Proceedings of the 2023 Winter Simulation Conference
C. G. Corlu, S. R. Hunter, H. Lam, B. S. Onggo, J. Shortle, and B. Biller, eds.

SIMULATION, OPTIMIZATION AND CONTROL OF TRAJECTORIES OF ASVS PERFORMING HACBS MONITORING MISSIONS IN LENTIC WATERS

Alfredo González-Calvin
Lía García-Perez
José L. Risco-Martín
Eva Besada-Portas

Department of Computer Architecture and Automation
Universidad Complutense de Madrid
Calle Prof. José García Santesmases, 9
28040 Madrid, SPAIN

ABSTRACT

Harmful Algae and Cyanobacteria Blooms (HACBs) are dangerous dynamic processes for the users/inhabitants of the hydric resources. Their development and contingency plans can be anticipated by using Autonomous Surface Vehicles (ASVs) equipped with a self-driven system capable of deciding how to displace the ASV and its multi-parametric probe to take measurements in the 3D locations of the water body where the HACB is likely to occur. This paper presents a new self-driven system for that purpose, consistent on 1) an offline trajectory planner for the ASV that exploits the information provided by a commercial HACBs simulator to optimize, in turn, the ASV horizontal and probe vertical displacements; and 2) a guidance and control system specially designed for making the ASV follow the planned trajectories. The paper also presents a comprehensive set of simulations to evaluate our proposal's performance and adjust its parameters.

1 INTRODUCTION

The sixth goal of the United Nations 2030 Agenda for Sustainable Development is to ensure the availability and sustainable management of water and sanitation for all (United Nations 2023). This objective is motivated by an alarming growth of the degradation of water-related ecosystems and the lack of monitoring of the water quality used by billions of people. Reducing water misuse and pollution; mitigating climate change effects such as extreme floods, droughts, and global warming; and carefully controlling/managing the water resources will not only strengthen the resilience of watersheds, but will also help national health systems and economies to fulfill their populations needs (United Nations Environment Programme 2023).

The need for autonomous real-time water monitoring systems is of particular concern in developing countries. The situation in developed countries, with automatic monitoring stations located upstream of water treatment stations or as elements of the early warning systems of their river basins, also requires improvement. This happens because these means, located in fixed positions, do not provide information with the spatio-temporal resolution necessary to know the local state of the variables of interest in the different elements of the hydrographic system (e.g., rivers, reservoirs, lakes, or marshes).

This situation is especially problematic when the pollutants to be detected are Harmful Algae Blooms (HABs, Shumway et al. 2018) or Cyanobacteria Blooms (CBs, Meriluoto et al. 2017), which occur when algae or cyanobacteria colonies grow uncontrollably. On the one hand, HABs can 1) consume much the oxygen in the water as they decay, 2) smooth fish and submerged aquatic vegetation, 3) create smelly heaps/scums on the shore/water surface, and 4) produce toxic or harmful substances for the fauna inhabiting the aquatic environment or the people using the water resource (for drinking or recreation). On the other

hand, CBs share the proprieties of the HABs and add their capability to appear and disappear from the water surface (by displacing themselves vertically in the water column) and the extreme toxicity of their secondary metabolites, which make them a relevant public health issue. Finally, the dynamic behavior of both (which will be called here-after HACBs) is related, among other causes, to their biological growth, to the competition between colonies, and to the water currents and wind that move them in the three-dimensional (3D) space of the water body. Hence, the awareness of the water authorities of the toxicity and complex behavior of HACBs are hardening the international regulations related to water (e.g., those in United States Environmental Protection Agency 2021 and in European Commission 2021), which currently indicate that HACBs should be monitored more frequently and at more locations to be able to anticipate their presence and improve the management of the water bodies where they appear.

The fulfillment of the indications of the new regulations can be partially supported by the use of Autonomous Surface Vehicles (ASVs, a type of robotized boat) specially designed to take the multi-parametric probes that measure the variables under study to the points of interest within the water body (Hitz et al. 2012; Siyang and Kerdcharoen 2016; Girón-Sierra and Chacón-Sombría 2021). Moreover, to make an ASV an intelligent self-driving 3D monitoring platform, it is also necessary to equip it with 1) a motorized reel in charge of moving the probe vertically and 2) a complete automation system (incorporating advanced localization, sensing, planning, guidance and control techniques) responsible for deciding how to move the ASVs (and its probe) and for moving it safely (Liua et al. 2016). This paper focuses on developing several elements of the complete automation system for HACBs monitoring, and analyzing their performance through multiple simulations.

The planning, guidance, and control of an ASV depend largely on the available information about the water body and the phases contemplated by its automation system. For instance, when there is no information about the water body, the ASVs displacements can be directly driven/controlled (without a pre-planning step) by the measurements taken by their onboard probes (de Marina et al. 2021; Besada-Portas et al. 2021), or pre-planned as a trajectory (to be followed by the ASV during the monitoring mission) by solving a coverage problem with different types of patterns (Valada et al. 2012; N. Karapetyan and Rekleitis 2019) or as a Travelling Salesman Problem among shore points (Arzamendia et al. 2016; Arzamendia et al. 2019). Otherwise, when the information to be gathered in the regions of interest is relevant, the ASV trajectory can be pre-planned by setting up an information gain problem and solving it with different evolutionary optimizers (Xiong et al. 2019; Xiong et al. 2020). Another alternative consists of building online the trajectory of the ASVs according to water quality uncertainty models adjusted to the information gathered by the ASVs in previous monitoring missions and updated with the ASVs measurements obtained during the current one (Kathen et al. 2021; Peralta et al. 2023). Finally, it is possible to exploit the information provided by simulators of the water body dynamics and of the HACBs dispersion to pre-plan 3D trajectories for the ASV/probe horizontal/vertical displacements that maximize the chances of detecting them while minimizing the mission time and trajectory length (Carazo-Barbero et al. 2021). The trajectory pre-planner presented in this paper falls within this last category, differing from Carazo-Barbero et al. (2021) by 1) optimizing a 3D probe trajectory that alternates ASVs displacements with the probe fixed at a constant depth and vertical profiles of the probe while keeping the ASV fixed at a given location, and by 2) exploiting a commercial simulator of the water body and HACB.

The approaches presented in the previous paragraph can be further classified into two categories. The first two ones (i.e., de Marina et al. 2021 and Besada-Portas et al. 2021) directly control the ASV based on the information provided by the probes without planning any trajectory. In contrast, the remaining ones are centered on trajectory planning without considering how to make the ASV follow the trajectory. That is, they are focused on the development of two *complementary* parts of the complete automation systems, since the performance of the guidance and control (G&C) usually improves when it is programmed/fitted for its particular intent (in this case, making an ASV follow a given type of trajectory). Hence, this paper also presents the G&C designed for our automation system, which is partially based on the Vector-Field guided Path-Following (VF-PF) algorithm presented by de Marina et al. (2017), where the guiding vector field is

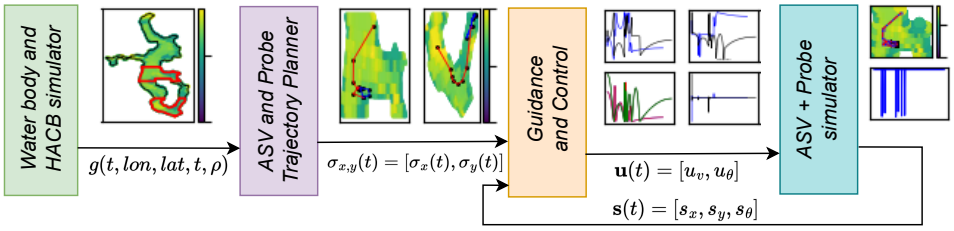


Figure 1: Schematic representation of the relationships between the different elements

a) designed to make the field integral curves converge to the desired path and b) used to derive the control law for the vehicle. Besides, to adapt the path-following part of VF-PF for the trajectory tracking required by our G&C, we follow the strategy presented by Yao et al. (2021). Other trajectory followers/trackers that could be of interest to our G&C are the ones introduced by: de la Cruz et al. (2015), which is capable of making the ASV adapt its movement to any trajectory of changing curvature; Wang et al. (2022) and Fan et al. (2023), which is supported by sliding mode controllers; and Liu et al. (2021), which is based on a line-of-sight guidance algorithm commonly used for ASV path-following.

In summary, the three main contributions of this paper are the following. Fist, an offline planner that exploits the information of a commercial water body and HACB simulator to optimize, in turns, the ASV horizontal trajectory and probe vertical displacement. Second, a G&C especially designed to make the ASV follow the trajectory returned by the planner as closely as possible. And finally, the analysis of the performance of both subsystems under simulations, which is extremely important to support the different phases of our research by helping us, for example, to tune the parameters and validate the different systems, characterize their behavior as a whole, and avoid later experimental problems.

The rest of the paper is organized as follows. Sections 2, 3 and 4 describe the main properties of the water body and HACB simulator, of the USV trajectory planner, and the G&C, which are respectively used and presented in this paper for HACBs monitoring missions, while Figure 1 schematizes the relationships among them (and anticipates the name of some variables that will be used through the paper). Section 5 shows several simulations with the purpose of characterizing the system's behavior. Finally, Section 6 draws the conclusions and presents some future lines of research.

2 WATER BODY AND HACB SIMULATOR

The simulator of the 3D water body and HACB behavior must include, at least, models of the fluid dynamics, and the HACB growth and 3D transportation. When considering those requirements, many existing simulators related to water quality ecosystems are not valid since they work in lower dimensions (e.g., CE-QUAL-W2 , DYRESM-CAEDYM , GLM , or ATK, LAKE2K and QUAL2K), or do not include hydrodynamics (e.g., WASP) or eutrophic modules (e.g., ECOMSED or UNTRIM). Besides, others (e.g., EFDC-EPA or Delft3D) are too difficult to use since they lack a Graphical User Interface (GUI) to specify the initial and contour conditions of all their modules.

After discarding the previous options, we decided to use EEMS , which extends EFDC-EPA by including a GUI, and by supporting new functionalities such as the capability of simulating several kinds of algae and cyanobacteria. However, it is worth noting that EEMS discretizes the water environment into a 3D grid and the time evolution in periodic steps, in order to perform the simulations and manage the data storage. However, our ASV + probe trajectory planner needs to evaluate the state of some given variables (e.g., algae or cyanobacteria concentration) at any time instant t and 3D location (lon, lat, z) of the water body. Hence, we have implemented a function $g(t, lon, lat, z, v)$ that obtains the value of the variable of interest v at the given (t, lon, lan, z) by interpolating the value of v linearly in the time, and linearly or with the nearest neighbors in the 3D space.

3 TRAJECTORY PLANNER

This sections explains the main characteristics of our planner, starting with a description of the trajectory encoding, following with the criteria used to evaluate the trajectories manipulated by the planner, and ending with the main characteristics of the optimizer used to determine a feasible overall good trajectory.

3.1 Trajectory Encoding

The 3D trajectory to be optimized alternates ASVs displacements with the probe depth fixed and vertical profiles of the probe with the ASV fixed at its current location. The planner optimizes this type of trajectory (instead of a 3D trajectory with simultaneous ASV and probe displacements as in Carazo-Barbero et al. 2021) to combine the two types of displacements that are usually performed when sampling the water manually. Besides, as the N_v probe vertical profiles are all equal except for the ASV locations from where they are performed, the 3D probe trajectory can be built up from the ASV horizontal trajectory.

The ASV 2D trajectory $\boldsymbol{\sigma}(t) = [\sigma_{lon}(t), \sigma_{lat}(t)]^T$ is encoded as a natural cubic spline (Ravanka et al. 2018), constructed using $N_f = N_v + 1$ third order polynomials for each dimension, each of them 1) defining the value of the corresponding coordinate between two given time steps (i.e., $f_{i,l}(t) = \sigma_l(t)$ when $t_{i-1} < t < t_i$, $i \in \{1 : N_f\}$ and $l \in \{lon, lat\}$) and 2) having to pass at t_i through one of the $i \in \{1 : N_v\}$ intermediate control points $\boldsymbol{c}_i = [c_{i,lon}, c_{i,lat}]$ of the spline (i.e., $f_{i,l}(t_i) = f_{i+1,l}(t_i) = c_{i,l}$, with $i \in \{1 : N_v\}$ and $l \in \{lon, lat\}$). The spline is also required to have C^2 continuity (i.e., $\dot{f}_{i,l}(t_i) = \dot{f}_{i+1,l}(t_i)$ and $\ddot{f}_{i,l}(t_i) = \ddot{f}_{i+1,l}(t_i)$ for $i \in \{1 : N_v\}$ and $l \in \{lon, lat\}$) and a null second derivative in its initial and end locations. This encoding is chosen since 1) its C^2 continuity makes the ASV position, velocity, and acceleration change smoothly; 2) each $\boldsymbol{c}_i$ can be used as the ASV fixed location for a vertical probe profile; and 3) when the ASV starts at a point that is not the first of the spline, it follows a straight line to start the trajectory.

The planner optimizes the location of the N_v spline control points $\boldsymbol{c}_i$ and their passing time t_i, manipulating $3 \cdot N_v$ decision variables. With that information, obtaining the ASV trajectory at any instant between $t_0 < t < t_{N_f}$ is possible. Nevertheless, this trajectory needs to be discretized and extended to the third dimension (including the probe depth) for evaluation. To do it, we carry out the following process. First, we discretize with a sampling period of T_{UAV} each segment of the spline $f_{i,l}(t)$, valid from t_{i-1} to t_i, and store the information in the auxiliary vectors $\boldsymbol{v}_{i,l} = [\sigma_{i,l}(t_{i-1}), \sigma_{i,l}(t_{i-1} + T_{UAV}), \sigma_{i,l}(t_{i-1} + 2 \cdot T_{UAV}), \ldots, \sigma_{i,l}(t_i)]$ and $\boldsymbol{v}_{i,\Delta T} = [0, T_{UAV}, 2 \cdot T_{UAV}, \ldots, t_i - t_{i-1}]$. Next, we initialize the discretized location, probe, time, and type segment vectors with the information related to the first segment of the spline and the probe placed at fixed depth d^{probe}_{fixed}, (i.e., we make $\boldsymbol{p}_l = [\boldsymbol{v}_{1,l}]$ for $l \in \{lon, lat\}$, $\boldsymbol{p}_z = \text{repmat}(d^{probe}_{fixed}, 1, \text{length}(\boldsymbol{v}_{1,1}))$, $\boldsymbol{p}_{time} = [t_0 + \boldsymbol{v}_{1,\Delta T}]$ and $\boldsymbol{p}_{probe} = \text{repmat}(\text{fixed}, 1, \text{length}(\boldsymbol{v}_{1,1}))$). Afterwards, maintaining the last ASV location, we add the information related with the first downward vertical displacement of the probe, that starts at d^{probe}_{fixed} and moves every T_{probe} seconds a distance of Δ_{probe} during N_{probe} steps (by extending vectors $\boldsymbol{p}_l = [\boldsymbol{p}_l, \text{repmat}(\boldsymbol{p}_l(\text{end}), 1, N_{probe})]$ for $l \in \{lon, lat\}$, $\boldsymbol{p}_{probe} = [\boldsymbol{p}_{probe}, \text{repmat}(\text{moving}, 1, N_{probe})]$, $\boldsymbol{p}_{time} = [\boldsymbol{p}_{time}, \boldsymbol{p}_{time}(\text{end}) + T_{probe}, \boldsymbol{p}_{time}(\text{end}) + 2 \cdot T_{probe}, \cdots, \boldsymbol{p}_{time}(\text{end}) + N_{probe} \cdot T_{probe}]$ and $\boldsymbol{p}_z = [\boldsymbol{p}_z, d^{probe}_{fixed} + \Delta_{probe}, d^{probe}_{fixed} + 2 \cdot \Delta_{probe}, \ldots, d^{probe}_{fixed} + N_{probe} \cdot \Delta_{probe}]$). At this stage, the discretized vectors contain information about the spline's first segment and the first downward vertical probe displacement. Afterwards, all the remaining spline segments must be included, adding first the information of the current i segment of the spline (by making $\boldsymbol{p}_l = [\boldsymbol{p}_l, \boldsymbol{v}_{i,l}]$ for $i \in \{lon, lat\}$, $\boldsymbol{p}_z = [\boldsymbol{p}_z, \text{repmat}(d^{probe}_{fixed}, 1, \text{length}(\boldsymbol{v}_{1,1}))]$, $\boldsymbol{p}_{time} = [\boldsymbol{p}_{time}, \boldsymbol{p}_{time}(\text{end}) + N_{probe} \cdot T_{probe} + \boldsymbol{v}_{i,\Delta T}]$ and $\boldsymbol{p}_{probe} = [\boldsymbol{p}_{probe}, \text{repmat}(\text{fixed}, 1, \text{length}(\boldsymbol{v}_{i,1}))]$) and afterwards, except for the last segment of the spline, the downward vertical probe displacement (as explained before). In short, the process discretizes and alternates N_v pairs of ASV + probe displacements and ends with a last ASV displacement. Besides, in each segment inclusion (except for the initial ASV displacement), $\boldsymbol{p}_{time}$ gets delayed $2 \cdot N_{probe} \cdot T_{probe}$ seconds with respect to the original spline due to the downward and upward vertical probe displacements (the first half of the delay is accounted in the N_{probe}

Table 1: Planner evaluation criteria.

Type	Purpose	Expression
Const.	Maximal time duration	$\mathrm{CF}_1 = t_{N_f} - t_0 - 2 \cdot N_v \cdot N_{probe} \cdot T_{probe} \leq T_{mission}$
	Maximal ASV velocity	$\mathrm{CF}_2 = \max(\|\|\dot{\boldsymbol{p}}_{xy}(t_k)\|\|) \leq v_{max}, \forall t_k \in TimeSubset(\boldsymbol{p}_{tp}, fixed)$
	Spatial Domain	$\mathrm{CF}_3 = \sum_{t_k \in \boldsymbol{p}_{time}} OutSide(\boldsymbol{p}_{llz}(t_k)) = 0$
Objectives	Minimize ASV trajectory length	$\min \mathrm{OF}_1 = \sum_{t_k \in TimeSubset(\boldsymbol{p}_{tp}, fixed)} \|\|\dot{\boldsymbol{p}}_{xy}(t_k)\|\| \cdot T_{UAV}$
	Maximize HACB concentration during ASV displacements	$\max \mathrm{OF}_2 = \sum_{t_k \in TimeSubset(\boldsymbol{p}_{tp}, fixed)} g(t_k, p_{lat}(t_k), p_{long}(t_k), p_z(t_k), \rho)$
	Maximize HACB concentration during probe displacements	$\max \mathrm{OF}_3 = \sum_{t_k \in TimeSubset(\boldsymbol{p}_{tp}, moving)} g(t_k, p_{lat}(t_k), p_{long}(t_k), p_z(t_k), \rho)$

increments of T_{probe} seconds of the downward probe displacement, while the second half is accounted for in the $N_{probe} \cdot T_{probe}$ seconds delayed at the beginning of each UAV segment addition).

Finally, the spline and its previous discretization are codified in geographical coordinates to help the planner to decide if the trajectory is contained within the water body and to measure the concentration directly from the interpolator of the HACBs simulator. However, part of the trajectory evaluation process presented in the next section uses the ASV location and velocity in a Cartesian coordinate space centered at the beginning of the ASV trajectory $\boldsymbol{\sigma}(t_0)$. To obtain the values of the locations in the new coordinate space, the geographical coordinates are projected into a plane using equirectangular projection, an effective streamlined method for small distances compared to the Earth's radius. In particular, assuming that the Earth is an ellipsoid of revolution, of radius D meters in the poles and of R meters in the equator, and pre-calculating $\Delta p_l(t_k) = p_l(t_k) - \sigma_l(t_0)$ for $i \in \{lon, lat\}$, we can obtain $[p_x(t_k), p_y(t_k)] = \frac{1}{360}[\Delta p_{lon}(t_k)\cos(\sigma_{lat}(t_0)) \cdot R, \Delta p_{lat}(t_k) \cdot D]$. Finally, the ASV horizontal speeds ($\dot{p}_x(t_k)$ and $\dot{p}_y(t_k)$) are obtained by finite differences between two consecutive locations of the trajectory in the Cartesian coordinate space.

3.2 Evaluation Criteria

This section presents the Constraint and Optimization Functions (CF and OFs) that the planner uses to evaluate the feasibility and goodness of the trajectories. They are summarized in Table 1, where for shortness, $\boldsymbol{p}_{llz}(t_k) = [p_{lon}(t_k), p_{lat}(t_k), p_z(t_k)]$, $\boldsymbol{p}_{xy}(t_k) = [p_x(t_k), p_y(t_k)]$, $\boldsymbol{p}_{tp} = [\boldsymbol{p}_{time}, \boldsymbol{p}_{probe}]$, $OutSide(\boldsymbol{p}_{llz}(t_k)) = \{1 \; if \; p_{llz}(t_k) \notin \text{waberbody}, 0 \text{ otherwise}\}$ and $TimeSubset(\boldsymbol{p}_{tp}, v) = \{t_k \in \boldsymbol{p}_{time} | \boldsymbol{p}_{probe}(t_k) = v\}$. On one hand, the constraint function CF_1 ensures that the ASV and probe perform the complete monitoring mission in at most $T_{mission}$ hours; CF_2 guarantees that the trajectory is feasible from the ASV maneuverability point of view (i.e., constrained by the maximal velocity v_{max}); and CF_3 ensures that the trajectory is contained within the water body. On the other hand, the optimization function OF_1 minimizes the trajectory length (without accounting for the vertical probe displacement because it is the same for all the trajectories); OF_2 maximizes the accumulated observed algae or cyanobacteria concentration (ρ) along the ASV trajectory at constant depth; while OF_3 maximizes the accumulated observed ρ while the probe is moving downwards. In short, the planner is expected to select feasible short trajectories that pass over and make vertical profiles in regions of high algae or cyanobacteria concentration.

3.3 Optimizer

We decided to solve the constrained multi-objective optimization problem of obtaining the ASV trajectory for HACB monitoring with the Non-Dominated Sorting Genetic Algorithm (NSGA-II, Deb et al. 2002), since it is a well-known evolutionary approach successfully applied to many other real-world problems.

We configure NSGA-II as follows. In the initialization step, the values of the $3 \cdot N_v$ decision variables of a population of N_p solutions are sampled from a uniform distribution, bounded in the case of $\boldsymbol{c}_i$ between

the horizontal limits of the water-body and in the case of t_i between $[t_0, t_0 + T_{mission}]$. The parents' selection is made using binary tournament. The mating of each pair of parents is performed with a probability p_{cross} using simulated binary crossover with distribution index η_{cross} (Deb and Agrawal 1995). The values of some decision variables of some children (selected uniformly with mutation probability p_{mut}) are slightly modified using polynomial-based mutation with distribution index η_{mut} (Deb and Agrawal 1999). The population of the next generation is formed by the best N_p old and new solutions sorted in Pareto fronts and with the crowding distance. And the stop criterion is to reach a given number of generations N_{stop}. Finally, it is worth noting that after running NSGA-II, it is necessary to select one solution among all those returned by the planner that, belonging to the first Pareto front, are equally good.

4 ASV GUIDANCE AND CONTROL

This section explains the G&C approach that we have developed to make the ASV follow the planned trajectories, and which is inspired by the vector-field guided path-following (VF/PF) algorithm presented by Yao et al. (2021). Due to its closed-loop iteration with the ASV model, we start introducing this element.

4.1 ASV Model

A common approach in path-following problems is to consider that the control loop that acts over the system dynamics is fast and accurate enough to assume that the control inputs to the system model are the linear and angular velocities (Kapitanyuk et al. 2017). Under this assumption, the horizontal dynamics of the ASV can be modeled as a non-holonomic vehicle in two dimensions (de Marina et al. 2017), with the following kinematic expressions: $\dot{\boldsymbol{s}}(t) = [\dot{s}_x, \dot{s}_y, \dot{s}_\theta]^T = [u_v \cdot \cos(s_\theta) + w_x, u_v \cdot \sin(s_\theta) + w_y, u_\theta]^T$, where $\boldsymbol{s}(t) = [s_x, s_y, s_\theta]^T$ stands for the ASV state, $[s_x, s_y]^T$ for its location over the water body surface in the Cartesian coordinates, s_θ for its heading angle, $\boldsymbol{u} = [u_v, u_\theta]^T$ for the control vector input, u_v and u_θ for the ground and angular velocity control signals, and $\boldsymbol{w} = [w_x, w_y]^T$ for the water speed disturbances.

4.2 Law for the Angular Velocity Control Signal

This law, responsible for obtaining u_θ and for making the ASV state converge to the trajectory despite the disturbances $\boldsymbol{w}$ on its dynamics, is constructed in two steps (de Marina et al. 2017). In the first step, we define a vector field that will make the ASV converge to the desired trajectory, while in the second one, we identify the control law that will make the vehicle converge to the vector field.

In our case, the difficulty arises because the desired trajectory is encoded as a natural cubic spline, which is a parametrically-defined self-intersecting path. Overcoming its parametric definition by transforming it into implicit functions (to be used in the vector field computation) can be computationally expensive to be used often. Besides, Vector-Field Path-Following (VF-PF) algorithms such as the one presented by de Marina et al. (2017) can not handle self-intersecting trajectories because near the intersections, the vector field degenerates to zero and the guidance signal cannot be calculated. Nevertheless, both issues can be avoided following the strategy proposed by Yao et al. (2021), which extends the vector field by adding an extra dimension that allows the elimination of all singular points at once. The resultant expressions are summarized in Table 2 and explained next from a user perspective.

In the first column of the table, the desired path $\mathscr{P}$ is defined (using the generalized states of the ASV $\boldsymbol{\zeta} = [s_x, s_y, s_w]$, which include the additional dimension s_w) as the set where the error $\boldsymbol{e}_{x,y}$, defined as the difference between the ASV location and the spline evaluated at s_w (i.e., $\boldsymbol{e}_{x,y} = [\phi_x, \phi_y]^T = [s_x - \sigma_x(s_w), s_y - \sigma_y(s_w)]^T$) is equal to zero. To fulfill this requirement during the whole mission in spite of the water disturbances, we must define a vector field $\boldsymbol{\chi}$ and controller that makes $\boldsymbol{e}_{x,y} \to 0$ as the time evolves. For this purpose, the dynamics of the extra dimension may be modeled as $\dot{s}_w = \chi_3 / \sqrt{\chi_1^2 + \chi_2^2}$ (where χ_i are the elements of the extended guiding vector field) and added to the extended ASV's dynamical model. Besides, and as a consequence of having followed the methodology proposed by Yao et al. (2021), the new dimension and

Table 2: Controller equations.

Desired Path	Guiding Vector Field	Orientation Error	Control Signal
$\mathscr{P} = \{\boldsymbol{\zeta} \in \mathbb{R}^3 \mid \phi_x = 0, \phi_y = 0\}$	$\boldsymbol{\chi} = [\chi_1, \chi_2, \chi_3]^T$		$u_\theta = \dot{\theta}_d - k_\theta \boldsymbol{h}^T E \hat{\boldsymbol{\chi}}^P$
$\boldsymbol{\zeta} = [s_x, s_y, s_w]^T$	$\chi_1 = k_L^3 \dot{\sigma}_x(s_w) - k_L k_x \phi_x$	$\boldsymbol{e}_{ori}(t) = \begin{bmatrix} \boldsymbol{h}_\theta - \hat{\boldsymbol{\chi}}^P \\ \dot{s}_w - \frac{\chi_3}{\sqrt{\chi_1^2+\chi_2^2}} \end{bmatrix}$	$\dot{\theta}_d = -\frac{1}{\|\boldsymbol{\chi}^P\|} (\hat{\boldsymbol{\chi}}^P)^T E J (\boldsymbol{\chi}^P) \dot{\boldsymbol{\zeta}}$
$\phi_x = s_x - \sigma_x(s_w)$	$\chi_2 = k_L^3 \dot{\sigma}_y(s_w) - k_L k_y \phi_y$	$\boldsymbol{h}_\theta = [\cos(s_\theta), \sin(s_\theta)]^T$	E is the 90 degrees rotation matrix
$\phi_y = s_y - \sigma_y(s_w)$	$\chi_3 = k_L^3 - k_L k_x \phi_x \dot{\sigma}_x(s_w) - k_L k_y \phi_y \dot{\sigma}_y(s_w)$	$\hat{\boldsymbol{\chi}}^P = [\chi_1, \chi_2]^T / \sqrt{\chi_1^2+\chi_2^2}$	J is the Jacobian matrix

its dynamics avoid the desired path $\mathscr{P}$ to self-intersect in any point, ensures that our ASV does not stall at the control points, and guarantees by construction that the extended guiding vector field $\boldsymbol{\chi} \neq \boldsymbol{0}, \forall \mathbb{R}^3$.

After defining the desired path $\mathscr{P}$ and constructing the vector field $\boldsymbol{\chi}$ (according to the expressions in the second column of Table 2) that guides our ASV, we can compute the control signal u_θ that makes the ASV converge to $\mathscr{P}$. For this purpose, we could define the orientation error $\boldsymbol{e}_{ori}$ as the difference between the ASV's generalized speed $\dot{\boldsymbol{\zeta}}$ and the guiding vector field orientations $\hat{\boldsymbol{\chi}} = \boldsymbol{\chi} / \sqrt{\chi_1^2 + \chi_2^2}$. As the dynamical model of s_w makes $\dot{s}_w - \chi_3 / \sqrt{\chi_1^2 + \chi_2^2} = 0$, our problem can be transformed from $\mathbb{R}^3$ to $\mathbb{R}^2$, by redefining $\boldsymbol{e}_{ori}$ as the actual physical error between the ASV's orientation $\boldsymbol{h}_\theta = [cos(s_\theta), sin(s_\theta)]^T$ and the physical guiding vector field $\hat{\boldsymbol{\chi}}^P$. Finding a Lyapunov function V dependent on this error and a control signal u_θ that makes $\dot{V}$ negative-semidefinite will make $\boldsymbol{e}_{ori} \to 0$ as time evolves. Selecting $V = \frac{1}{2} ||\boldsymbol{e}_{ori}||^2$, whose time derivative is $\dot{V} = (\dot{s}_\theta - \dot{\theta}_d) \boldsymbol{h}^T E \hat{\boldsymbol{\chi}}^P$, makes $\dot{V} = -k_\theta (\boldsymbol{h}^T E \hat{\boldsymbol{\chi}}^P)^2 \leq 0$ when $k_\theta \in \mathbb{R}^+$ and the u_θ of last column of Table 2 is chosen. Besides, if $\boldsymbol{h}_\theta$ and $\hat{\boldsymbol{\chi}}^P$ are aligned, $\dot{V} = 0$, keeping the ASV on $\mathscr{P}$.

4.3 Law for the Ground Velocity Control Signal

Finally, we assume that the speed u_v is fixed by an external controller and equal to that of the spline evaluated at s_w since this choice will make the G&C behave as follows. On the one hand, when the ASV is on track, $\dot{s}_w \sim 1$ to measure the concentration at the time instants specified by the planner. This will ensure that the elapsed time between two control points of the spline is sufficiently close to the difference of the time breaks $t_i - t_{i-1}$ of the spline evaluated $f_i(s_w)$. On the other one, if the ASV is out of track, $\dot{s}_w \sim 0$ to evaluate the spline at an almost constant s_w, making the ASV on track again, since $\boldsymbol{\chi}$ will guide it to the constant point $f_i(s_w)$.

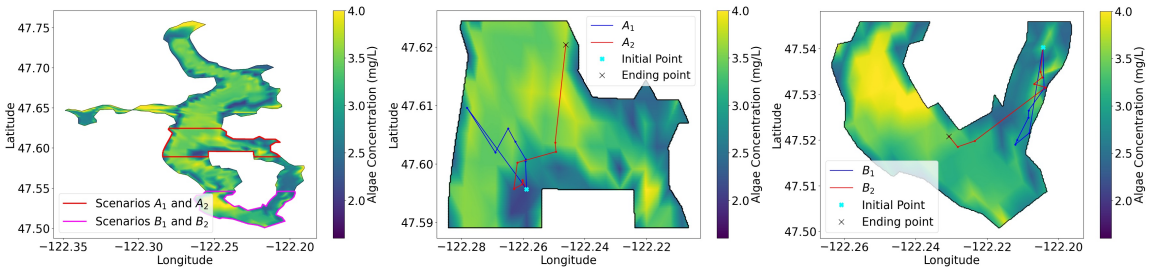
For this purpose, four constants were introduced in the angular velocity control law presented in the previous section. Within the vector field computation, $k_x \in \mathbb{R}^+$ and $k_y \in \mathbb{R}^+$ tune how aggressive the vector field is in the s_x, s_y and s_w directions. In contrast, $k_L \in (0, 1]$ scales down the vector field and makes s_w less sensitive to the remaining parametrization. That is, if k_L is small, each vector field component becomes smaller, s_w changes slower, and u_θ is lower and smoother. The constant $k_\theta \in \mathbb{R}^+$ used in u_θ, determines how fast the ASV converges to $\boldsymbol{\chi}$, since $\dot{V} = -k_\theta (\boldsymbol{h}^T E \hat{\boldsymbol{\chi}}^P)^2$. Thus, as big values of k_θ make $\dot{V}$ more negative, $\boldsymbol{e}_{ori} \to 0$ faster.

5 SIMULATIONS

This section analyzes via simulation the performance of the different parts of our proposal after presenting the main properties of the scenarios under study. It is also worth noting that all the software is programmed in Python, using the open-source framework pymoo (Blank and Deb 2020) for the evolutionary optimizer.

5.1 Scenarios Under Study

The water body under study is Lake Washington, since EEMS provides a pre-tuned ready-to-use example that includes the initial state and contour conditions for the hydrodynamics, temperature, wind, solar



(a) Algae concentration (b) Trajectories for A1 and A2 (c) Trajectories for B1 and B2

Figure 2: Scenario setup and selected optimized trajectory by the operator.

radiation, and water quality modules related to the transport and the concentration changes of inorganic compounds (e.g., oxygen, phosphorous, nitrogen, and ammonia) and algae. Although Figure 2a shows the algae concentration (using greener/bluer colors for higher/lower values) at the fixed probe depth d_{fixed}^{probe} and at t=1.5 hours, the algae concentration at the probe measuring time and depth will be used by the planner.

To set up four different scenarios, we select two particular regions of the lake and provide an initial trajectory location $\boldsymbol{\sigma}(t_0)$ and two final locations $\boldsymbol{\sigma}(t_{N_f})$ within each region. In particular:

Scenarios A1 and A2 are carried in the middle of Lake Washington, framed in red in Figure 2a, where algae concentration is high in several zones. As the ASV starts in a low-concentration region, these scenarios are set up to see if the planner can obtain trajectories that identify areas of high algae concentration. Besides, *A1* differs from *A2* only in the final location of the trajectory, since in *A1*, the ASV must start and end in the same place, while in *A2* the last location is different from the initial. Hence, in *A1*, the ASV must perform a closed trajectory, while in *A2*, it has to follow an open one.

Scenarios B1 and B2 are carried at the bottom of the lake, framed in violet in Figure 2a, to see how the planner adapts the trajectories to the curvature of the lake and this tighter area. Again, *B1* and *B2* differ in the final location of the trajectory, forcing the ASV to perform a close trajectory in *B1* and an open in *B2*.

5.2 Performance Analysis of the Trajectory Planner

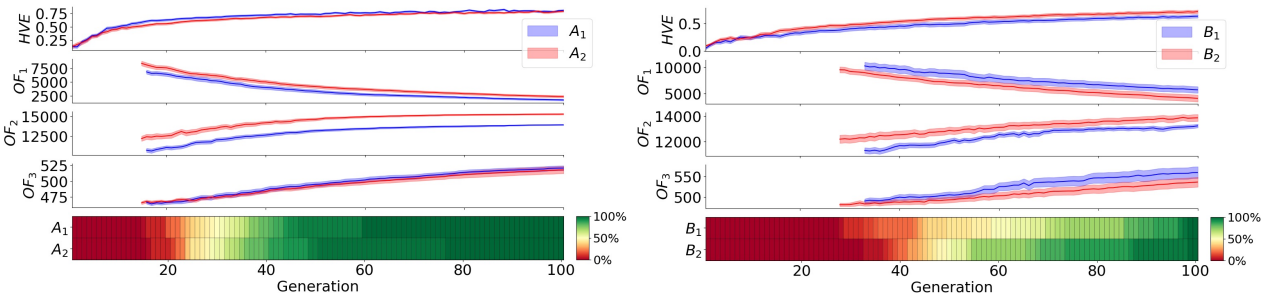
The optimizer of the planner, supported by NSGA-II, is expected to return a different set of solutions whenever it is run. Hence, to analyze its behavior statistically, we run the planner, parameterized as Table 3 summarizes, 40 times for each scenario and store, the CF and OF values of all the solutions that belong to the best Pareto front of each of the 100 generations. From those values, we can first calculate the HyperVolume Estimation (HVE, Bader et al. 2010) for each run and generation and the best OF_i in the best front, and afterward their mean and standard deviation within each generation. The top 4 subplots of Figure 3 represent the evolution of those means and of twice the standard deviations for each scenario (using blue for the closed-loop ones and red for the open-loop ones), while the bottom color plots represent the number of runs where the solutions of the first Pareto front fulfill the constraints.

Observing all those graphics, we can conclude the following. In Scenario *A1* and *A2*, the planner often finds feasible solutions reasonably quickly (after generation 30) and improves them during the remaining generations, where the trajectory length (OF_1) decreases, and the algae concentration accumulated along the ASV trajectory (OF_2) and at the probe profiles (OF_3) increase, as expected. Moreover, the longer the trajectory (which corresponds to A_2), the higher the algae concentration measured along it. However, feasible solutions are usually found later for scenarios *B1* and *B2* since the planner must build a trajectory that fits within a tighter area. Besides, while OF_1 decreases, OF_2 and OF_3 increase with the generation number as expected, and in this case, the trajectory is longer for *B1* than for *B2*, due to the tighter region.

Besides, from the final set of non-dominated solutions returned NSGA-II, the operator systematically selects the one with the best OF_3 value, which is represented in Figures 2b and 2c, using a blue line for the closed trajectories of scenarios *A1* and *B1*, a red line for the open trajectories of scenarios *A2* and *B2*,

Table 3: Trajectory planner parameters.

Encoding						Evaluation		NSGA-II					
N_v	T_{UAV} (s)	N_{probe}	T_{probe} (s)	d_{fixed}^{probe} (m)	Δ_{probe} (m)	$T_{mission}$ (hour)	v_{max} (m/s)	N_p	p_{cross}	η_{cross}	p_{mut}	η_{mut}	N_{stop}
6	0.5	36	1	2	0.5	3.5	1.4	50	0.9	15	0.9	20	100



(a) Scenarios A1 and A2

(b) Scenarios B1 and B2

Figure 3: Evolution of the HVE and objective functions.

a cyan cross for $\boldsymbol{\sigma}(t_0)$, a black cross for $\boldsymbol{\sigma}(t_{N_f})$, and dots of the color used for the trajectory for its splines control points (where the probe profiles are performed).

Finally, it is worth noting that our planner needs 22.66 minutes to optimize the ASV trajectory during 100 generations in a Dual-Core 2.3 GHz Intel Core i5 with 8 GB RAM PC with macOS Monterey. Note also that some extra time can be required to make OF_i curves reach a converging plateau.

5.3 Performance Analysis of the G&C

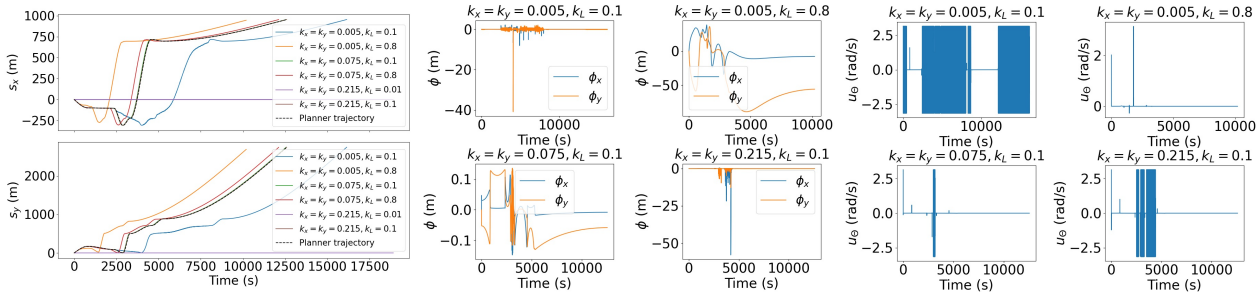
The following simulations allow to determine a suitable G&C parameterization from the combinations of all the values presented in Table 4, where k_θ is fixed since the G&C is more sensitive to the other parameters.

The results of those simulations, for the trajectory selected by the operator for scenarios *A2* and some representative parameterizations of the table, are presented in Figure 4. In particular, Figure 4a represents the time evolution of x and y components of the planner trajectory $[\sigma_x, \sigma_y]^T$ with a black dashed line and the trajectories followed by the ASV $[s_x, s_y]$ with different colored solid lines for the selected parameterizations. Figure 4b and 4c represent respectively, the evolution of the distance errors $\{\phi_x, \phi_y\}$ and of the control signal u_θ. Note that the limits of the ordinate axis change and that the control signal is bounded to $u_\theta \in \{-\pi, \pi\}$ for a better simulation of a real-world scenario. The observation of the figures allows us to conclude the following. First, regardless of the values of k_x and k_y, the low value of $k_L = 0.01$ scales down too much the vector field, making $\dot{s}_w \sim 0$ and stopping the ASV at the beginning of the trajectory (as the representative trajectory with $k_L = 0.01$ of Figure 4a shows). Second, the high value of $k_L = 0.8$ makes the ASV turn too soon before reaching the control splines (as the big ϕ values, associated to control points, of the case of Figure 4b with $k_L = 0.8$ shows). Third, if $k_L = 0.1$ and k_x k_y are low, the trajectory and control signal u_θ have an oscillatory behavior (observable in the cases of Figure 4b and 4c with $k_L = 0.1$ and $k_x = k_y = 0.005$). Forth, when $k_L = 0.1$ and $k_x = k_y = 0.075$, the vector field guides the ASV to the desired trajectory. Finally, if k_x and k_y are further increased, the control signal oscillates again. Figure 5 shows a similar overall behavior of all the graphics for the trajectory selected for scenario *B1*, despite its higher complexity (due to the multiple intersections of the spline).

All these simulations allow us to set our G&C parameters to $k_x = k_y = 0.075$ and $k_L = 0.1$ since 1) ϕ_x and ϕ_y are small for all t, 2) the control signal is smooth enough, and 3) the ASV reaches the control points with the smallest error possible both in time and space.

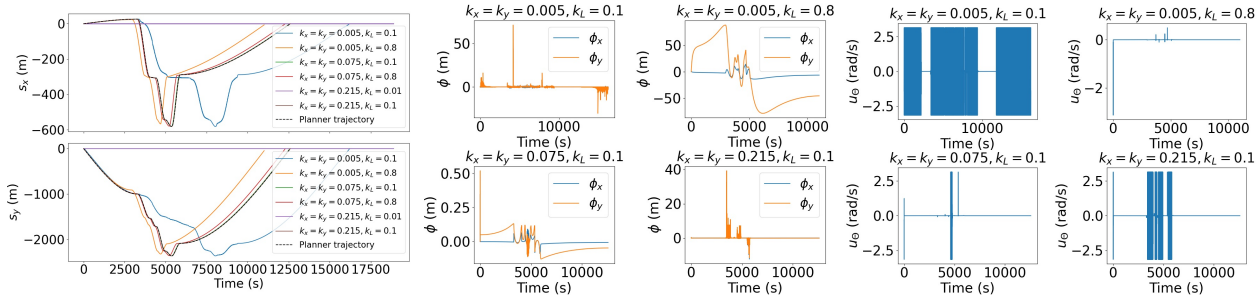
Table 4: Controller parameters.

	Vector Field Aggressiveness		Vector Field Scaling	Control Signal Constant
Parameter	k_x	k_y	k_L	k_θ
Testing range	$[0.005 : 0.070 : 0.285]$	$[0.005 : 0.070 : 0.285]$	$[0.01, 0.1, 0.8]$	Fixed
Selected value	0.075	0.075	0.1	4



(a) Trajectories (b) Distance error (c) Control signal

Figure 4: Controller testing for scenario *A2*.



(a) Trajectories (b) Distance error (c) Control signal

Figure 5: Controller testing for scenario *B1*.

6 CONCLUSIONS

This paper presents a new trajectory planner and a new guidance and control system for self-driving an ASV performing HACBs monitoring missions in lentic waters. The planner optimise the ASV's horizontal and probe vertical trajectories and the G&C makes the ASV move in accordance to the planned trajectory. To this end, on the one hand, the planner optimizes the control nodes of a 2D spline with NSGA-II (a proven multi-objective method), taking into account the simulation of the evolution of the algae o cyanobacteria concentration. On the other one, the G&C system combines the definition of an appropriated vector field for the splines with a Lyapunov-inspired non-linear controller for the ASV orientation. Finally, the multiple optimizations of the planner and simulations of the G&C show that the new system can determine overall good trajectories followable by the ASV, providing that the water body and HACB model is well-tuned.

Some future lines of research can be the following. Improving the planner, e.g. including new constraints related to the ASV maneuverability or new decision variables to determine the maximal depth of the probe at each profile. Adapting the G&C to more complex dynamical models of the ASV (e.g. for a catamaran with two propellers). Defining the dynamic behavior of the probe before developing a controller to make the vertical profiles with it.

ACKNOWLEDGMENTS

This work has been supported by the Research Project IA-GES-BLOOM-CM (Y2020/TCS-6420) of the Synergic program of the Comunidad Autónoma de Madrid, by SMART-BLOOMS (TED2021-130123B-I00) funded by MCIN/AEI/10.13039/501100011033 and the European Union NextGenerationEU/PRTR, and by INSERTION (PID20211-27648OB-C33) funded by the Knowledge Generation Program of the Science and Innovation Ministry of Spain. Gonzalez-Calvin's work is specifically supported by the Collaboration Scholarship Program of the Ministry of Education and Vocational Training of Spain.

REFERENCES

Delft3D. "Delft3D Wepbage". https://oss.deltares.nl/web/delft3d. Accessed 5th Abril 2023.

EFDC-EPA. "Environmental Fluid Dynamics Code Wepbage". https://www.epa.gov/ceam/environmental-fluid-dynamics-code-efdc. Accessed 5th Abril 2023.

Arzamendia, M., D. Gregor, D. G. Reina, and S. L. Toral. 2019. "An Evolutionary Approach to Constrained Path Planning of an Autonomous Surface Vehicle for Maximizing the Covered Area of Ypacarai Lake". *Soft Computing* 23:1723–1734.

Arzamendia, M., D. Gregor, D. G. Reina, S. L. Toral, and R. Gregor. 2016. "Evolutionary Path Planning of an Autonomous Surface Vehicle for Water Quality Monitoring". In *Int. Conf. on Developments in e-Systems Engineering*. Aug. 31st- Sept. 2nd, Liverpool, UK.

ATK, LAKE2K and QUAL2K. "ATK, LAKE2K and QUAL2K Wepbage". https://www.qual2k.com/index.html. Accessed 5th Abril 2023.

Bader, J., K. Deb, and E. Zitzler. 2010, 01. *Faster Hypervolume-Based Search Using Monte Carlo Sampling*, Volume 634, 313–326.

Besada-Portas, E., J. M. Girón-Sierra, J. Jiménez, and J. A. López-Orozco. 2021. "Data-Driven Exploration of Lentic Water Bodies with ASVs Guided by Gradient-Free Optimization/Contour Detection Algorithms". In *Winter Simulation Conf. (WSC)*, 1–12. Phoenix, AZ, USA: Institute of Electrical and Electronics Engineers, Inc.

Blank, J., and K. Deb. 2020. "Pymoo: Multi-Objective Optimization in Python". *IEEE Access* 8:89497–89509.

Carazo-Barbero, G., E. Besada-Portas, J. M. Girón-Sierra, and J. A. López-Orozco. 2021. "EA-based ASV Trajectory Planner for Pollution Detection in Lentic Waters". In *Applications of Evolutionary Computation*, edited by P. A. Castillo and J. L. Jiménez Laredo, 812–827. Cham: Springer International Publishing.

CE-QUAL-W2. "Hydrodynamcis and Water Quality Model Wepbage". http://www.ce.pdx.edu/w2/. Accessed 5th Abril 2023.

de la Cruz, J. M., J. A. Lopez-Orozco, E. Besada-Portas, and J. Aranda-Almansa. 2015. "A Streamlined Nonlinear Path Following Kinematic Controller". In *IEEE Int. Conf. on Robotics and Automation (ICRA)*, 6394–6401.

de Marina, H. G., J. Jiménez, and W. Yao. 2021. "Leaderless Collective Motions in Affine Formation Control". In *IEEE Conf. on Decision and Control (CDC)*, 6433–6438.

de Marina, H. G., Y. A. Kapitanyuk, M. Bronz, G. Hattenberger, and M. Cao. 2017. "Guidance Algorithm for Smooth Trajectory Tracking of a Fixed Wing UAV Flying in Wind Flows". In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, 5740–5745. IEEE.

Deb, K., and R. B. Agrawal. 1995. "Simulated Binary Crossover for Continuous Search Space". *Complex Systems* 9:115–148.

Deb, K., and R. B. Agrawal. 1999. "A Niched-Penalty Approach for Constraint Handling in Genetic Algorithms". In *Inf. Conf. on Artificial Neural Networks and Genetic Algorithms (ICANNGA)*, 235–243.

Deb, K., A. Pratap, S. Agarwal, and T. Meyarivan. 2002. "A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II". *IEEE Transactions on Evolutionary Computation* 6(2).

DYRESM-CAEDYM. "DYRESM-CAEDYM Wepbage". https://teamwork.niwa.co.nz/display/IFM/DYRESM-+CAEDYM. Accessed 5th Abril 2023.

ECOMSED. "Water Quality and Hydrodynamic Modeling Software Wepbage". https://www.hdrinc.com/services/environmental-sciences/water-quality-hydrodynamic-modeling-software. Accessed 5th Abril 2023.

EEMS. "EE Modeling System Wepbage". https://www.eemodelingsystem.com. Accessed 5th Abril 2023.

European Commission 2021. "European Commission Drinking Water Directive". https://www.europeandrinkingwater.eu/initiative/dwd-2021/. Accessed 5th Abril 2023.

Fan, Y., B. Qiu, L. Liu, and Y. Yang. 2023. "Global Fixed-Time Trajectory Tracking Control of Underactuated USV Based on Fixed-Time Extended State Observer". *ISA Transactions* 132:267–277.

Girón-Sierra, J. M., and J. Chacón-Sombría. 2021. "Application of Teams of USVs for Cyanobacteria Monitoring: Initial Steps". In *IFAC Conf. on Control Applications in Marine Systems, Robotics, and Vehicles (CAMS)*, 812–827.

GLM. "The General Lake Model Wepbage". https://aed.see.uwa.edu.au/research/models/glm/. Accessed 5th Abril 2023.

Hitz, G., F. Pomerleau, M. Ève Garneau, C. Pradalier, T. Posch, J. Pernthaler, and R. Siegwart. 2012. "Design and Application of a Surface Vessel for Autonomous Inland Water Monitoring". *IEEE Robotics & Automation Magazine* 19:62–72.

Kapitanyuk, Y. A., A. V. Proskurnikov, and M. Cao. 2017. "A Guiding Vector-Field Algorithm for Path-Following Control of Nonholonomic Mobile Robots". *IEEE Transactions on Control Systems Technology* 26(4):1372–1385.

Kathen, M. J. T., I. J. Flores, and D. G. Reina. 2021. "An Informative Path Planner for a Swarm of ASVs Based on an Enhanced PSO with Gaussian Surrogate Model Components Intended for Water Monitoring Applications". *Electronics* 10(13).

Liu, Z., L. Yu, Q. Xiang, T. Qian, Z. Lou, and W. Xue. 2021. "Research on USV Trajectory Tracking Method Based on LOS Algorithm". In *2021 14th International Symposium on Computational Intelligence and Design (ISCID)*, 408–411.

Liua, Z., Y. Zhanga, X. Yua, and C. Yuana. 2016. "USVs: An Overview of Developments and Challenges". *Annual Reviews in Control* 41.

Meriluoto, J., L. Spoof, and G. Codd. 2017. *Handbook of Cyanobacterial Monitoring and Cyanotoxin Analisis*. Wiley.

N. Karapetyan, J. M., and I. Rekleitis. 2019. "Meander-Based River Coverage by an Autonomous Surface Vehicle". In *Int. Conf. on Field and Service Robotics*.

Peralta, F., D. G. Reina, and S. Toral. 2023. "Water Quality Online Modeling Using Multi-Objective and Multi-Agent Bayesian Optimization with Region Partitioning". *Mechatronics* 91.

Ravanka, A., A. A. Ravankar, Y. Kobayashi, Y. Hoshino, and C.-C. Peng. 2018. "Path Smoothing Techniques in Robot Navigation: State-of-the-Art, Current and Future Challenges". *Sensors* 18(9).

Shumway, S. E., J. M. Burkholder, and S. L. Morton. 2018. *Harmful Algal Blooms: A Compendium Desk Reference*. Wiley.

Siyang, S., and T. Kerdcharoen. 2016. "Development of Unmanned Surface Vehicle for Smart Water Quality Inspector". In *Int. Conf. on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology*.

United Nations 2023. "2030 Agenda for Sustainable Development". https://sdgs.un.org/goals. Accessed 5th Abril 2023.

United Nations Environment Programme 2023. "GOAL 6: Clean Water and Sanitation". https://www.unep.org/explore-topics/sustainable-development-goals/why-do-sustainable-development-goals-matter/goal-6. Accessed 5th Abril 2023.

United States Environmental Protection Agency 2021. "Drinking Water Requirements for States and Public Water Systems". https://www.epa.gov/dwreginfo/drinking-water-regulations. Accessed 5th Abril 2023.

UNTRIM. "UNTRIM Wepbage". https://wiki.baw.de/en/index.php/Mathematical_Model_UNTRIM. Accessed 5th Abril 2023.

Valada, A., P. Velagapudi, B. Kannan, C. Tomaszewski, G. Kantor, and P. Scerri. 2012. "Development of a Low Cost Multi-Robot Autonomous Marine Surface Platform". In *Int. Confl on Field and Service Robotics*.

Wang, H., Z. L. Jiao Dong, L. Yan, and S. Wang. 2022. "Control Algorithm for Trajectory Tracking of an Underactuated USV under Multiple Constraints". *Mathematical Problems in Engineering* 2022.

WASP. "Water Quality Analysis Simulation Program Wepbage". https://www.epa.gov/ceam/water-quality-analysis-simulation-program-wasp. Accessed 5th Abril 2023.

Xiong, C., D. Chen, D. Lu, Z. Zeng, and L. Lian. 2019. "Path Planning of Multiple Autonomous Marine Vehicles for Adaptive Sampling using Voronoi-Based ant Colony Optimization". *Robotics and Autonomous Systems* 115:90 – 103.

Xiong, C., H. Zhou, D. Lu, Z. Zeng, L. Lian, and C. Yu. 2020. "Rapidly-Exploring Adaptive Sampling Tree*: A Sample-Based Path-Planning Algorithm for Unmanned Marine Vehicles Information Gathering in Variable Ocean Environments". *Sensors* 20(9):18 pages.

Yao, W., H. G. de Marina, B. Lin, and M. Cao. 2021. "Singularity-free Guiding Vector Field for Robot Navigation". *IEEE Transactions on Robotics* 37(4):1206–1221.

AUTHOR BIOGRAPHIES

ALFREDO GONZALEZ-CALVIN is a Master Student of Systems Engineering and Control at the University Complutense of Madrid (UCM). He already holds an undergraduate degree on Electrical Engineering from the same university. His research interests are non-linear systems, control and optimization. His email address is alfredgo@ucm.es.

LIA GARCIA-PEREZ is a Doctor Assistant Professor of Systems Engineering and Automation at UCM. She holds a PhD in Computer Architecture and Automation from UCM. Her research interests include mobile robot navigation, control for autonomous robots and robotics education. Her email address is liagar05@ucm.es.

JOSÉ L. RISCO-MARTÍN received his Ph.D. from UCM, where he is currently Full Professor in the Department of Computer Architecture and Automation. His research interests include systems modeling, simulation, and optimization. He can be reached at jlrisco@ucm.es.

EVA BESADA-PORTAS is an Associate Professor of Systems Engineering and Automation at UCM. She also holds a PhD in Computer Systems from UCM. Her research interests include uncertainty modeling and simulation, heuristic optimization, optimal control and planning of unmanned vehicles. Her email address is ebesada@ucm.es.