ResiPlan: Closing the Planning-Acting Loop for Safe Underwater Navigation

Marios Xanthidis, Eleni Kelasidi, and Kostas Alexis

***Abstract*— Autonomous operation in underwater environments is, arguably, one of the most complex domains. It requires safe operations under the presence of unpredictable surge, currents, uncertainty, and dynamic obstacles that challenges to the highest degree real-time motion planning; the primary focus of this paper. Although previous work addressed the problem of safe real-time 3D navigation in cluttered underwater environments, it did not account explicitly for disturbances, currents, dynamic obstacles, or uncertainty growth. This paper presents ResiPlan, a novel motion planning framework that utilizes past information of errors monitoring the path follower's performance, along with estimation of dynamic obstacles and uncertainty, to produce adaptive paths by adjusting the safety margins accordingly. Extensive numerical experiments and simulations validate the safety guarantees of the technique, in a variety of different environments with various types of disturbance, showcasing the strong potential to be utilized for operations in challenging underwater environments.**

I. Introduction

Autonomous underwater navigation is a very challenging fundamental problem in underwater robotics; a domain that becomes increasingly relevant both academically and commercially. Several challenges need to be addressed in order to enable robust and resilient autonomous navigation of Autonomous Underwater Vehicles (AUVs) in challenging scenarios such as aquaculture, fisheries, oil and gas, cave exploration. Such environments involve state and map uncertainty due to poor sensing conditions, motion uncertainty due to surge and currents, presence of dynamic obstacles, and the necessity to generate real-time solutions. Our goal with this paper is to provide the fundamentals for a robust real-time 3D navigation framework that can resiliently operate safely in these challenging conditions.

Real-time underwater navigation remains a hard, yet to be addressed sufficiently problem as shown by recent reviews of the domain [1]–[3]. Studies have provided solutions for real-time navigation [4]–[9], though, only for robots moving in 2D or with assumptions of vertical relief [10].

For operations in dynamic environments, fundamental work has been performed to produce safe trajectories [11]–[13], and especially in the maritime domain [14]–[16], but are limited to 2D. Safe navigation in dynamic 3D environments has been performed [17]–[24], but without considering uncertainty or motion errors due to currents. Furthermore, several methods [25]–[33] enable operations with disturbances and currents in the maritime domain, but are limited to 2D. Efficient 3D frameworks have been introduced for planning with currents and uncertainty [34]–[43], however, not for dynamic obstacles.

To the best of our knowledge, none of the methodologies mentioned above could guarantee safe operations in the intersection of uncertainty, motion errors, and dynamic obstacles, which is the research gap our proposed method attempts to address. The proposed technique called ResiPlan, utilizes AquaNav [39], our past work. AquaNav's core is an efficient modified path optimization planner based on Trajopt [44], along with a warm-restarting module to handle local minima issues, and a path follower with a low level controller. Inheriting many positive properties of Trajopt, it is a computationally efficient and real-time 3D navigation framework with strong safety guarantees in unstructured cluttered environments, but with no safety guarantees in environments with dynamic obstacles, uncertainty, or currents.

Similarly to AquaNav, ResiPlan exploits the computational efficiency of path optimization, and extends the safety guarantees to environments and systems prone to the aforementioned conditions, by computing an appropriate clearance to avoid any collisions. The novel method provides solutions accommodating the imperfections of the path follower in a closed loop way, by monitoring its performance due to disturbances or currents in the environment, detected dynamic obstacles, and uncertainty growth. The proposed framework remains agnostic when it comes to the source or the nature of the motion errors to promote generality, mitigating errors caused by the path follower or the environment in a unified way. In summary, the contributions of this work are:

1) A computationally efficient 3D real-time navigation framework that provides safe solutions in environments with unknown disturbances, detected dynamic obstacles of arbitrary shape, and bounded uncertainty growth.
2) Proof on correctness and safety guarantees, assuming successful convergence.
3) Extensive numerical experiments and simulations with the AquaNav baseline, validating the contributions for different maps and conditions.

II. Problem Statement

The goal of this work is to produce safe trajectories for underwater robots operating in open-water settings, with unknown underwater conditions (such as currents or surge),

Marios Xanthidis and Eleni Kelasidi are with the Department of Aquaculture, SINTEF Ocean AS [marios.xanthidis, eleni.kelasidi]@sintef.no, and Kostas Alexis is with the Department of Engineering Cybernetics at NTNU konstantinos.alexis@ntnu.no

This work was made possible through the support of the Research Council of Norway (ResiFarm, NO-327292).

inaccurate controls, and dynamic environments. For simplicity, we assume a holonomic AUV that utilizes an arbitrary path follower. The guidance methodology employed by the path follower is beyond the scope of this research, but extensions to adapt different guidance policies are possible, since no explicit limitations are introduced by the proposed methodology. A typical 4D configuration space $\mathcal{C}$ will be utilized in the next section which includes the 3 axes of translation and horizontal heading ψ respectively, $\mathcal{C} = X \times Y \times Z \times \Psi$.

More formally, let $s_0 = [x_0, y_0, z_0, \psi_0]$ indicate the initial state of the robot, G the goal region defined as a small closed set around the final state $\lceil s_n \rceil = [x_n, y_n, z_n]$, where $\lceil s \rceil$ is the translational component of a state $s \in \mathcal{C}$). Additionally, let $S = \{s_1, s_2, \ldots, s_n\}$ denote a path to be followed by the AUV from the initial s_1 to a goal s_n ($s_n \in G$); $n \in \mathbb{N}$ and $n \geq 2$, so that for different values of n the trajectory could be approximated to an arbitrary resolution. Furthermore, $\widetilde{S} = \{\widetilde{s}_{t_1}, \widetilde{s}_{t_2}, \ldots, \widetilde{s}_{t_n}\}$, with $\widetilde{s}_{t_n} \in G$, describes the true trajectory executed by the robot represented at corresponding time intervals $T = [t_1, t_2, \ldots, t_n]$, prone to disturbances and motion errors when the path follower processed S. Additionally, let $\widehat{S} = \{\widehat{s}_1, \widehat{s}_{t_2}, \ldots, \widehat{s}_{t_n}\}$ represent the estimated trajectory prone to errors bounded by uncertainty $P(t_i)$, so that $\|\widehat{s}_{t_i} - \widetilde{s}_{t_i}\| \leq P(t_i)$ for $t_i \in T$. Finally, let $O = \{o_0, o_1, \ldots o_n\}$ denote a set of known static and dynamic convex obstacles in the environment, $B^t_{o_j}$ the 3D shape occupied by the obstacle $o_j \in O$ at time $t_i \in T$, and B_s the 3D space occupied by the robot at an arbitrary state $s \in \mathcal{C}$. Then our objective is summarized as:

$$f(S) = \min_{S} \sum_{i=1}^{n-1} \|s_{i+1} - s_i\|, \tag{1}$$

$$\text{s.t.} \bigcup_{\substack{t_i \in T \\ \widetilde{s}_{t_i} \in \widetilde{S}}} (B_{\widetilde{s}_{t_i}} \cap \bigcup_{o_j \in O} B^{t_i}_{o_j}) = \varnothing$$

Practically, the goal is to produce locally optimal paths of minimal length that avoid any collisions of the robot with any obstacles in the environment during the entire execution of the query.

III. Proposed Approach

The novel framework introduced in this paper is taking in to account errors experienced during the robots motion, metrics of the replanning process, and knowledge of the environment. Figure 1 shows how the novel enhancements adapts to a typical navigation framework. A main limitation of AquaNav is that it requires tuning by a human operator to determine an appropriate safe clearance for specific conditions, through extensive experimentation [39]. ResiPlan extends autonomy capabilities by providing a system that utilizes information from the state estimator and the path follower's performance. The method is adaptive to the system itself and the environment considering uncertainty growth, imperfect path following, and external conditions, such as currents or dynamic obstacles.

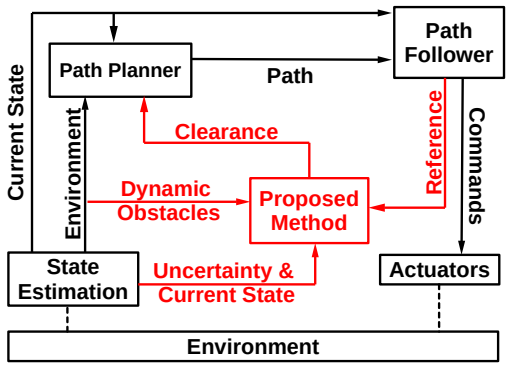

Fig. 1. The proposed adaptive framework, shown in red. It computes appropriate clearance informing the optimization process, by utilizing localization, positions of dynamic obstacles, and uncertainty estimation from state estimation, along with the reference state of the path follower in a closed-loop way.

The remainder of the section starts with a short description of the core planner used, and continues with the computation of clearance for zero-mean errors and a proof regarding its limitations. Then the computation guaranteeing safe navigation under generic errors is derived, along with proposed methodologies for expanding the clearance formulation to ensure safety around dynamic obstacles. Finally, the section ends with an extension to the proposed model for safe navigation while incorporating uncertainty growth.

A. Core Path Planner

The core path planner of ResiPlan is Trajopt [44]. Trajopt receives an initial path, typically a linear interpolation between the initial and goal configuration, and then it attempts to optimize it. The optimization minimizes the objective function of Equation 1 with sequential convex optimization, by utilizing a variant of the ℓ_1 penalty method [45] to overcome non-differential convex constraints. If c_i is a desired clearance corresponding to state $s_i \in S$ with $c_i \in C$ being the set of all clearances $C = \{c_1, c_2, \ldots, c_n\}$, continuous time safety with a desired clearance could be guaranteed in an environment of static obstacles O by enforcing the following constraints:

$$H(s_i, s_{i+1}) = \sum_{o_j \in O} \left| c_i - \text{sd}(L^{s_{i+1}}_{s_i}, o_j) \right|, \tag{2}$$

where $\text{sd}()$ represents the signed distance of two 3D convex objects, and $L^{s_{i+1}}_{s_i}$ the convex hull of $B_{s_i} \cup B_{s_{i+1}}$, which corresponds to the volume occupied by the robot during the transition. The signed distance between the convex objects is computed efficiently for non overlapping objects by the Gilbert-Johnson-Keerthi algorithm [46], and for in-collision objects by the Expanding Polytope Algorithm [47]. Although Trajopt operates on convex polytopes, build-in functions performing convex decomposition allow for safe underwater navigation through highly non-convex environments [39].

B. Clearance Calculation for Zero-mean Oscillations

In general, zero-mean errors and minor oscillations are present due to imperfections of hardware, sensing, dynamics analysis, or parameter tuning, although often they could be

neglected during path planning. In the underwater domain, though, such zero-mean errors, due to inaccurate dynamics or surge in complex open water settings, could compromise safe operations and it's essential to be taken into account. Similarly to the feedback controllers, albeit at a higher level, we utilize the error of the current state of the system to the desired corresponding state of the planned path. So, if at time $\tau \in T$ for some value of n, $\widetilde{s}_\tau$ is the true state of the robot, and s_τ the desired reference state of the path received by the path follower, then the path following error is defined as:

$$e_\tau = \|\lceil s_\tau \rceil - \lceil \widetilde{s}_\tau \rceil\| \tag{3}$$

Thus, to guarantee safe operations and avoid collisions with any detected obstacles, the planned path S needs to maintain a clearance $c_\tau^0 \geq e_\tau$ for any $s_\tau \in S$ and $\tau \in T$. Let $T_w = \{t_{w_1}, \ldots, t_{w_n}\} \subseteq T$ be the corresponding times to reach the waypoints $S_w^{t_{w_1}} = \{s_{w_1}, \ldots, s_{w_n}\}$, $S_w \subseteq S$, where t_{w_1} indicates the time the specific planning process initiated. If the velocity of the robot $\|u_r\|$ is constant, the time instances in T_w can be computed recursively:

$$t_{w_{i+1}} = t_{w_i} + \frac{\left\|\lceil s_{w_{i+1}} \rceil - \lceil s_{w_i} \rceil\right\|}{|u_r|}. \tag{4}$$

For every $\tau \in T$ and for two consecutive time instance $t_{w_i}, t_{w_{i+1}} \in T_w$ so that $t_{w_i} < \tau \leq t_{w_{i+1}}$, the desired state $\lceil s_\tau \rceil$ is calculated as:

$$\lceil s_\tau \rceil = \lceil s_{w_i} \rceil + \frac{\tau - t_{w_i}}{t_{w_{i+1}} - t_{w_i}} (\lceil s_{w_{i+1}} \rceil - \lceil s_{w_i} \rceil). \tag{5}$$

With the assumption that potentially future critical disturbances were present also in past experience, if $E_\tau^{t_o}$ is the set of past path following errors from an arbitrary past time t_o to τ, $t_o < \tau$, safe performance can be guaranteed with a worst case reasoning policy:

$$c_\tau^0 = \max(E_\tau^{t_o}), \tag{6}$$

where t_o could be chosen to keep errors for an arbitrary period of time. Finally, to account for potential unexpected larger errors, we propose utilizing a sliding window policy of period t_w, and calculating the desired clearance as:

$$c_\tau^0 = \alpha \ \max(E_\tau^{\tau - t_w}) + \epsilon \tag{7}$$

where the factors $\alpha \geq 1$ and $\epsilon \geq 0$ are chosen to offer an additional safety margin mitigating future errors that might surpass $\max(E_\tau^{t_o})$. Equation 3 utilizes ground truth to uncover the exact navigation error. In practice, the estimated state $\widehat{s}_\tau$ will be used, which is also prone to error. If the different type of zero-mean errors are independent, statistically, locally maximum outliers providing the additive combination of the errors exist. Thus, ResiPlan can provide safe performance, by processing the resulting error holistically.

C. Improved Safety for Persistent or Generic Disturbances

This section provides the methodology to guarantee safety while operating also under persistent disturbances, a common error occurrence in underwater environments due to the presence of currents. First, we provide a proof that the clearance calculated in the previous section cannot handle persistent errors, following with the proposed formulation to deal with such persistent disturbances by construction.

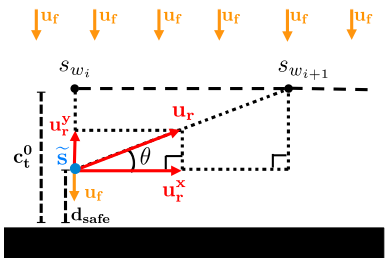

Fig. 2. An instance of a robot at state $\widetilde{s}$ (blue), moving with linear velocity to the next waypoint (red), attempting to follow the planned dashed path under constant current (orange), used to prove Theorem 3.1.

Theorem 3.1: Equation 7 cannot guarantee safety if the disturbances are not zero-mean.

Proof: Lets assume for simplicity, without loss of generality, a 2D point robot with linear speed u_r following a path at distance c_t^0, $\forall t \in T$, from an arbitrary long rectangular obstacle, and a constant error towards the obstacle at velocity u_f, as shown in Figure 2. Since the motion error drifts the robot towards the obstacle, to guarantee collision free performance, at least one state should exist with clearance $d_{safe} \in (0, c_t^0]$, where:

$$|u_r^y| \geq \|u_f\| \tag{8}$$

if u_r^y is the vertical resultant of the velocity vector u_r, which is in opposite direction of u_f. For an arbitrary state $\widetilde{s}$, the magnitude of u_r^y could be calculated as:

$$|u_r^y| = \|u_r\| \, sin(\widehat{\vec{u}_r \vec{u}_r^x}). \tag{9}$$

Given that $\sin(\widehat{\vec{u}_r \vec{u}_r^x}) = \sin(\theta)$:

$$\sin(\widehat{\vec{u}_r \vec{u}_r^x}) = \frac{c_t^0 - d_{safe}}{\sqrt{(c_t^0 - d_{safe})^2 + \left\|\lceil s_{w_{i+1}} \rceil - \lceil s_{w_i} \rceil\right\|^2}}. \tag{10}$$

Since $\theta \in (0, \frac{\pi}{2})$, and $0 < \sin(\theta) < 1$, and the function $f(x) = \frac{x}{\sqrt{x^2 + a^2}}$ for $x \in [0, \infty)$ is monotonically increasing, then Equation 10 is maximized for $d_{safe} = 0$. So:

$$\max(|u_r^y|) = \|u_r\| \frac{c_t^0}{\sqrt{(c_t^0)^2 + \left\|\lceil s_{w_{i+1}} \rceil - \lceil s_{w_i} \rceil\right\|^2}}. \tag{11}$$

Since the error is constant, assuming a replanning time of Δt, for $\alpha = 1$ and $\epsilon = 0$, $c_t^0 = \|u_f\| \Delta t$. Thus, for $\|u_r\| > \|u_f\|$ it must be true that:

$$\|u_f\| \leq \|u_r\| \frac{\|u_f\| \Delta t}{\sqrt{(\|u_f\| \Delta t)^2 + \left\|\lceil s_{w_{i+1}} \rceil - \lceil s_{w_i} \rceil\right\|^2}} \Rightarrow$$
$$\left\|\lceil s_{w_{i+1}} \rceil - \lceil s_{w_i} \rceil\right\| \leq \sqrt{\|u_r\|^2 \Delta t - (\|u_f\| \Delta t)^2}. \tag{12}$$

Thus, shown by contradiction, for step size $\left\|\lceil s_{w_{i+1}} \rceil - \lceil s_{w_i} \rceil\right\| > \sqrt{\|u_r\|^2 \Delta t - (\|u_f\| \Delta t)^2}$, there are no guarantees of safe performance using solely Equation 7. ■

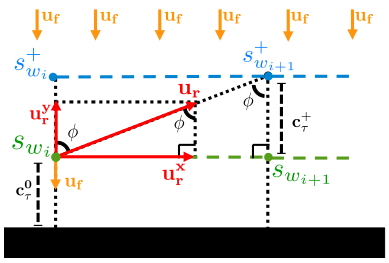

Fig. 3. A segment of the path for two consecutive waypoints, s_{w_i} and $s_{w_{i+1}}$, using the clearance of Equation 7 is shown with green, while the corresponding segment of the target path, $s^+_{w_i}$ and $s^+_{w_{i+1}}$, using an additional safety margin is shown with blue.

We compute an additional safety margin c^+_τ to c^0_τ, by employing worst case reasoning to treat the measured error only as persistent. Considering the example in Figure 3, the extraction of the formula for c^+_τ follows.

If the error of Equation 3 is calculated in the beginning of the replanning process, then let $\Delta T^{t_0}_\tau$ be the set of replanning times corresponding to each measured error in $E^{t_o}_\tau$, and $\Delta t^{\max(E^{t_o}_\tau)} \in \Delta T^{t_0}_\tau$ the corresponding replanning period when $\max(E^{t_o}_\tau)$ was recorded. By utilizing Equation 8:

$$|u^y_r| = \|u_f\| = \frac{\max(E^{t_o}_\tau)}{\Delta t^{\max(E^{t_o}_\tau)}}. \tag{13}$$

Then for the resultant u^y_r:

$$|u^y_r| = \|u_r\| \cos(\widehat{\vec{u}_r \vec{u}^y_r}) \Rightarrow \cos(\widehat{\vec{u}_r \vec{u}^y_r}) = \frac{|u^y_r|}{\|u_r\|}. \tag{14}$$

From the triangle formed by the lines c^+_τ, $\lceil s^+_{w_{i+1}} \rceil - \lceil s_{w_i} \rceil$, and the angle ϕ:

$$\begin{aligned} \cos(\phi) = \cos(\widehat{\vec{u}_r \vec{u}^y_r}) = \frac{c^+_\tau}{\left\| \lceil s^+_{w_{i+1}} \rceil - \lceil s_{w_i} \rceil \right\|} \Rightarrow \\ c^+_\tau = \left\| \lceil s^+_{w_{i+1}} \rceil - \lceil s_{w_i} \rceil \right\| \cos(\widehat{\vec{u}_r \vec{u}^y_r}). \end{aligned} \tag{15}$$

Combining the equations above:

$$\begin{aligned} c^+_\tau = \left\| \lceil s_{w_{i+1}} \rceil - \lceil s_{w_i} \rceil \right\| \cos(\widehat{\vec{u}_r \vec{u}^y_r}) = \\ \left\| \lceil s^+_{w_{i+1}} \rceil - \lceil s_{w_i} \rceil \right\| \frac{\frac{\max(E^{t_o}_\tau)}{\Delta t^{\max(E^{t_o}_\tau)}}}{\|u_r\|} \Rightarrow \\ c^+_\tau = \frac{\max(E^{t_o}_\tau)}{\Delta t^{\max(E^{t_o}_\tau)}} \frac{\left\| \lceil s^+_{w_{i+1}} \rceil - \lceil s_{w_i} \rceil \right\|}{\|u_r\|}. \end{aligned} \tag{16}$$

With the clearance calculated above, the AUV will at worst come in proximity c^0_τ to the obstacles. In such case, it is true for the first state that $\widetilde{s} = s_{w_1} = s^+_{w_1}$, so $\left\| \lceil s^+_{w_2} \rceil - \lceil s_{w_1} \rceil \right\| = \left\| \lceil s^+_{w_2} \rceil - \lceil s^+_{w_1} \rceil \right\|$. Additionally, the paths produced by the optimization process are uniform, so the distance between each consecutive waypoints of path S^τ_w is equal to the average step distance $av(S^\tau_w)$:

$$\left\| \lceil s_{w_{i+1}} \rceil - \lceil s_{w_i} \rceil \right\| = av(S^\tau_w) = \frac{\sum_{i=1}^{n-1} \left\| \lceil s_{w_{i+1}} \rceil - \lceil s_{w_i} \rceil \right\|}{n-1}. \tag{17}$$

Then for $S^{\tau\,+}_w$ the average step as a function of S^τ_w is:

$$av(S^{\tau\,+}_w) = av(S^\tau_w) + \frac{\left\| \lceil s^+_{w_2} \rceil - \lceil s_{w_1} \rceil \right\|}{n-1}. \tag{18}$$

The above requires knowledge of the average step distance of a path that has not been computed yet. We propose using a scalar factor $\beta > 1$ describing an expectation for the minor increment so that $av(S^{\tau\,+}_w) \simeq \beta av(S^\tau_w)$. Also, a parameter $k > 0$ is introduced, indicating the maximum allowed average step distance $av(S^{\tau\,+}_w)$, as an upper bound. In a single instance, this simplification might produce an inaccurate clearance, similarly to Equation 7 given no previous experience. However, the future paths adapt to the environment shrinking their path length difference, and quickly minimizing their average step distance difference. Then, using k the maximum average step is enforced by calculating the desired number of states n for the optimization:

$$n = \left| \frac{\sum_{i=1}^{n-1} \left\| \lceil s_{w_{i+1}} \rceil - \lceil s_{w_i} \rceil \right\|}{k} \right| + 1. \tag{19}$$

Finally, the above simplification combined with Equation 7 and Equation 16, results to the proposed clearance with strong safety guarantees under generic disturbances, for each state $i \in [0, n]$:

$$c_i = c^0_\tau + c^+_\tau = \alpha \ \max E^{\tau - t_w}_\tau (1 + \frac{\beta av(S^\tau_w)}{\|u_r\| \, \Delta t^{\max(E^{t_o}_\tau)}}) + \epsilon. \tag{20}$$

Thus, the clearance needed to guarantee safety depends on the distance between the waypoints, the linear velocity of the robot, the error observed, and the replanning time.

D. Extension to Dynamic Environments

ResiPlan guarantees safe navigation in the presence of currents and dynamic obstacles, with the assumption that the robot can detect all nearby obstacles, and that it can move faster than the combined velocity of both the dynamic obstacles and the currents. If two consecutive positions $\lceil p^{\tau - \Delta t}_{o_j} \rceil$ and $\lceil p^\tau_{o_j} \rceil$ of obstacle $o_j \in O$ are known, with Δt be the replanning period, then the set of the current observed linear velocities $U^\tau_O = \{u^\tau_{o_0}, u^\tau_{o_1}, \dots\}$ can be formed as:

$$u^\tau_{o_j} = \frac{\lceil p^\tau_o \rceil - \lceil p^{\tau - \Delta t}_o \rceil}{\Delta t}. \tag{21}$$

From the perspective of the robot, a dynamic obstacle moving towards the robot with a speed $\vec{u}^\tau_o$ is equivalent to a flow of speed $-\vec{u}^\tau_o$ pushing the robot towards a static obstacle. By computing a proper additional clearance, the robot can guarantee safe performance even though the optimization process takes place in a static map. A conservative additional clearance is taking into account the speed of the fastest moving obstacle in proximity along with the slowest observed replanning time. A factor $\gamma \geq 1$ is added to account for potential acceleration of the obstacles, appending Equation 20 the proposed clearance is computed as:

$$\begin{aligned} c_i = (\alpha \ \max E^{\tau - t_w}_\tau + \gamma \max_{u \in U^i_O} (\|v\|) \max_{t \in \Delta T^{t_0}_\tau} (t)) \\ (1 + \frac{\beta av(S^\tau_w)}{\|u_r\| \, \Delta t^{\max(E^{t_o}_\tau)}}) + \epsilon, \end{aligned} \tag{22}$$

where U^i_O is a set of velocities of obstacles in proximity to waypoint s_{w_i}. A conservative policy could be to set

$U_O^i = U_O^\tau$ and calculate the same clearance for all the states with respect to the fastest moving obstacle. Alternatively, clearance could be minimized for some states if the dynamic obstacles are propagated at the corresponding times given by Equation 4, and considered only within a specified proximity. So for the waypoint s_{w_i} corresponding to time t_{w_i}, and propagated position $p_o^{t_{w_i}}$ for $o_j \in O$, U_O^i could be formed the following way:

$$U_O^i = \{u_{o_j} | o_j \in O \wedge \left\| \lceil s_{w_i} \rceil - \lceil p_{o_j}^{t_{w_i}} \rceil \right\| < c_\tau^0 + c_\tau^+ + \left\| u_{o_j}^\tau \right\| \max_{t \in \Delta T_\tau^{t_0}} (t)\}. \tag{23}$$

The main idea is that we consider obstacles that are at least in a range of the desired clearance to avoid static obstacles, expanded by the maximum potential distance traveled by the obstacles during replanning.

Since the states might have different desired clearances, and the length of the future solution is prone to change, a process is introduced for propagating the desired clearances towards the next replanning process that might have a different amount of states, as calculated by Equation 19. Let $C_t = \{c_1^t, c_2^t, \ldots, c_n^t\}$ be the set of all desired clearances with $t \in T$ indicating the time the replanning process initiated, with τ be the time of the current planning query, and $\tau - \Delta t$ the time of the previous query. If $|C_\tau| = n_\tau$ and $|C_{\tau-\Delta t}| = n_{\tau-\Delta t}$ with $n_\tau \neq n_{\tau-\Delta t}$, the desired clearance is calculated for n_τ states as:

$$c_i^\tau = \begin{cases} \text{if } n_{\tau-\Delta t} < n_\tau : \\ \quad \max(\{c_j^{\tau-\Delta t} | j \in \\ \quad\quad [\min(\{1, i + n_{\tau-\Delta t} - n_\tau\}, \min(\{i, n_\tau\}]\}) \\ \text{if } n_{\tau-\Delta t} > n_\tau : \\ \quad \max(\{c_j^{\tau-\Delta t} | j \in [i, i + n_{\tau-\Delta t} - n_\tau]\}) \\ \text{if } n_{\tau-\Delta t} = n_\tau : \\ \quad c_i^{\tau-\Delta t} \end{cases} \tag{24}$$

The main idea is to maintain safety guarantees by informing the waypoints of the new — to be optimized — path with the largest clearance from the nearby waypoints of the previous.

E. Uncertainty Handling

Uncertainty, either related to map or the state, could be mitigated by inflating progressively the clearance appropriately for each waypoint. Given a bounded state uncertainty as a function of time $P()$, the additional safety margin is added to the already calculated clearance for each c_i, using the projected time instances $t_{w_i} \in T_w$ by Equation 4:

$$c_i^\tau = c_i + P(t_{w_i}). \tag{25}$$

Although not explored in this study, $P()$ could also be informed by a Gaussian Process *et al.* [40].

IV. Experimental Results

To validate the proposed methodology, extensive numerical experiments were performed with OpenRave [48] using a 3D model of BlueROV2 in various combinations of different environments, disturbances, and planning configurations. A machine with 7.7GB RAM, and an Intel® Core i7-7500 processor at 2.70GHz was utilized to evaluate computational needs for future deployment on autonomous underwater platforms. The main environments used in our simulations are shown in Figure 4; two static environments shown (a) and (b), referred to as m1 and m2 respectively, and a dynamic one called m3, two snapshots thereof shown in (c) and (d). The m1 and m2 were constructed in a way that 3D motion are not necessary, to focus on testing safety with lateral currents, while m3 required 3D motions. Four different cases for disturbances were simulated for each environment by adding translational errors during motion propagation with both constant and zero-mean error components.

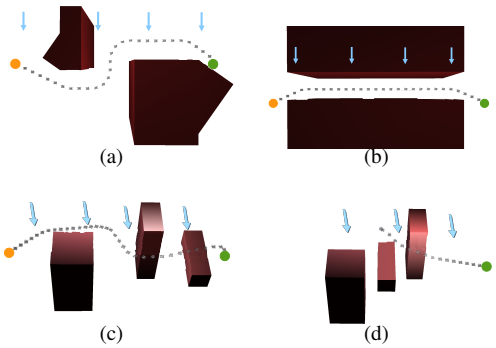

Fig. 4. The maps used for our evaluations along with instances of planned paths for an AUV moving from the initial (orange) to the goal (green), at a distance of 25 m and current direction shown with the blue arrows. (a) and (b) show the two static environments named m1 and m2 respectively. (c) and (d) shows the same dynamic environment of m3 consisting of 3 dynamic obstacles, moving with a constant speed of 0.3m/s in two different time instances along with the current solution path. In (c), the 2 left-most obstacles are moving perpendicular between the initial and goal configurations, and the third towards the initial position.

The safety of the proposed framework was tested for different planning configurations of ResiPlan, the new framework, and AquaNav, the old technique, indicated with "N" and "O" respectively. Linear velocities of 0.5m/s and 1m/s were tested, encoded with "s" and "f" respectively, and a variable step distance between the waypoints of 1.5m, 1.0m, and 0.5m, corresponding to "1", "2", and "3". The vehicle's assumed velocity is the hypothetical speed of the robot in error-free conditions, thus the resulted speed with respect to a static environment could diverge due to the disturbances. The clearance of AquaNav was chosen to be 0.5m, which is sufficient for solving a subset of the navigation problems.

The parameters of the new planners were intuitively chosen as follows: $\alpha = 1.1$ to allow for a potentially fast increase of the future error by 10%, $\epsilon = 0.1\text{m}$ as the minimum possible clearance, $\beta = 1.1$ for a potential 10% increase of the waypoint distance in the next planning query, $\gamma = 1$ assuming that the speed of the dynamic obstacles will not change severely, and finally the sliding window error queue E was chosen to keep only measurements from the last 10s. In the totality of the experiments, the minimum number of states considered during optimization was at minimum 10 in order to provide meaningful replanning frequency comparisons. AquaNav in most cases was able to replan with at least 150Hz, while due to extra computations, ResiPlan

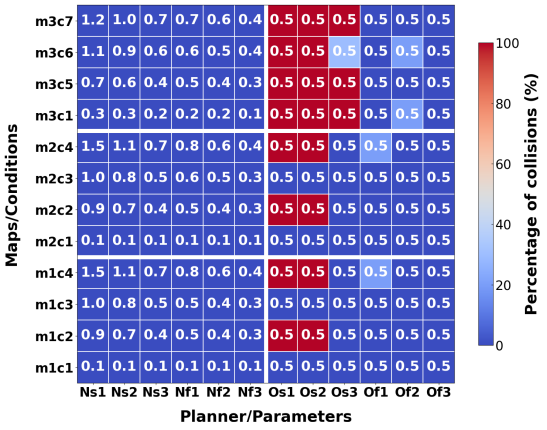

Fig. 5. The results in the environments of Figure 4 in different conditions, and for 6 different planning configurations. The color represents the percentage of collisions and numbers the average clearance used in meters.

was replanning at least around 30Hz; to our knowledge both frameworks surpass the requirements for real-time performance in the underwater domain by an order of magnitude.

The environments m1 and m2 were used with 4 different disturbances configurations: a) c1: no disturbances, b) c2: a constant downward current of $0.3\mathrm{m/s}$ with respect to the top-down perspective of Figure 4, c) c3: a combination of zero-mean disturbance on the three axis chosen independently for each from a uniform distribution $\mathcal{U}(-0.3, 0.3)\mathrm{m/s}$, and d) c4: the combination of c2 and c3.

For the m3 environment, given that with some conditions and planning setups safety could not be guaranteed since the combined speed of the obstacles at $0.3\mathrm{m/s}$ and the flow were surpassing the robot's velocity, different conditions were used: a) c5: reducing the constant current of c2 to $0.2\mathrm{m/s}$, b) c6: reducing the range of the uniform distribution of c3 to $\mathcal{U}(-0.2, 0.2)\mathrm{m/s}$, and c) c7: the combination of c5 and c6. Figure 5 shows the results of the numerical experiments for the combination of all different planning configurations and the aforementioned cases for 10 times each. No collisions took place with ResiPlan.

For m1 and m2, both ResiPlan and AquaNav are able to solve all the queries with no collision when no disturbances are present, though the proposed framework produced in general smaller safety margins. In the other cases, AquaNav suffered on solving queries where a constant current was present, which supports further the theoretical contributions of Section III. Moreover AquaNav was failing in almost all cases where ResiPlan was providing a larger clearance. As expected, longer waypoint step size and lower speeds are positively correlated to increase of collisions for AquaNav, and increase of safety distance for ResiPlan.

Additionally, regarding planning with dynamic obstacles in m3, an instance of the entire execution of a navigation query is shown in Figure 6, for a vehicle capable of holonomic 6D motions. As it can be noted, only a subset of waypoints at each replanning cycle (in proximity to the projected positions of the tracked obstacles) were using a larger clearance, while with the more conservative policy the clearance was maximized in the entire path.

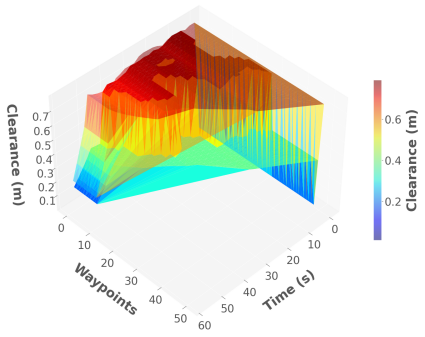

Fig. 6. An instance of using ResiPlan in m3c7 with configuration Ns3, from the initial (orange) to the goal state (green). The clearance is displayed as a function of the waypoint index and the time the robot needed to achieve the goal, for a conservative policy assigning the same clearance to all waypoints (shown with the transparent plot), and the improved enhancement utilizing proximity to dynamic obstacles (solid plot).

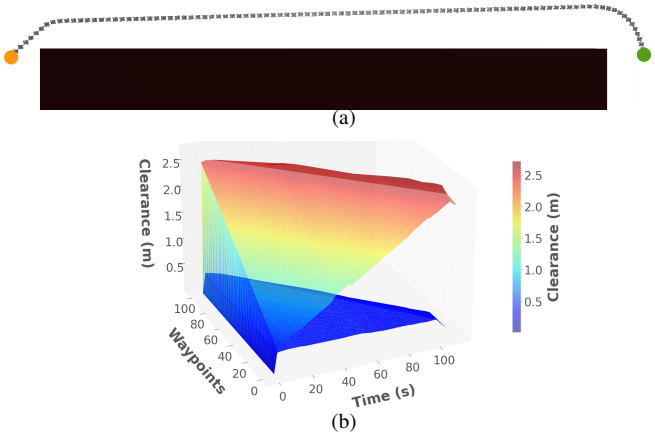

Fig. 7. In (a) the map considered to showcase the progressive clearance increase due to uncertainty of $0.02\mathrm{m/s}$. The planned path is shown for c2 conditions and Ns3 configuration, moving the robot at a distance of 50 m. In (b) a comparison of the proposed framework when accounting for uncertainty (transparent plot) and when not (solid plot).

Finally, to showcase the ability of ResiPlan to plan under uncertainty, a new longer environment was constructed to display the progressive increase on the clearance, and the resulted clearance plot for the entire execution is provided in Figure 7. It is worth noting that when not accounting for uncertainty the clearance is kept minimal and relatively constant, while in the other case, the clearance is progressively increased until the projected time to reach the goal.

V. Conclusions

To conclude, a novel general and efficient real-time underwater navigation planner was presented which enables safe autonomy under arbitrary disturbances in a holistic way by remaining agnostic, yet adaptive to the conditions of the environments and the performance of the path follower. The proposed framework requires minimal parameter tuning, and can sufficiently deal with currents, dynamic obstacles, and bounded uncertainty increase. Future work will focus on field testing in challenging dynamic underwater environments and extensions towards non-holonomic robotic systems.

REFERENCES

[1] Y. Guo, H. Liu, X. Fan, and W. Lyu, "Research progress of path planning methods for autonomous underwater vehicle," *Mathematical Problems in Engineering*, vol. 2021, 2021.

[2] M. Panda, B. Das, B. Subudhi, and B. B. Pati, "A comprehensive review of path planning algorithms for autonomous underwater vehicles," *International Journal of Automation and Computing*, vol. 17, no. 3, pp. 321–352, 2020.

[3] R. Kot, "Review of collision avoidance and path planning algorithms used in autonomous underwater vehicles," *Electronics*, vol. 11, no. 15, p. 2301, 2022.

[4] C. J. Green and A. Kelly, "Toward optimal sampling in the space of paths." in *International Symposium on Robotics Research (ISRR)*. Citeseer, 2007, pp. 281–292.

[5] R. A. Knepper and M. T. Mason, "Empirical sampling of path sets for local area motion planning," in *Experimental Robotics*. Springer, 2009, pp. 451–462.

[6] M. Pivtoraiko, R. A. Knepper, and A. Kelly, "Differentially constrained mobile robot motion planning in state lattices," *Journal of Field Robotics*, vol. 26, no. 3, pp. 308–333, 2009.

[7] M. S. Branicky, R. A. Knepper, and J. J. Kuffner, "Path and trajectory diversity: Theory and algorithms," in *International Conference on Robotics and Automation*. IEEE, 2008, pp. 1359–1364.

[8] J. D. Hernández, M. Moll, E. Vidal, M. Carreras, and L. E. Kavraki, "Planning feasible and safe paths online for autonomous underwater vehicles in unknown environments," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2016, pp. 1313–1320.

[9] E. Vidal, J. D. Hernández, N. Palomeras, and M. Carreras, "Online robotic exploration for autonomous underwater vehicles in unstructured environments," in *2018 OCEANS-MTS/IEEE Kobe Techno-Oceans (OTO)*. IEEE, 2018, pp. 1–4.

[10] H. Yuan and Z. Qu, "Optimal real-time collision-free motion planning for autonomous underwater vehicles in a 3d underwater space," *IET Control Theory & Applications*, vol. 3, no. 6, pp. 712–721, 2009.

[11] P. Fiorini and Z. Shiller, "Motion planning in dynamic environments using velocity obstacles," *The international journal of robotics research*, vol. 17, no. 7, pp. 760–772, 1998.

[12] D. Fox, W. Burgard, and S. Thrun, "The dynamic window approach to collision avoidance," *IEEE Robotics & Automation Magazine*, vol. 4, no. 1, pp. 23–33, 1997.

[13] L. Tychonievich, D. Zaret, J. Mantegna, R. Evans, E. Muehle, and S. Martin, "A maneuvering-board approach to path planning with moving obstacles," in *11th international joint conference on Artificial intelligence*, vol. 2, 1989, pp. 1017–1021.

[14] C. Lin, H. Wang, J. Yuan, D. Yu, and C. Li, "An improved recurrent neural network for unmanned underwater vehicle online obstacle avoidance," *Ocean Engineering*, vol. 189, p. 106327, 2019.

[15] M. R. Benjamin, M. DeFilippo, P. Robinette, and M. Novitzky, "Obstacle avoidance using multiobjective optimization and a dynamic obstacle manager," *IEEE Journal of Oceanic Engineering*, vol. 44, no. 2, pp. 331–342, 2019.

[16] W. Zhang, S. Wei, Y. Teng, J. Zhang, X. Wang, and Z. Yan, "Dynamic obstacle avoidance for unmanned underwater vehicles based on an improved velocity obstacle method," *Sensors*, vol. 17, no. 12, p. 2742, 2017.

[17] C. Lin, H. Wang, M. Fu, J. Yuan, and D. Yu, "A convolution neural network based obstacle avoiding method for unmanned underwater vehicle," *ICIC Express Letters*, vol. 13, no. 11, pp. 1079–1086, 2019.

[18] Y. Zhuang, S. Sharma, B. Subudhi, H. Huang, and J. Wan, "Efficient collision-free path planning for autonomous underwater vehicles in dynamic environments with a hybrid optimization algorithm," *Ocean Engineering*, vol. 127, pp. 190–199, 2016.

[19] C. Zammit and E. J. Van Kampen, "Real-time 3d uav path planning in dynamic environments with uncertainty," *Unmanned Systems*, pp. 1–17, 2022.

[20] A. Cardaillac and M. Ludvigsen, "Ruled path planning framework for safe and dynamic navigation," in *OCEANS 2021: San Diego–Porto*. IEEE, 2021, pp. 1–7.

[21] F. Gao, S. B. Moltu, E. R. Vollan, S. Shen, and M. Ludvigsen, "Increased autonomy and situation awareness for rov operations," in *Global Oceans 2020: Singapore–US Gulf Coast*. IEEE, 2020, pp. 1–8.

[22] M. Føre, S. Fjæra, S. J. Ohrem, E. Kelasidi, N. Bloecher, and H. B. Amundsen, "Adaptive motion planning and path following for permanent resident biofouling prevention robot operating in fish farms," in *OCEANS 2021: San Diego–Porto*. IEEE, 2021, pp. 1–10.

[23] R. A. Knepper, S. S. Srinivasa, and M. T. Mason, "Toward a deeper understanding of motion alternatives via an equivalence relation on local paths," *The International Journal of Robotics Research (IJRR)*, vol. 31, no. 2, pp. 167–186, 2012.

[24] R. A. Knepper and M. T. Mason, "Real-time informed path sampling for motion planning search," *The International Journal of Robotics Research*, vol. 31, no. 11, pp. 1231–1250, 2012.

[25] C. Petres, Y. Pailhas, P. Patron, Y. Petillot, J. Evans, and D. Lane, "Path planning for autonomous underwater vehicles," *IEEE Transactions on Robotics*, vol. 23, no. 2, pp. 331–341, 2007.

[26] S. Williams, G. Dissanayake, and H. Durrant-Whyte, "Towards terrain-aided navigation for underwater robotics," *Advanced Robotics*, vol. 15, no. 5, pp. 533–549, 2001.

[27] J. Moulton, N. Karapetyan, M. Kalaitzakis, A. Q. Li, N. Vitzilaios, and I. Rekleitis, "Dynamic autonomous surface vehicle controls under changing environmental forces," *arXiv preprint arXiv:1908.02850*, 2019.

[28] A. Alvarez, A. Caiti, and R. Onken, "Evolutionary path planning for autonomous underwater vehicles in a variable ocean," *Journal of Oceanic Engineering, IEEE*, vol. 29, no. 2, pp. 418–429, 2004.

[29] N. K. Yilmaz, C. Evangelinos, P. F. Lermusiaux, and N. M. Patrikalakis, "Path planning of autonomous underwater vehicles for adaptive sampling using mixed integer linear programming," *Journal of Oceanic Engineering, IEEE*, vol. 33, no. 4, pp. 522–537, 2008.

[30] C. V. Caldwell, D. D. Dunlap, and E. G. Collins, "Motion planning for an autonomous underwater vehicle via sampling based model predictive control," in *OCEANS 2010 – Seattle*. IEEE, 2010, pp. 1–6.

[31] D. N. Subramani and P. F. Lermusiaux, "Risk-optimal path planning in stochastic dynamic environments," *Computer Methods in Applied Mechanics and Engineering*, vol. 353, pp. 391–415, 2019.

[32] È. Pairet, J. D. Hernández, M. Lahijanian, and M. Carreras, "Uncertainty-based online mapping and motion planning for marine robotics guidance," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 2367–2374.

[33] B. Garau, M. Bonet, A. Alvarez, S. Ruiz, and A. Pascual, "Path planning for autonomous underwater vehicles in realistic oceanic current fields: Application to gliders in the western mediterranean sea," *Journal of Maritime Research*, vol. 6, no. 2, pp. 5–22, 2009.

[34] I. Tusseyeva, S.-G. Kim, and Y.-G. Kim, "3d global dynamic window approach for navigation of autonomous underwater vehicles," *International Journal of Fuzzy Logic and Intelligent Systems*, vol. 13, no. 2, pp. 91–99, 2013.

[35] C. S. Tan, R. Sutton, and J. Chudley, "An incremental stochastic motion planning technique for autonomous underwater vehicles," *IFAC Proceedings Volumes*, vol. 37, no. 10, pp. 483–488, 2004.

[36] J. McMahon and E. Plaku, "Mission and motion planning for autonomous underwater vehicles operating in spatially and temporally complex environments," *IEEE Journal of Oceanic Engineering*, vol. 41, no. 4, pp. 893–912, 2016.

[37] H. Yu, W. Lu, and D. Liu, "A unified closed-loop motion planning approach for an i-auv in cluttered environment with localization uncertainty," in *International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 4646–4652.

[38] M. S. Wiig, K. Y. Pettersen, and T. R. Krogstad, "A 3d reactive collision avoidance algorithm for underactuated underwater vehicles," *Journal of Field Robotics*, vol. 37, no. 6, pp. 1094–1122, 2020.

[39] M. Xanthidis, N. Karapetyan, H. Damron, S. Rahman, J. Johnson, A. O'Connell, J. M. O'Kane, and I. Rekleitis, "Navigation in the presence of obstacles for an agile autonomous underwater vehicle," in *International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 892–899.

[40] È. Pairet, J. D. Hernández, M. Carreras, Y. Petillot, and M. Lahijanian, "Online mapping and motion planning under uncertainty for safe navigation in unknown environments," *Transactions on Automation Science and Engineering, IEEE*, 2021.

[41] E. Vidal, M. Moll, N. Palomeras, J. D. Hernández, M. Carreras, and L. E. Kavraki, "Online multilayered motion planning with dynamic constraints for autonomous underwater vehicles," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 8936–8942.

[42] J. J. H. Lee, C. Yoo, R. Hall, S. Anstee, and R. Fitch, "Energy-optimal kinodynamic planning for underwater gliders in flow fields," in *Australasian Conference on Robotics and Automation (ACRA)*, 2017.
[43] J. D. Hernández, E. Vidal, M. Moll, N. Palomeras, M. Carreras, and L. E. Kavraki, "Online motion planning for unexplored underwater environments using autonomous underwater vehicles," *Journal of Field Robotics*, vol. 36, no. 2, pp. 370–396, 2019.
[44] J. Schulman, Y. Duan, J. Ho, A. Lee, I. Awwal, H. Bradlow, J. Pan, S. Patil, K. Goldberg, and P. Abbeel, "Motion planning with sequential convex optimization and convex collision checking," *The International Journal of Robotics Research*, vol. 33, no. 9, pp. 1251–1270, 2014.
[45] J. Nocedal and S. J. Wright, *Numerical optimization*. Springer, 1999.
[46] E. Gilbert, D. Johnson, and S. Keerthi, "A fast procedure for computing the distance between complex objects in three space," in *International Conference on Robotics and Automation (ICRA)*, vol. 4. IEEE, 1987, pp. 1883–1889.
[47] G. Van Den Bergen, "Proximity queries and penetration depth computation on 3d game objects," in *Game developers conference*, vol. 170, 2001.
[48] R. Diankov and J. Kuffner, "Openrave: A planning architecture for autonomous robotics," *Robotics Institute, Pittsburgh, PA, Tech. Rep. CMU-RI-TR-08-34*, vol. 79, 2008.