Effective Team Coordination through Intra-Robot Replanning to Restore Team Plan Rationale*

Philip Cooksey
Robotics Institute, Carnegie Mellon University,
Pittsburgh, Pennsylvania, USA.
pcooksey@andrew.cmu.edu

Manuela Veloso
School of Computer Science, Carnegie Mellon University,
Pittsburgh, Pennsylvania, USA.
veloso@cs.cmu.edu

***Abstract*—We consider multiple autonomous underwater vehicles (AUVs) engaged in oceanic missions that require collaboration. In this paper, we address the challenge of effective distributed multi-robot plan execution: when a failure occurs, an individual robot needs to decide how to proceed to maintain the success of the multi-robot plan. Failures occur when the modeled actions of the AUVs fail to result in their expected outcome, largely due in this domain to the poorly modeled dynamics of the ocean. Intra-robot replanning is advantageous because traditional purely centralized controlled methods use time-consuming and expensive satellite communication in order to repair all failures. Therefore, we present a rationale-driven team plan representation that explicitly includes the rationale of the individual robots' planned actions. We then present our intra-robot rationale-driven algorithm that enables the selection and execution of replan policies for individual robots to overcome failure. The algorithm's selection is improved through learning the predicted cost of each replan policy, which ultimately improves the overall team performance. This paper is focused on examples of concrete team missions and the collaboration of AUVs that clearly demonstrates the effectiveness of our approach.**

I. Introduction and Domain

We focus on multiple autonomous underwater vehicles (AUVs), shown in Figure 1, in the dynamic ocean environment. Specifically, we are concerned with team plan failures and replanning opportunities for AUVs while they are tracking an ocean front, a barrier between two bodies of water. In Figure 2, there is a visible ocean front where fresh river water is mixing into ocean salt water. An ocean front is full of life and biological material, and it is a hot spot for oceanographic research in understanding their function in the ocean ecosystem. The AUVs are used to collect valuable samples in and around the ocean front. Some research has already been conducted on AUVs tracking water fronts in the ocean [1], [2], [3]. For our simulated experiments, we have multiple AUVs follow the same path, synchronizing their movements in order to collect different samples at different heights in the water column, so that they can provide more information about the ocean front at the same timestamp. The AUVs are not particularly fast, moving at roughly 2.5 knots, so multiple AUVs give us more data before the ocean front moves or changes.

*This research is partially sponsored by DARPA under agreements FA87501620042 and FA87501720152 and NSF grant number IIS1637927. The views and conclusions contained in this document are those of the authors only.

Fig. 1. Three Light Autonomous Underwater Vehicles (LAUVs) from the Laboratório de Sistemas e Tecnologia Subaquática (LSTS) in Porto, Portugal.

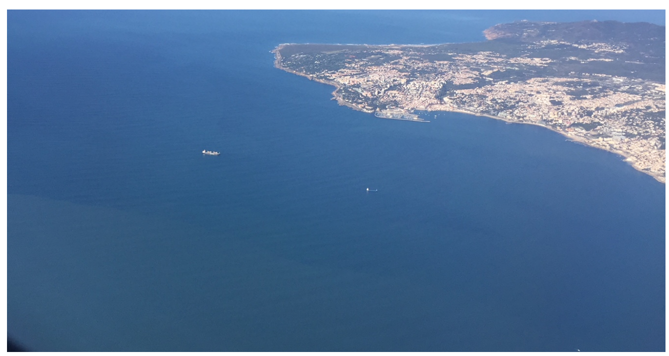

Fig. 2. Coastal waters off of southern Portugal. Discoloration, near bottom left corner, shows the outline of an ocean front between river and ocean water.

The dynamics of the ocean pose some challenges for multi-robot planning for AUVs. The ocean currents are dynamic, unknown, and can push the AUVs off their target locations or desired paths when they are operating underwater. Communication with the AUVs is, for the most part, only possible at the surface and over satellite, and satellite communication is costly and takes a considerable amount of time to complete [4]. In short, the AUVs are operating in a dynamic environment that can cause failures in their executed actions and communicating with the centralized team planner to fix those failures is costly and slow.

II. Problem

The standard approach for operating multiple AUVs is to have a centralized controller plan the actions for all of the

Algorithm 1 Intra-Robot Replanning algorithm for selecting applicable replan policies, sorting them, and then executing the policies until the rationales are true.

Require: $\mathcal{F} \wedge \mathcal{S} \wedge \mathcal{R} \wedge \mathcal{P}$

1: **for all** $\mathcal{F}_i \in \mathcal{F}$ **do**
2:   **for all** $\mathcal{P}_i \in \mathcal{P}$ **do** # Find applicable replan policies
3:     **if** $\mathcal{F}_i \subseteq \beta \in \mathcal{P}_i$ **then**
4:       $\mathbb{P}$.push($\mathcal{P}_i$)
5:     **end if**
6:   **end for**
7:   $\mathbb{P} \leftarrow$ sort($\mathbb{P}$)
8:   **repeat**
9:     $\mathbb{P}.front(\alpha)$ # Execute with $\alpha \subseteq \mathcal{S} \wedge \mathcal{R}$
10:     $\mathbb{P}$.pop()
11:   **until** $\mathcal{F}_i(p_1, ..., p_n) = true$
12: **end for**

AUVs [5]. With this approach, the AUVs simply execute actions blindly until a failure occurs and then wait for the centralized controller to provide a new plan. However, this method can be very costly and time consuming when considering that every failure will require the AUV to communicate over satellite and waste valuable time that would be better spent collecting scientific data.

A better approach is to have the AUVs repair failures locally and only use satellite communication when absolutely necessary. However, current AUVs are not able to replan locally to fix their failures. Moreover, if they had such an ability, they would still lack the information necessary for choosing an appropriate replan action. As an example, consider that the AUV has failed to be At a certain location, and therefore the effects of the current action and some precondition(s) of the next action have failed. If the AUV is going to fix the failed action, it would need to consider the action's rationales, if the failure can be fixed locally, and how to fix it.

III. Method

In order to properly fix a failure, the AUV needs to understand the rationale behind why the action was chosen by the centralized team planner. Without the rationale, the AUV may try to fix the failure by doing something counter-productive towards the centralized team planner's overall objective. For example, if the AUV was assigned to get a time-sensitive water sample but fails, then attempting to redo the exact same action may violate the time limit that the centralized team planner understood. Instead, if the AUV knew the rationale of the chosen action was to collect a sample within a specific area, it may choose to collect a sample in a slightly different but still acceptable location within the time limit. We add these rationales to the team plan to provide the AUV with more information on why actions were picked by the team planner, previously described in [6].

In order to determine if the failure can be fixed and what actions to take to fix it, we use intra-robot replanning, theoretically formulated in [7]. We define our intra-robot replanning problem, as described in [6], as a tuple $\langle \mathcal{R}, \mathcal{S}, \mathcal{P}, \mathcal{T}, \mathcal{F} \rangle$:

- $\mathcal{R}$ is the set of robots on the team;
- $\mathcal{S}$ is the current state;
- $\mathcal{P}$ is the set of replan policies (i.e., replan tactics); where a policy $\mathcal{P}_j = \langle \alpha, \beta \rangle$:
  - α are the input parameters to the replan policy,
  - β are the rationales that can be enabled by this policy (effects);
- $\mathcal{T}$ is a set of rationales for the current tactic;
- $\mathcal{F} \subseteq \mathcal{T}$ are the rationales that are invalid, i.e., failing.

For algorithm 1, the AUVs each have a set of replan policies, $\mathcal{P}$, that can restore certain team plan rationales when they fail, $\mathcal{F}$. In lines 1-6, the intra-robot replanning algorithm selects applicable replan policies based on line 3 where the team plan rationale must be a subset of the team plan rationales that $\mathcal{P}_i$ can enable. On line 7, the algorithm sorts the policies based on some cost function. In lines 8-11, the policies are executed to restore the team plan rationale locally. We assume there is always at least one replan policy that calls the centralized team planner thereby solving any failure, ultimately ending the final loop. We have shown in previous work the benefits of using intra-robot replanning over a purely centralized method [6], [8]. In this paper, we expand upon those experiments and delve deeper into the advantages and disadvantages of using intra-robot replanning and team plan rationale with AUVs in the ocean.

IV. Implementation

This section concerns the implementation details of using intra-robot replanning by AUVs in the ocean. First, we will detail the actions used by AUVs in our simulated domain. Then we will describe the development of team plan rationales and replan polices for the intra-robot replanning. Lastly, we will discuss the computational and communicational budgets of using intra-robot replanning and team plan rationale.

A. Actions

The actions used in our experiments are:

1) Initial: Starting action for AUV.
2) Connect ⟨ID⟩: Connect over wifi to the AUV with the defined ID.
3) Goto ⟨X⟩ ⟨Y⟩: The AUV stays on the surface and heads directly to the (X, Y) location. We assume it can receive a GPS signal so it does not get lost, but it cannot receive a wifi signal unless it stops.
4) Yoyo ⟨X⟩ ⟨Y⟩: The AUV performs a series of descents and ascents in the water column towards the (X, Y) location. As the AUV is underwater, it cannot receive a GPS or wifi signal so the Yoyo maneuver oftentimes does not arrive at the destination location.
5) Satellite: The AUV connects to the centralized controller through a satellite link.

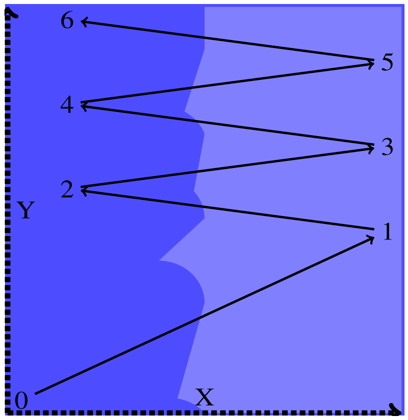

Fig. 3. A simulated environment where two AUVs criss-cross the boundary line of fresh water (light blue) and salt water (dark blue) to gather scientific data about the boundary.

B. Team plan rationale

The team plan rationale are derived from the planning problem and/or the human operator developing the team plan. We will describe some team plan rationales from our example domain in which two AUVs are criss-crossing the boundary line of a water front, shown in Figure 3. We define a rationale as a function with a finite set of parameters:

$$f(p_1, p_2, ..., p_n) = true \lor false \tag{1}$$

that returns true or false given those specific parameters.

Imagine the AUV must take a water sample precisely $AT\{X = 1, Y = 2\}$, a typical constraint in planning. We can define this constraint as the following rationale:

$$\text{AT}(x, y, 1, 2) = x \equiv 1 \land y \equiv 2 \tag{2}$$

where x and y are the location variables of the AUV. This rationale is only true if the AUV is at the specific location. The two constants can be replaced for any location where the rationale for the plan defines that the AUV must be at a defined location.

Imagine the human operator wants to be more lenient with the location assuming the AUVs will be near enough to the location. We could define the NEAR rationale:

$$\begin{aligned}\text{NEAR}(x, y, l_x, l_y, d_x, d_y) =\\ (|x - l_x| < d_x) \land (|y - l_y| < d_y)\end{aligned} \tag{3}$$

where x, y are the AUV's location, l_x, l_y is the destination location, and d_x, d_y are the distance thresholds.

Two or more AUVs must connect over wifi to synchronize before moving to the next location. We assume the AUV's state, $\mathcal{S}$, defines if the AUV is connected over wifi, $\mathbb{C}$, and to a defined ID, $\mathbb{C}_{ID}$. We define the CANCONNECT rationale:

$$\begin{aligned}\text{CANCONNECT}(\mathbb{C}, \mathbb{C}_{ID}, ID) =\\ (\mathbb{C} = true) \land (\mathbb{C}_{ID} = ID)\end{aligned} \tag{4}$$

These team plan rationale add information about why certain locations were picked (to fulfill AT or NEAR), or about an assumption made by the planner or human operator that both AUVs should be able to connect with one another.

C. Replan Policies

When designing the replan policies for the AUVs, the team plan rationales need to be considered. We will only consider the CANCONNECT rationale during the experiments due to limited space for evaluating the different experimental parameters. As previously stated, there is an assumption that there exists at least one replan policy that enables any rationale. The most obvious in our domain is a replan policy that simply calls the satellite, and we also considered another alternative:

1) R-SATELLITE $\langle\beta$ = any rationale$\rangle$: SATELLITE to receive a new plan.
2) R-GOTO-SATELLITE $\langle\beta$ = any rationale$\rangle$: Uses GOTO to the failed location and then SATELLITE.

For the CANCONNECT rationale, we designed a few different options that will attempt to reconnect with the other AUV when it fails to connect over wifi:

1) R-GOTO $\langle\beta = \text{CANCONNECT}()\rangle$: Uses GOTO to the failed location and then CONNECT.
2) R-CONNECT-GOTO $\langle\beta = \text{CANCONNECT}()\rangle$: Attempts CONNECT, and then proceeds like R-GOTO.
3) R-STAR-GOTO $\langle\beta = \text{CANCONNECT}()\rangle$: Attempts CONNECT. Then centered at its current location, it creates five evenly spaced points on the circumference of a circle with a radius equal to the wifi distance. It will GOTO each location and attempt to CONNECT. The first location is chosen at random and then it always proceeds to the point farthest away from its current location until all points have been attempted. If unsuccessful, it proceeds like R-GOTO. This was inspired by the sector search pattern described in [9] for search and rescue, but was changed to ensure that the AUVs always traveled the same distance before attempting to connect.

For these three replan policies, an attempted CONNECT lasts 3 seconds and it uses the ID provided by the rationale.

D. Computation and communication costs

The intra-replanning algorithm is computationally efficient for an AUV. The replan polices are simple state-machines that are selected by their applicability and their cost. This will not be very taxing for the on-board computation of the AUVs. The communication cost is slightly increased when the team plan is delivered to the AUVs due to the addition of rationales which come with each action e.g.,

YOYO ⟨X⟩ ⟨Y⟩; CANCONNECT(⟨ID⟩);

The communication cost saved from replanning is significant enough to outweigh the addition of some communication overhead due to the addition of team plan rationales.

V. EXPERIMENTS

We designed a simulated environment to test different ocean currents, communication times, and replan policies for the AUVs. The ocean currents are modeled using simple random distributions that can potentially push the AUVs off their intended courses. The ocean current is applied once after the

TABLE I
TEAM PLAN PROVIDED TO AUVS WITH RATIONALES

AUV1, ID = 1	**AUV2, ID = 2**
INITIAL	INITIAL
CONNECT 2	CONNECT 1
YOYO 8 4; CANCONNECT(2)	YOYO 8 4; CANCONNECT(1)
CONNECT 2	CONNECT 1
YOYO 1 5; CANCONNECT(2)	YOYO 1 5; CANCONNECT(1)
CONNECT 2	CONNECT 1
YOYO 8 6; CANCONNECT(2)	YOYO 8 6; CANCONNECT(1)
CONNECT 2	CONNECT 1
YOYO 1 7; CANCONNECT(2)	YOYO 1 7; CANCONNECT(1)
CONNECT 2	CONNECT 1
YOYO 8 8; CANCONNECT(2)	YOYO 8 8; CANCONNECT(1)
CONNECT 2	CONNECT 1
YOYO 1 9; CANCONNECT(2)	YOYO 1 9; CANCONNECT(1)
END	END

AUV surfaces from under the water in order to simplify the computation. We assume that surfaced AUVs can use GPS to localize and adjust their headings as the current pushes them, but when operating underwater, the AUVs cannot adjust their course. While previous work has modeled highly detailed ocean currents, this level of simulation goes beyond our current scope [10]. The satellite communication link between the AUVs and the centralized team planner can be adjusted to understand the effect that the communication time has on the benefits of intra-robot replanning. The rationales added to the team plans can also affect the efficacy of intra-robot replanning because it will change the selection of replan policies.

A. Constants

We assume everything in the simulation is defined in kilometers and seconds. The constants below are assumed unless otherwise discussed in each individual experiment:

1) AUV speed is 0.00128 km/s
2) Wifi radius ($\mathbb{R}$) is 0.1 km
3) Satellite cost ($\mathbb{S}$) is 600 seconds (10 mins)
4) Wifi cost is 3 seconds
5) Maximum wait time ($\mathbb{W}$) for a wifi connection is 300 seconds (5 mins)

For the team plan, the AUVs will criss-cross a boundary line using the action YOYO to each location. The map is shown in Figure 3. The team plan used in all the experiments is shown in Table I, the x- & y-axis are in kilometers. The CANCONNECT rationale has been added to the YOYO with the assumption that after the yoyo maneuver is finished the AUVs should be able to connect over wifi.

B. Satellite communication costs

We ran multiple experiments to test the effects of different satellite costs to show the increasing benefits of intra-robot replanning. For each experiment, a single replan policy was run two hundred times and then the team plan completion times were averaged. We assumed the satellite replan policy was still available if the other replan policy failed. The satellite cost was then changed and the process repeated for each replan policy. The x-axis ocean current was modeled with a normal distribution with mean 1 and variance 0.02, and the y-axis had no ocean current. With a mean of 1, the x-axis ocean current is going to push the AUVs off from their intended target locations causing the AUVs to replan.

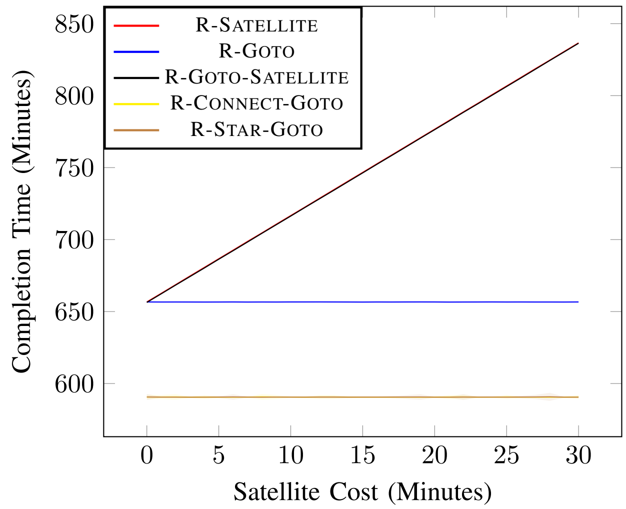

Fig. 4. The relationship between team plan completion time and the satellite cost used when replanning for each replan policy. Brown and yellow lines are nearly identical, as are black and red. See Section V-B.

As shown in Figure 4, the cost of communicating with the satellite can greatly increase the time for the AUVs to complete their assigned team plan. On the other hand, the replan polices R-GOTO, R-CONNECT-GOTO, & R-STAR-GOTO remain constant because they can successfully replan without needing to connect to the satellite. Although, this is affected by the type of currents and their variances in our modeled ocean currents.

C. The effect of modeled ocean variance

In the previous section, we demonstrated that some replan policies can greatly reduce the time it takes to complete a team plan, but the ocean current's variance was static at 0.2 km. In this section, we vary the ocean current variance to see the effect on the overall performance of each replan policy. For this experiment, we have modeled both the x-axis and y-axis by a normal distribution. Both have a mean of zero and the variance will go from 0 to 0.5 km stepping by 0.02 km. The experiment is run 400 times for each replan policy for each variance.

Shown in Figure 5, the increase in ocean variance steadily increases the team plan completion time. Of course, with more ocean variance the AUVs may have to travel farther that will automatically increase the total time. The graph shows that as the ocean variance increases the AUVs are less likely to be successful when using the replan policy. The shaded area around the graph lines show one standard deviation that increases as the ocean variance increases. Surprisingly, R-STAR-GOTO performance degrades quickly, eventually crossing beyond R-SATELLITE. This demonstrates that R-STAR-GOTO is not very useful with large variance in

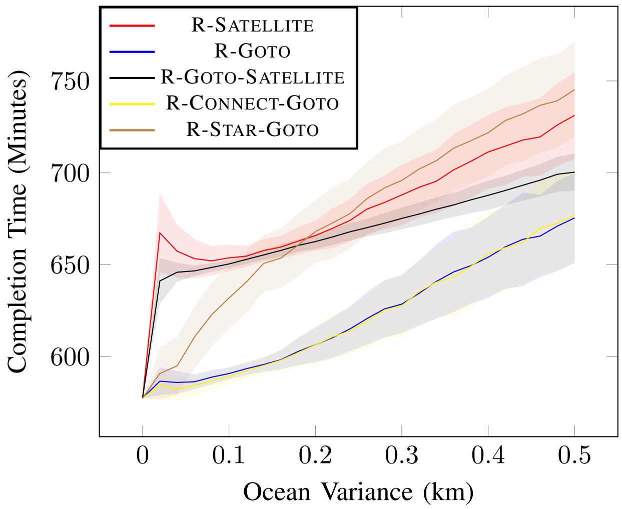

Fig. 5. The relationship between team plan completion time and variance of the ocean currents for each replan policy. Highlighted areas show 1σ standard deviation. See Section V-C.

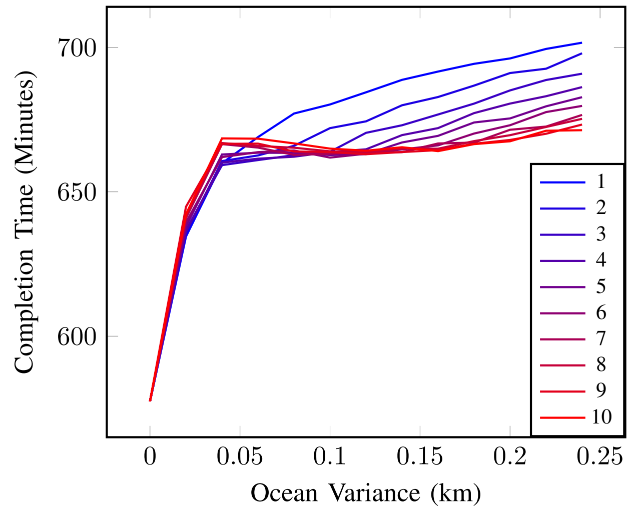

Fig. 6. The relationship between team plan completion time, the variance of the ocean currents, and the wifi wait time for R-SATELLITE. Multiply the legend by 30 seconds to get the wifi wait time. See Section V-D.

the ocean currents. R-CONNECT-GOTO and R-GOTO have not yet reached R-GOTO-SATELLITE so they successfully replan locally sometimes even with the very large ocean variance. An interesting aspect of the graph is the u-shaped curve between 0 and 0.2 variances for R-SATELLITE. The initial sharp increase highlights the quick increase in cost when the ocean variances gets just large enough to cause failures for the AUVs. However, the cost goes down afterwards for awhile and then picks back up. We will explore why in the next section.

D. *Maximum wait time* ($\mathbb{W}$)

A rather interesting observation was made about the simple constant $\mathbb{W}$. This is the time that the AUVs will attempt to connect over wifi before failing and calling the satellite. In the previous section, we described a u-shaped curve in the performance of R-SATELLITE that we will attribute to $\mathbb{W}$ in this section. For this experiment, we simplified the ocean current to only a normal distribution on the x-axis with mean 0 and variance will range from 0 to 0.25 km incrementing by 0.02 km. $\mathbb{W}$ will go from 30 seconds to 5 minutes incrementing by 30 seconds. The experiment is run 400 times for each ocean variance and $\mathbb{W}$.

In Figure 6, we show the results of R-SATELLITE where each line is a different $\mathbb{W}$. Reducing the ocean variance to one axis reduced the u-shape curve but it remains prominent. Clearly, $\mathbb{W}$ causes the u-shape as it goes away once the value is very low. There are three cases to consider: 1) both AUVs are near the target location, 2) only one is near the target location, or 3) both are far from the target location. We will consider the cost of the two AUVs reuniting using R-SATELLITE. The first case has cost zero because they are both correctly at the target location, i.e., no reunion cost. The next two scenarios are more complicated and their combination results in the u-shape curve. We assume that $\mathbb{W} < \mathbb{S}$.

Consider the second scenario where one AUV is near the target location and the other is not. Let T_1 be the time it takes for the far AUV to get to the target location. For simplicity, we assume that $T_1 \leq 2\mathbb{W} + \mathbb{S}$.

$$Cost = \begin{cases} \mathbb{W} + \mathbb{S} & T_1 \leq \mathbb{W} \\ T_1 + \mathbb{S} & \mathbb{W} < T_1 \leq 2\mathbb{W} \\ T_1 + 2\mathbb{S} + \mathbb{W} & 2\mathbb{W} < T_1 \leq 2\mathbb{W} + \mathbb{S} \end{cases} \tag{5}$$

There is a large initial cost for a higher wifi wait time in the first case. Once T_1 is larger than $\mathbb{W}$, the cost increases with T_1 until it passes $2\mathbb{W}$ where the waiting AUV must then call the satellite for the second time, greatly increasing the cost. With a small ocean variance, the cost is mostly attributed to the first case.

Consider the third scenario where both AUVs are far away from the target location. Let T_1 and T_2 be the times it takes for the AUVs to get to the target location. Let $T_d = |T_1 - T_2|$. For simplicity, we assume that $T_d \leq 2\mathbb{W} + \mathbb{S}$.

$$Cost = \begin{cases} Max(T_1, T_2) + \mathbb{S} & T_d \leq \mathbb{W} \\ Max(T_1, T_2) + 2\mathbb{S} + \mathbb{W} & \mathbb{W} < T_d \leq 2\mathbb{W} + \mathbb{S} \end{cases} \tag{6}$$

In this scenario, $\mathbb{W}$ does not increase the cost for the first case, but it does increase the threshold before jumping into the more costly next case, which is beneficial. With these equations we can give a simple intuitive example on why the u-shape occurs.

Let $\mathbb{S}$ be 10 minutes and $\mathbb{W}$ be 5 minutes. For the second scenario, let T_1 be 2 minutes. The cost is then $\mathbb{W} + \mathbb{S} = 15$ minutes. For the third scenario, let T_1 be 3 minutes and T_2 be 4 minutes. The cost is then $Max(T_1, T_2) + \mathbb{S} = 14$ minutes. Therefore, even though in the third scenario the AUVs were farther out, the total cost for reuniting was less. The additional cost for the second scenario is because the AUV at the target location will waste time trying to connect and then connect over satellite, which has a strict cost of 10 minutes.

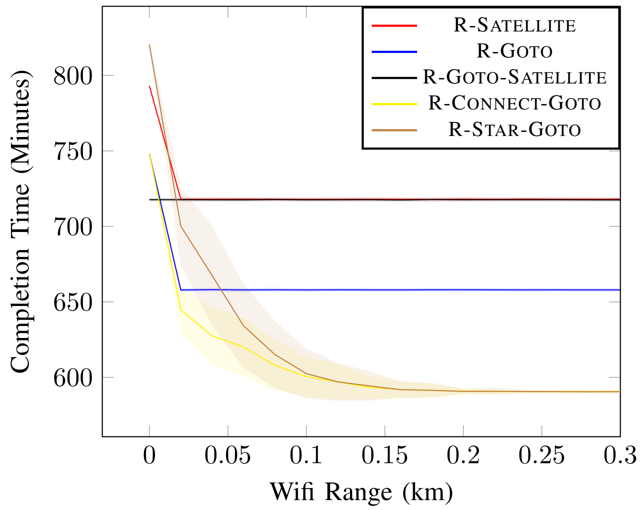

Fig. 7. The relationship between team plan completion time and the wifi range for each replan policy. Highlighted area shows 1σ. See Section V-E.

E. Wifi range

The range of the wifi signal can have implications on the usability of intra-robot replanning with AUVs. We have assumed that there exists a wifi signal with enough range that the AUVs can communicate locally. However, the range of the signal may not be that far or non-existent at all, e.g., on a very turbulent ocean surface.

In Figure 7, we ran multiple experiments testing different wifi range values for all the replan polices. We went from 0 to 0.3 km incrementing by 0.02 km. For ocean currents, we have a normal distribution on the x-axis with mean 1 and variance 0.05 and no y-axis current. Each replan policy was run 400 times for each wifi range.

Notably, R-GOTO-SATELLITE performs the best when the wifi range is zero. This is because it goes immediately to the target location and then calls the satellite. The AUVs sync over the satellite without the need for wifi. While on the other hand, R-SATELLITE does worse because in the worst case the AUVs will call the satellite twice, first at the failed location and then at the target location to sync. R-STAR-GOTO does the worst because it wastes so much time attempting to connect over wifi. As the wifi range expands, the improvements of R-GOTO, R-CONNECT-GOTO, and R-STAR-GOTO can be seen. Therefore, it is important to understand the wifi capabilities of the AUVs using intra-robot replanning and the effect it will have on the performance of an individual replan policy.

VI. CONCLUSION

Intra-robot replanning can be very beneficial when considering replanning for multiple AUVs in the ocean. We have showed how replanning locally rather than calling the satellite for every failure can greatly reduce the total time it takes for the AUVs to complete their assigned team plan. We then detailed how different variables can have profound effects on the performance of intra-robot replanning. Ocean currents, wifi range, and the time cost for communicating over satellite can change the performance of individual replan policies. The cost function for selecting replan policies should attempt to take these variables into account if possible. Another option is to learn the cost of each replan policy. Gathering enough data to properly learn the cost functions for each replan policy can be arduous. On the other hand with long-term autonomy, there may be enough time to let the AUVs gather data when using different replan policies and slowly learn an effective cost function. We believe this would be a valuable research path to explore.

REFERENCES

[1] Y. Zhang, M. Godin, J. G. Bellingham, and J. P. Ryan, “Ocean front detection and tracking by an autonomous underwater vehicle,” in *OCEANS’11 MTS/IEEE KONA*, Sept 2011, pp. 1–4.

[2] B. L. Reed and F. S. Hover, “Tracking ocean fronts with multiple vehicles and mixed communication losses,” in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Nov 2013, pp. 3374–3381.

[3] R. N. Smith, P. Cooksey, F. Py, G. S. Sukhatme, and K. Rajan, *Adaptive Path Planning for Tracking Ocean Fronts with an Autonomous Underwater Vehicle*. Cham: Springer International Publishing, 2016, pp. 761–775.

[4] S. Ferreira, J. Pereira, P. Dias, J. Sousa, T. A. Johansen, P. Loureno, and V. E. Hovstein, “Managing communication challenges in vehicle networks for remote maritime operations,” in *OCEANS 2017 - Aberdeen*, June 2017, pp. 1–7.

[5] Z. Yan, N. Jouandeau, and A. A. Cherif, “A survey and analysis of multi-robot coordination,” *International Journal of Advanced Robotic Systems*, vol. 10, December 2013.

[6] P. Cooksey and M. Veloso, “A rationale-driven team plan representation for autonomous intra-robot replanning,” in *Proceedings of IROS’18, the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Madrid, Spain, October 2018.

[7] K. Talamadupula, D. Smith, W. Cushing, and S. Kambhampati, “A theory of intra-agent replanning,” *ICAPS 2013 Workshop on Distributed and Multi-Agent Planning (DMAP)*, 2013.

[8] P. Cooksey and M. Veloso, “Intra-robot replanning to enable team plan conditions,” in *Proceedings of IROS’17, the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Vancouver, Canada, September 2017.

[9] S. Bernardini, M. Fox, and D. Long, “Combining temporal planning with probabilistic reasoning for autonomous surveillance missions,” *Autonomous Robots*, vol. 41, no. 1, pp. 181–203, Jan 2017.

[10] E. Omerdic and D. Toal, “Modelling of waves and ocean currents for real-time simulation of ocean dynamics,” in *OCEANS 2007 - Europe*, June 2007, pp. 1–6.