  $\mathbf{g}(\boldsymbol{\eta})$  is the vector of gravitational and buoyancy forces and moments.  $\boldsymbol{\tau}_C$  is a vector of external control forces based on the AUV's actuator configuration.

#### F. Motion Controller

The control law we propose is summarised in Fig. 4. It relies on the chlorophyll  $a$  gradient  $\nabla \delta$  and the latest chlorophyll  $a$  concentration measurement  $\delta$  to produce a control command  $\mathbf{u}$ . First, we define a front as a level set of a time-varying scalar field  $\delta: \mathbb{R} \times \mathbb{R}^2 \rightarrow \mathbb{R}$ :

$$F(t) = \{\mathbf{p} \in \mathbb{R}^2 : \delta(t, \mathbf{p}) = \delta_{\text{ref}}\}, \quad (9)$$

where  $\delta_{\text{ref}}$  is some reference value,  $\mathbf{p}$  the position and  $t$  time.

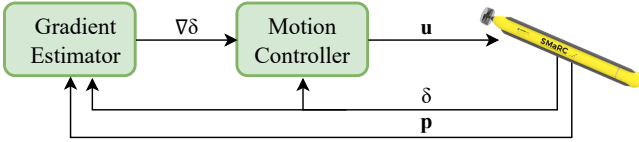


Fig. 4: Control architecture with the motion controller, the gradient estimator, and the AUV.

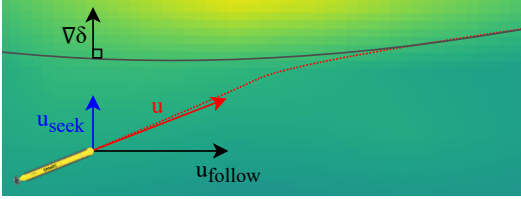


Fig. 5: Seek and follow components of the control law and gradient.

Assume that the reference value  $\delta_{\text{ref}}$  is known, then, we used the previously developed control law as in [25]. There, we define the control law as

$$\begin{aligned} \mathbf{u}(t, \mathbf{p}) &= \mathbf{u}_{\text{seek}}(t, \mathbf{p}) + \mathbf{u}_{\text{follow}}(t, \mathbf{p}) \\ \mathbf{u}_{\text{seek}}(t, \mathbf{p}) &= -\alpha_{\text{seek}}(\delta(t, \mathbf{p}) - \delta_{\text{ref}})\nabla\delta(t, \mathbf{p}) \\ \mathbf{u}_{\text{follow}}(t, \mathbf{p}) &= \alpha_{\text{follow}}R_{\pi/2}\nabla\delta(t, \mathbf{p}), \end{aligned} \quad (10)$$

where  $\nabla\delta$  is the gradient of  $\delta$  with respect to  $\mathbf{p}$ ,  $R_{\pi/2}$  is a mapping which rotates vectors by 90 degrees, and  $\alpha_{\text{seek}}$  and  $\alpha_{\text{follow}}$  are tunable parameters.

As seen in Fig. 5, the control law consists of two components:  $\mathbf{u}_{\text{seek}}$ , which controls the AUV towards the front by following the gradient field, and  $\mathbf{u}_{\text{follow}}$  which controls the AUV to move along the front, perpendicularly to the gradient field. By designing the control law with these two components, we ensure convergence to the front [26]. Namely, note that if  $\delta(t, \mathbf{p}) \neq \delta_{\text{ref}}$ , the  $\mathbf{u}_{\text{seek}}$  component grows proportionally to this difference, making seeking the front a priority, comparing to following the front. On the other hand, if the AUV is on the front, the most prominent component becomes the  $\mathbf{u}_{\text{follow}}$ .

### III. EXPERIMENTAL EVALUATION

In this section, the proposed system architecture and its corresponding control and estimation components are tested in a scenario of an algal bloom front tracking mission. We first define the experimental setup for this section and then present and analyse the numerical results.

#### A. Experimental Setup

For this scenario, we consider the environment illustrated in Fig. 3 in which we will deploy the AUV and track the algal bloom front. Here, the chlorophyll  $a$  concentration is represented by a map that goes gradually from high concentration in yellow to low concentration in blue, these values will be measured by the chlorophyll  $a$  concentration sensor mounted on the AUV. More specifically, the simulated mission will occur inside of the red square. The data we use to simulate this environment has a spatial resolution of 300 m by 300 m [27], and the location is the same one as the satellite data we considered earlier.

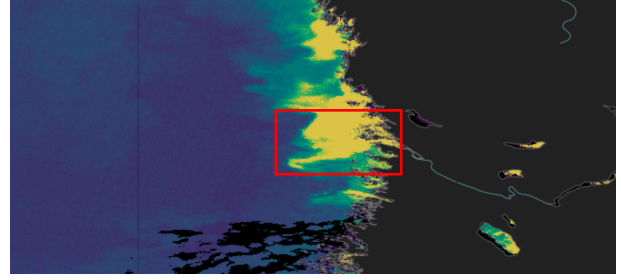


Fig. 6: CMEMS data of chlorophyll  $a$  concentration in the Baltic Sea (blue - yellow), clouds and cloud coverage (black), and land (dark grey).

The source code that implements the algorithm is available as an open-source contribution on two repositories. The first one is the Gaussian Processes for Adaptive Environmental Sampling (GP4AES) library which includes the GP model estimator, the gradient estimator, and the motion controller <https://github.com/JoanaFonsec/gp4aes>. The second one is the ROS [28] interface which uses the GP4AES library and handles the connection with the AUV's software <https://github.com/JoanaFonsec/algalbloom-tracking>.

The simulation starts by deploying the AUV close to the front and providing it an initial gradient estimation. When the AUV is near the front, the gradient estimation is triggered. The AUV speed is fixed to 1 m/s. For that reason, we are interested only in the ratio of  $\alpha_{\text{seek}}$  and  $\alpha_{\text{follow}}$ , and thus the latter is set to 1. Moreover, we consider  $\delta_{\text{ref}} = 7.45 \text{ mg/m}^3$ , based on the available satellite data. While tracking the front, the AUV collects a measurement at  $f = 1 \text{ Hz}$ , considering a standard deviation of the measurement noise of  $\sigma = 10^{-3} \text{ mg/m}^3$ . The measurements are filtered using a weighted moving average filter of size 3, with  $w = [0.2, 0.3, 0.5]$ :

$$\delta_{\text{filtered}}(t) = w_{-2}\delta(t-2) + w_{-1}\delta(t-1) + w_0\delta(t). \quad (11)$$

With the same sampling rate, the gradient is estimated as in (6), using data from the last  $n = 200$  measurements. Then we apply a first order low pass filter, with  $\alpha = 0.97$ ,

$$\nabla\delta_{\text{filtered}}(t) = \alpha\nabla\delta(t-1) + (1-\alpha)\nabla\delta(t). \quad (12)$$

The parameters described are summarised in Table I.

$\sigma$	$\alpha_{\text{seek}}$	$v$	$n$	$\delta_{\text{ref}}$	$\alpha$
$10^{-3} \text{ mg/m}^3$	20	1 m/s	200	$7.45 \text{ mg/m}^3$	0.97

TABLE I: Control algorithm parameters.

#### B. Numerical Results

We illustrate the complete AUV mission in Fig. 7 where the AUV is following the front while collecting chlorophyll  $a$  concentration measurements, estimating the chlorophyll  $a$  concentration field and its field gradient, and updating its direction. The complete mission has a duration of 33 hours, approximately. The starting position is far from the bloom and represented by the white star while the final position is on the front and represented by the white square. In this figure, the AUV seems to follow the algal bloom front closely. This is further analysed in the Fig. 8, in which we focus and zoom-in in the area inside of the blue square.



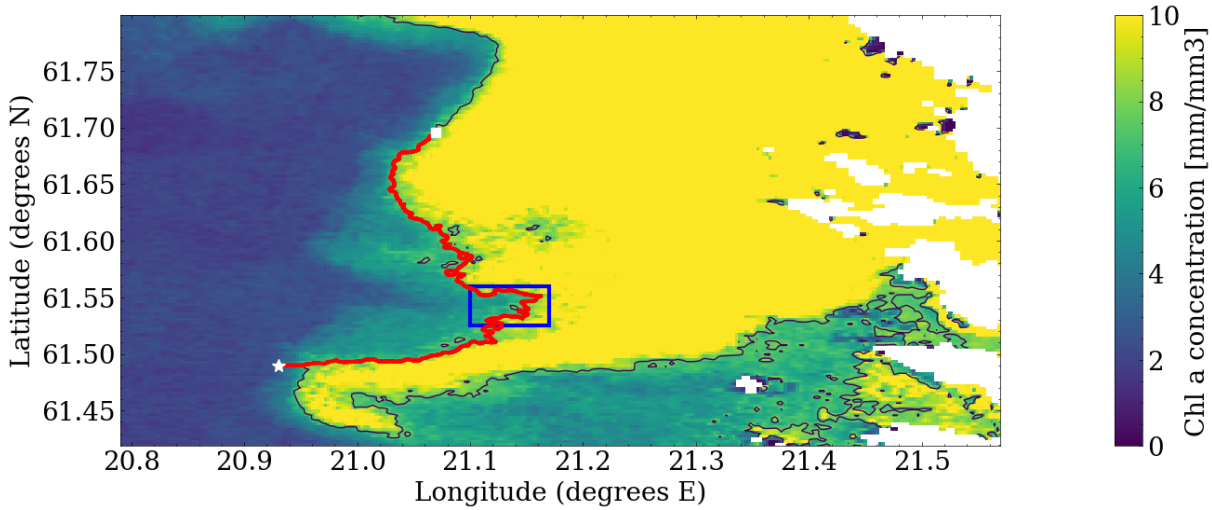


Fig. 7: Overview of the full mission having the trajectory of the AUV (red) tracking the front (black) in the chlorophyll  $a$  field (blue-yellow). The white star indicates the initial position and the white square the final position.

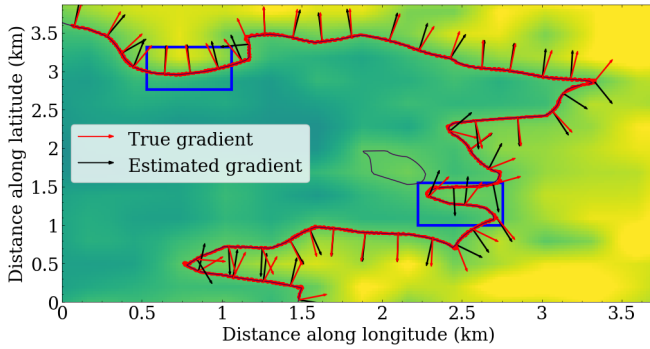
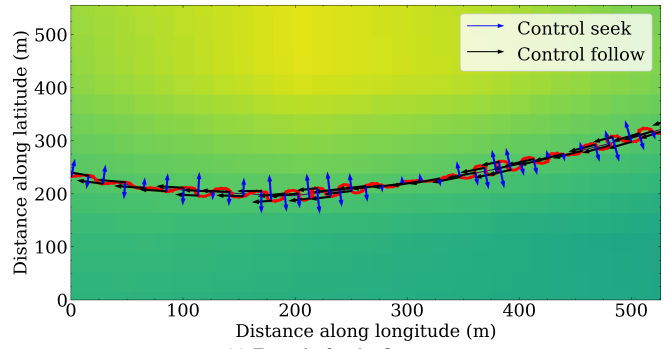


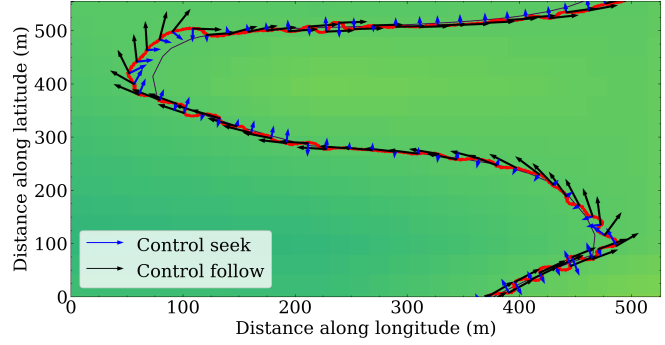
Fig. 8: Zoomed-in trajectory of the AUV (red) tracking the front (black), with arrows representing the true and estimated gradient

In Fig. 8 we zoom-in in a region of the mission, previously defined by a blue square, to focus on the performance of the gradient estimation. This region corresponds to about 7 hours of mission time. Here we illustrate the gradient performance through arrows representing both the true and estimated gradients along the path. The true gradient refers to the gradient that the AUV would be able to compute if it had access to the global information of the field. We compute it by taking the spatial derivative of the chlorophyll  $a$  concentration field. The estimated gradient refers to the output of the gradient estimator, as in (7). The angle between the true and estimated gradient arrows indicates the gradient error. However, in this scenario, the chlorophyll  $a$  concentration field is non-convex and fast-changing even in small areas. Therefore, the gradient is an abstraction that gives an idea of direction rather than an exact measure of the gradient. In fact, if we analyse the straight portions of the path we would say that the error is very close to zero, while by analysing the portions of the path with higher curvature, we could say that the error is larger while the gradient looks ambiguous and sensitive to small changes in position. As for the control performance, this figure doesn't allow for such analysis so we zoom-in in on the two areas inside the blue squares.

Let us now analyse the performance of the control and its control components in Fig. 9. These figures correspond



(a) Zoom-in for the first area



(b) Zoom-in for the second area

Fig. 9: Zoom-in trajectory of the AUV (red) tracking the front (black), with arrows representing seek and follow components of the control law

to the two zoom-in locations of the previous figure, the front is the thin black line and the AUV path is represented by a thicker red line. We also plot the seek and follow components of the control law using arrows along the AUV path. The control law is constructed as in eq.10, and it is a sum of the seek component which has the same direction as the estimated gradient and the follow component which has a perpendicular direction with respect to the estimated gradient. This sum constitutes the control law corresponding to the AUV's direction of movement. In the first zoom-in, in Fig. 9a, the AUV always follows the front closely and with minimal error. This is expected as the front is smooth in curvature and the AUV remains on top of the front. The

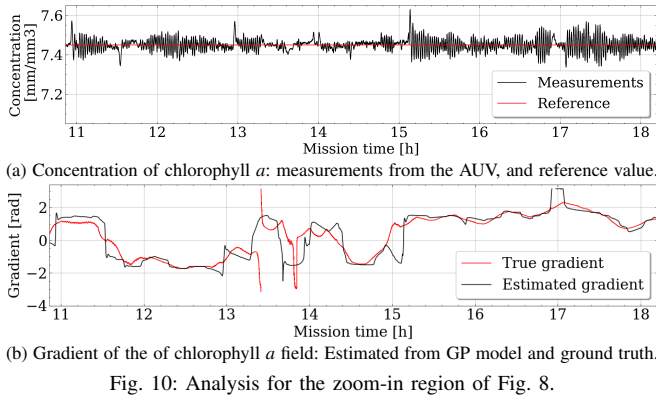


Fig. 10: Analysis for the zoom-in region of Fig. 8.

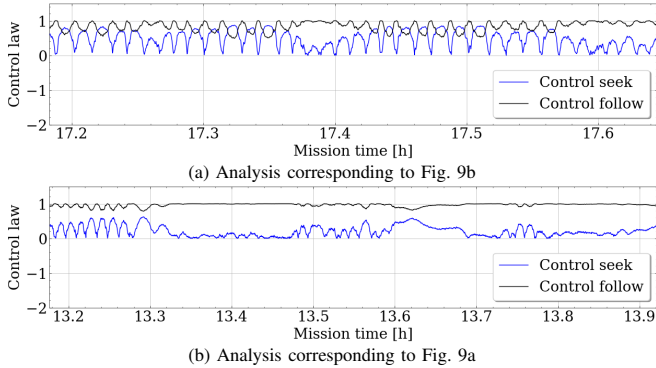


Fig. 11: Control law components: seek and follow, for the region in Fig. 9b

control seek component accounts for small adjustments in the trajectory. In the second zoom-in, in Fig. 9b, the AUV remains on top of the front most of the time, and thus the control follow component dominates the control law. On the other hand, once the curvature changes at a faster pace, the control follow component reduces and the control seek becomes the dominating component. In this scenario of fast-changing curvature, the AUV seems to have a delay in updating its direction. There are two leading causes for this behavior: the AUV’s turning radius and the update function with the update rate on the gradient. The gradient’s update function in (12) introduces a delay and a cut-off frequency. This cut-off frequency is inversely proportional to the update rate so the performance of the algorithm becomes a trade-off between the smoothness introduced by the update function with a smaller update rate and the delay introduced by it. For this scenario, we considered that smoothness of movement is a more important objective than the apparent delay in the tighter curvature of the front.

We further analyse the algorithm’s behavior through time series plots in Fig. 10 corresponding to the zoomed-in area in Fig. 8. First, we consider the chlorophyll  $a$  concentration measurements taken along the path in Fig. 10a. The time series indicates that, as seen in the previous figures, the AUV is always on top of the front, oscillating around it and with a minimal error, in this case, lower than  $\pm 0.1 \text{ mg/m}^3$ . Second, we consider the gradient field estimation also taken along the mission, in Fig. 10b. This time series also confirms what we saw in the previous figures. Here we can see clearly both the delay of the estimated gradient and its smoothness, in comparison with the true gradient. Finally, let us further

analyse the control law, considering the time series of the two control components in Fig. 11, for the region in Fig. 9. For the first zoom-in area, which considers an almost linear segment of the front, we get a small oscillation in Fig. 11a. This corresponds to the zig-zag around the curve that we see in the corresponding path figure. Whereas in the second zoom-in, in which we consider a segment with two tight curves, Fig. 11b, we see different patterns that represent oscillation around the front, followed by constant, almost linear tracking with the control follow component dominating. For both examples the follow component dominates the control law, and the seek component grows when the AUV is far from the front.

## IV. CONCLUSIONS

We considered the problem of how to use satellite data to improve adaptive sampling missions of an AUV equipped with a chlorophyll  $a$  concentration sensor. Our solution used GPs to model a chlorophyll  $a$  concentration front, from which we derived the chlorophyll  $a$  concentration field gradient, and integrated into a front tracking algorithm. The front tracking algorithm was tested on a simulated environment and resulted in the AUV finding and tracking the front closely. Our future plan is to run experiments using this method. This will consist on testing and improving the method implemented in ROS, running more simulations on *Stonefish*, integrating the chlorophyll  $a$  sensor on the AUV and, finally, running experiments on a local front in the summer of 2023 in the Stockholm archipelago.

## REFERENCES

- [1] S. Shumway, J. M. Burkholder, and S. Morton, *Harmful Algal Blooms: A Compendium Desk Reference*, 2018.
- [2] “Copernicus webpage,” <https://marine.copernicus.eu/about-us/about-eu-copernicus>, accessed: 2022-09-30.
- [3] NOAA. What is a harmful algal bloom? [Online]. Available: <https://www.noaa.gov/what-is-harmful-algal-bloom>
- [4] N. Wasmund, *Harmful Algal Blooms in Coastal Waters of the South-Eastern Baltic Sea*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2002, pp. 93–116. [Online]. Available: [https://doi.org/10.1007/978-3-662-04769-9\\_8](https://doi.org/10.1007/978-3-662-04769-9_8)
- [5] G. Hallegraeff, “Harmful algal blooms: a global overview,” *Manual on harmful marine microalgae*, vol. 33, pp. 1–22, 2003.
- [6] W. Munk, “Testimony to the u.s. commission on ocean policy.” 2002. [Online]. Available: [http://govinfo.library.unt.edu/oceancommission/meetings/apr18\\_19\\_02/munk\\_statement.pdf](http://govinfo.library.unt.edu/oceancommission/meetings/apr18_19_02/munk_statement.pdf)
- [7] V. Martinez-Vicente, A. Kurekin, C. Sá, V. Brotas, A. Amorim, V. Veloso, J. Lin, and P. I. Miller, “Sensitivity of a satellite algorithm for harmful algal bloom discrimination to the use of laboratory bio-optical data for training,” *Frontiers in Marine Science*, vol. 7, 2020.
- [8] R. Millet, F. Plumet, and J.-C. Dorn, “Autonomous surface vehicle for oceanographic survey,” 2008.
- [9] Z. Liu, Y. Zhang, X. Yu, and C. Yuan, “Unmanned surface vehicles: An overview of developments and challenges,” *Annual Reviews in Control*, vol. 41, pp. 71 – 93, 2016.
- [10] F. P. Chavez, J. T. Pennington, R. Herlien, H. Jannasch, G. Thurmond, and G. E. Friederich, “Moored and drifters for real-time interdisciplinary oceanography,” *Journal of Atmospheric and Oceanic Technology*, vol. 14, no. 5, pp. 1199–1211, 1997.
- [11] J. S. Willcox, J. G. Bellingham, Y. Zhang, and A. B. Baggeroer, “Performance metrics for oceanographic surveys with autonomous underwater vehicles,” *IEEE Journal of Oceanic Engineering*, vol. 26, no. 4, pp. 711–725, 2001.

- [12] J. Das, F. Py, T. Maughan, T. O'reilly, M. Messié, J. Ryan, G. S. Sukhatme, and K. Rajan, "Coordinated sampling of dynamic oceanographic features with underwater vehicles and drifters," *The International Journal of Robotics Research*, vol. 31, no. 5, pp. 626–646, 2012.
- [13] O. Özkahraman and P. Ögren, "Efficient navigation aware seabed coverage using auvs," in *2021 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, 2021, pp. 63–70.
- [14] P. Ögren, E. Fiorelli, and N. E. Leonard, "Cooperative control of mobile sensor networks: Adaptive gradient climbing in a distributed environment," *IEEE Transactions on Automatic control*, vol. 49, no. 8, pp. 1292–1302, 2004.
- [15] E. Fiorelli, N. E. Leonard, P. Bhatta, D. A. Paley, R. Bachmayer, and D. M. Fratantoni, "Multi-auv control and adaptive sampling in monterey bay," *IEEE journal of oceanic engineering*, vol. 31, no. 4, pp. 935–948, 2006.
- [16] J. Hwang, N. Bose, and S. Fan, "Auv adaptive sampling methods: A review," *Applied Sciences*, vol. 9, no. 15, p. 3145, 2019.
- [17] S. McCammon, G. Marcon dos Santos, M. Frantz, T. P. Welch, G. Best, R. K. Shearman, J. D. Nash, J. A. Barth, J. A. Adams, and G. A. Hollinger, "Ocean front detection and tracking using a team of heterogeneous marine vehicles," *Journal of Field Robotics*, vol. 38, no. 6, pp. 854–881, 2021.
- [18] R. N. Smith, Y. Chao, P. P. Li, D. A. Caron, B. H. Jones, and G. S. Sukhatme, "Planning and implementing trajectories for autonomous underwater vehicles to track evolving ocean processes based on predictions from a regional ocean model," *The International Journal of Robotics Research*, vol. 29, no. 12, pp. 1475–1497, 2010.
- [19] J. Das, F. Py, J. B. Harvey, J. P. Ryan, A. Gellene, R. Graham, D. A. Caron, K. Rajan, and G. S. Sukhatme, "Data-driven robotic sampling for marine ecosystem monitoring," *The International Journal of Robotics Research*, vol. 34, no. 12, pp. 1435–1452, 2015.
- [20] "Copernicus dataset for the 2km resolution data, *dataset-bal-analysis-forecast-bio-dailymeans*," [https://resources.marine.copernicus.eu/product-detail/BALTICSEA\\_ANALYSISFORECAST\\_BGC\\_003\\_007/INFORMATION](https://resources.marine.copernicus.eu/product-detail/BALTICSEA_ANALYSISFORECAST_BGC_003_007/INFORMATION), accessed: 2022-09-30.
- [21] A. Bilaletdin, H. Kaipainen, and T. Frisk, "Dynamic nutrient modelling of a large river basin in finland," vol. 111, 05 2008, pp. 53–61.
- [22] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*. Cambridge, MA: MIT Press, 2006.
- [23] P. Erickson, M. Cline, N. Tirpankar, and T. Henderson, "Gaussian processes for multi-sensor environmental monitoring," in *2015 IEEE International conference on multisensor fusion and integration for intelligent systems (MFI)*. IEEE, 2015, pp. 208–213.
- [24] T. Fossen, *Handbook of Marine Craft Hydrodynamics and Motion Control*. John Wiley & Sons Ltd., April 2011.
- [25] J. Fonseca, M. Aguiar, J. B. d. Sousa, and K. H. Johansson, "Algal bloom front tracking using an unmanned surface vehicle: Numerical experiments based on baltic sea data," in *OCEANS 2021: San Diego – Porto*, 2021, pp. 1–7.
- [26] J. Fonseca, J. Wei, T. A. Johansen, and K. H. Johansson, "Cooperative circumnavigation for a mobile target using adaptive estimation," in *CONTROL 2020*. Springer International Publishing, 2021, pp. 33–48.
- [27] "Copernicus dataset for the 300m resolution data, *dataset-oc-bal-chl-olci-l3-nn\_300m-daily-rt*," [https://resources.marine.copernicus.eu/product-detail/OCEANCOLOUR\\_BAL\\_CHL\\_L3\\_NRT\\_OBSERVATIONS\\_009\\_049/INFORMATION](https://resources.marine.copernicus.eu/product-detail/OCEANCOLOUR_BAL_CHL_L3_NRT_OBSERVATIONS_009_049/INFORMATION), accessed: 2022-09-30.
- [28] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, A. Y. Ng *et al.*, "Ros: an open-source robot operating system," in *ICRA workshop on open source software*, vol. 3, no. 3.2. Kobe, Japan, 2009, p. 5.