Planning for Fish Net Inspection with an Autonomous OSV

Tony X. Lin

Electrical and Computer Engineering
Georgia Institute of Technology
Atlanta, GA USA
tlin339@gatech.edu

Qiuyang Tao
Electrical and Computer Engineering
Georgia Institute of Technology
Atlanta, GA USA
qtao7@gatech.edu

Fumin Zhang

Electrical and Computer Engineering

Georgia Institute of Technology

Atlanta, GA USA

fumin@gatech.edu

Abstract—In aquaculture farming, escaping fish can lead to large economic losses and major local environmental impacts. As such, the careful inspection of fishnets for breaks or holes presents an important problem. In this paper, we extend upon our previous work in the design of an omnidirectional surface vehicle (OSV) for fishnet inspection by incorporating AI (artificial intelligence) planning methods. For large aquaculture sites, closely inspecting the surface of the net may lead to inefficient performance as holes may occur infrequently. We leverage a hierarchical task network planner to construct plans on when to evaluate a net closely and when to evaluate a net at a distance in order to survey the net with a wider range. Simulation results are provided.

Index Terms—Omnidirectional Surface Vehicle, Hierarchical Task Network Planning

I. INTRODUCTION

Aquaculture provides for approximately half of all fish consumption and is a fast-growing industry that relies on raising fish inside of cage nets underwater [1], [2]. However, these cages break over time and allow fish to escape, leading to large economic losses and possibly negative impacts in the local marine environment. According to a report from Norweigian aquaculture, cage net failure is the dominant cause of fish loss as approximately two-thirds of reported incidents are a result of cage holes [3]. While manual inspection may be used to identify holes, fish cages are generally deployed far from shore where strong waves and currents may inhibit manual inspection efforts as divers may be endangered [2], [4].

As an alternative, underwater remotely operated vehicles (ROVs) have been recently applied for cage inspection [5], [6] and companies such as Teledyne and DeepTrekker offer commercial ROV solutions [7], [8]. These remote operations, while safe, are time-consuming and expensive as a human operator must continuously control or guide the ROV in potentially large aquaculture sites. Designing the path of the ROV also depends on the operator's experience and makes it unlikely that full coverage of the fish cage is ensured [5], [9], [10].

The research work is supported by ONR grants N00014-19-1-2556 and N00014-19-1-2266; AFOSR grant FA9550-19-1-0283; NSF grants CNS-1828678, S&AS-1849228 and GCR-1934836; NRL grants N00173-17-1-G001 and N00173-19-P-1412; and NOAA grant NA16NOS0120028.

The use of autonomous underwater vehicles (AUVs) is a natural solution to the problems associated with ROVs as AUVs may automatically survey fish cages without extensive human intervention. The authors of [11] designed a small AUV for fish cage net inspection and the authors of [9] provided work on the modeling and control of an AUV for aquaculture net examination. However, AUVs struggle with localization, navigation, and communication as both GPS-based localization and standard radio-based communication are significantly impaired in aqueous environments. The authors of [2] utilize a custom sensor suite and some prior knowledge of the AUV and fishnet to perform localization while the authors of [10] utilized ultra-short baseline (USBL) and Doppler velocity log (DVL) systems to perform localization. However, as noted by the authors, USBL and DVL acoustic systems are disrupted by acoustic interactions with biomass in aquaculture sites, leading to possibly inaccurate localization.

We therefore designed an omnidirectional surface vehicle (OSV) in our previous work that allowed for aquaculture inspection through the use of a retractable underwater camera and deep-learning based vision algorithms [12]. As shown in Fig. 1, the surface-mounted sensors can utilize accurate vehicle pose measurements to perform localization and navigation while doing underwater inspection. In addition, since the OSV electronics are on the surface, standard WiFi connections can be used to allow for easy wireless access. However, in our previous work, we only investigated a small fish net in a pool in order to detect holes. For larger aquaculture sites, it may be useful to design various behaviors in order to more efficiently survey the fish net. We therefore propose the use of AI planning techniques to automatically select when to have our OSV carefully evaluate potential holes up close, and when to have our OSV survey the net quickly at a distance.

AI planning methods are well explored in the literature and have been used to control various robotic platforms. In [13], the authors provide a hierarchical task network (HTN) planning based approach to have a team of robots play soccer, while in [14], the authors provide a modified HTN planner that allows for the inference of human goals in order to improve human-robot collaboration. The authors of [15] propose an extension to planning methods that incorporates decisions on the belief space of an agent so as to ensure that agent actions

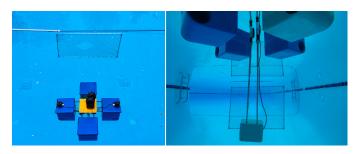


Fig. 1. Photo of the OSV and the onboard camera with adjustable depth [12].

may yield high confidence results. For example, a humanoid robot may be tasked with not only placing a block on a table, but placing a block on a table with high confidence based on sensor measurements.

In the aquatic environment, the authors of [16] design an intelligent fault-tolerant AUV using an HTN planner that is able to continue performing some operations in the event of a non-catastrophic failure, while in [17], the authors present an HTN planning method for AUVs in order to handle the complexity of planning over very large time horizons i.e. days or weeks. We therefore leverage HTN planners to construct mission plans that allow for the OSV to balance between searching for holes and closely inspecting holes where planning actions are directly associated with specific tracking waypoints. In future efforts, the HTN planner can be easily extended to balance other objectives as well, such as regularly recharging the OSV.

This paper is organized as follows. In section II, we reintroduce the OSV platform, and describe its capabilities including its neural-network based fish hole detection algorithm. Then, in section III, we review AI planning problems and the HTN planner and describe our integration of the HTN planner with the OSV. In section IV, we provide a simulation implementation of the AI planning approach and discuss further extensions in section V.

II. OSV PLATFORM OVERVIEW

We develop an omnidirectional surface vehicle (OSV) with the purpose of facilitating efficient aquaculture infrastructure inspection without human intervention [12].

As demonstrated in Fig. 1, the OSV features a symmetric multi-hull overactuated design for omnidirectional maneuverability and damage redundancy in aquaculture sites. All the thrusters of the OSV are protected inside the compartments, which is safer to operate in fish farms. An elevated underwater camera is installed at the center of the OSV to enable capturing pictures of the fish cage at different depths. The convenient features of the OSV significantly reduce the time for building, operating and maintaining the vehicle. The OSV can be easily operated by one person with laptops or smartphones. Moreover, the OSV has strong onboard processing power. A generic x86 onboard computer provides outstanding compatibility for various devices and software, while a discrete neural network accelerator hardware significantly speeds up the image processing for fish cage damage detection. Despite the OSV

is designed for aquaculture inspection, the vehicle is also capable of supporting various tasks. Commercial applications including environmental monitoring, submerged infrastructure inspection, and diver's companion are highly expected. The OSV is also competent for supporting scientific research, such as investigating underwater acoustic communications [18].

We incorporate neural network algorithms [19] to detect holes with different sizes and shapes. Fig. 2 demonstrates a hole on the fishcage can be effectively detected by the neural network. With strong onboard processing power, we allow each hole on the fish cage to be detected multiple times to improve the robustness and accuracy of the inspection. While the OSV is taking pictures of the fishcage and detecting potential holes, a localization device tracks the position and the heading of the OSV at centimeter-level accuracy. By combining the pose of OSV in the fish farm, and the pixel position of the holes on the image plane, the position of the hole can be estimated in the fish farm. A demonstration video on both OSV design and fishcage inspection methodology is available at: https://youtu.be/5XcMtZ4k3rY.

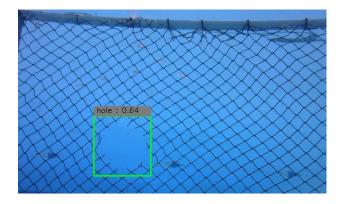


Fig. 2. Demonstration of hole detection with neural network [12], [19].

III. PLANNING FOR LARGE AQUACULTURE SITES

In this section, we detail our efforts to incorporate an HTN planner in order to automatically search for new fishnet holes in a timely manner. We extend our previous effort in [12] and consider larger net scenarios, where the OSV must decide between surveying the net at a closer distance with higher accuracy or at a farther distance with higher coverage.

A. AI Planning

A planning problem is described by a tuple [20] $\Sigma = (s_0, g, S, A, \gamma)$ where s_0 denotes an agent's initial abstract state which is described by a set of boolean-evaluated variables known as predicates, g denotes an agent's desired abstract state or abstract action to take, S denotes the set of all possible abstract states, A denotes a set of all possible abstract actions, and $\gamma: S \times A \to S$ is a transition function that maps the agent between states according to a specified action. Notably, γ defines preconditions and effects which specify when an action is valid from a given state and what state the agent will move to next when the action is taken. The solution to a

planning problem is a plan $\pi = a_1, a_2, \dots, a_n$ where $a_i \in A$ such that the agent, obeying γ , reaches the desired goal state or action q.

For a given problem Σ , various planning methods may be used to search for appropriate plans π in order to achieve the desired behavior. In this work, we use a Hierarchical Task Network (HTN) planner [21] to search for appropriate behaviors. HTN planners are search methods that can take advantage of task hierarchies. Fig. 3 demonstrates a car driving example of how an HTN planner can use various levels of abstracted actions to compile a plan quickly and efficiently.

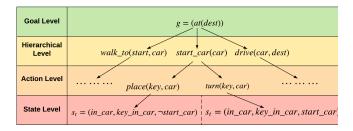


Fig. 3. Example decomposition of HTN for a car driving problem.

In the provided example, an HTN planner is tasked with constructing a sequence of actions in order to get the agent to some location called dest. Here, the user has not only defined the planning problem Σ but has also provided defined collections of actions called methods that represent specific sequences of actions. For example, the method $start\ car(car)$ requires two actions place(key, car)and turn(key, car) which directly transition the state using γ from $s = (in \ car, key \ in \ car, \neg start \ car)$ to s =(in_car, key_in_car, start_car). Methods may also be decomposed into other methods as well and may use logical operators (such as if-else conditions) to decompose in a variety of ways, allowing for potentially many levels of hierarchical depth. Methods also adopt the preconditions and effects of their composite actions, dictating specific orderings to accomplish goals. With the preconditions and effects of methods defined, the planner therefore plans at the state-hierarchical level only using actions when necessary instead of the full state-action level.

In some situations, the environment may pose some uncertainty and the agent must also make decisions based on confidence thresholds. We call such predicates in the state fluents which only evaluate to True if the corresponding confidence interval is achieved [15]. For example, one precondition on applying the action place(key, car) could be that a fluent describing high confidence on the location of the key in the agent's hand evaluates to True.

Given the description of an AI planning problem, our objective can be described as constructing a plan that chooses a sequence of reference signals to apply in order to ensure our agent efficiently surveys a large aquaculture net for holes with high accuracy. By balancing between a hole tracking signal and a circle tracking signal, our AI planner can be used to efficiently search for holes using the neural network algorithm

described previously. We now describe the states, actions, and methods used.

B. Efficient Searching for Aquaculture Holes

For our given OSV, we describe the discrete-time dynamics on the surface of the water as

$$x_{t+1} = x_t + \Delta t u_t \tag{1}$$

where $x_t \in \mathbb{R}^2$ is the 2D planar position of the OSV, $u_t \in \mathbb{R}^2$ is the omnidirectional velocity input applied to the OSV, and Δt is a discrete timestep. We assume that the net of interest is described by a circle with a center at the origin and the OSV has a heading tracking controller enabled continuously that ensures the camera view is always pointed towards the outward normal direction of the net from the perspective of the OSV.

The OSV therefore only needs to decide what 2D linear velocity to apply in order to ensure the on-board camera is best able to survey the fish net. We choose a feedback controller such that (1) becomes

$$x_{t+1} = x_t + \Delta t(k(z_t - x_t)) \tag{2}$$

where k > 0 is a feedback gain and $z_t \in \mathbb{R}^2$ is the desired position set by the planner.

For our planning problem, we associate abstract actions with two z_t signals. The first signal, $z_t = y_i$ (where y_i denotes the position of the detected hole with the lowest confidence) is used for close investigation. The second is associated with the signal

$$z_t = r_{far} \frac{\phi_t}{\|\phi_t\|}, \quad \phi_t = x_t + \alpha R \frac{x_t}{\|x_t\|}$$
 (3)

where R is the 2D rotation matrix associated with the angle $\frac{\pi}{2}$, α is a distance constant that dictates how far ahead of the OSV to generate the next waypoint, and r_{far} is the circle radius to track which is much smaller than the radius of the fish net. This desired position signal is computed by projecting the OSV's current position with a slight offset onto the circle with radius r_{far} .

For the two reference signals described above, we associate two actions move_close and move_far and two corresponding predicates, at close and at far, to indicate whether the OSV is close to the net or far away. In addition, upon detection of a new hole h_i at estimated position y_i , we generate three new fluents which each evaluate according to a range on an uncertainty value s_i associated with h_i . The three fluents are defined as $bLoc(h_i, [0, 1]), bLoc(h_i, (1, 3]),$ $bLoc(h_i,(3,\infty))$ where s_i is a function of the distance between the OSV's position and the estimated hole position: $s_i = min(s_i, 0.05(||x_t - y_i|| - 2)^2)$. We define the transition function γ with the preconditions and effects shown in Table I. Note that each action may only be active when the fishnet sensing from the neural network either registers a hole with low confidence or all holes in the camera view have been appropriately inspected. Finally, we define a method $search(s_t)$ that decomposes according to the current state

TABLE I PRECONDITIONS AND EFFECTS OF TAKING PLANNING ACTIONS

Action	Precondition	Effect
$move_close$	at_far and $\exists i \ s.t.$	at_close and
	$bLoc(h_i,(1,3])$ or	$bLoc(h_i, [0, 1]), \forall i$
	$bLoc(h_i,(3,\infty))$	
$move_far$	at_close and	at_far
	$bLoc(h_i, [0, 1]), \forall i$	

s which is updated at each time-step according to the true observations made in the environment. If s_t contains new hole observations, then $search(s_t)$ decomposes in order to evaluate each observed hole before returning to the surveying distance. At each iteration, our agent first observes the environment and its own position, updating any necessary predicates or fluents as needed. Then, searches for a plan that satisfies $search(s_t)$ and executes the first action of the plan. In future work, we may also incorporate other planning operations that the OSV must decide to switch between. For example, we can design additional predicates and fluents if there are multiple fishnets in an aquaculture site to inspect, or we can modify preconditions on all actions in order to incorporate a need to regularly recharge the OSV.

IV. SIMULATIONS

We demonstrate here a simulation of our OSV platform with simulated viewing capabilities that vary over distance, i.e., our OSV is able to identify the position of fish holes with greater confidence as the OSV gets closer to the hole. We assume that at a distance of 12m from the net (shown as the black dotted circle), the OSV is able to still detect new holes but has lower confidence on the location of the hole (where confidence is shown by a circle that is red for low confidence, yellow for medium confidence, and green for high confidence). In order to improve the confidence on the location of a hole, the OSV must move to a distance of 2m from the net, which is chosen based on our previous experimental results as the optimal distance from the net to detect fish holes. Using the fluents, predicates, actions, and method designed previously, Fig. 4 shows the progression of our simulated OSV as the OSV searches for new fish holes in the circular net.

In the first snapshot, the simulated OSV discovers a new hole at a 12m distance and the HTN planner generates the action $move_close$ since the uncertainty on the newly discovered hole is high. In the second snapshot, the OSV has successfully improved the confidence of the location of the first hole and has begun to closely survey a second discovered hole. Note that while the OSV had begun to move back to surveying distance after inspecting the first hole, the planner generated a new plan when a new hole observation appeared, allowing the OSV to immediately move towards inspecting the new hole. In the third snapshot, the OSV does not discover any holes and therefore chooses to survey the net at the 12m distance, allowing for a faster survey of the net. In the final snapshot, the OSV has discovered all holes and is in the process of examining the final hole.

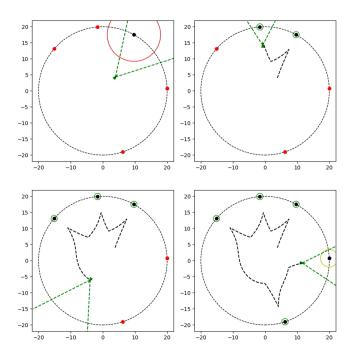


Fig. 4. Simulated OSV search for fish holes in a large aquaculture net. Order of simulation is top-left, top-right, bottom-left, bottom-right. Red dots indicate new holes that have not been discovered, circles around black holes indicate the confidence associated with the location of a discovered hole, green dotted lines indicate the viewing range while the green triangle indicates the position and heading of the OSV.

From here, the OSV may continuously patrol the perimeter of the fishnet in order to check for any new holes. We will pursue in future work allowing the OSV to decide whether to search other fishnet sites nearby or to return and recharge between regular fishnet site inspections. In addition, given fishnets may be repaired without informing the OSV, we may include behavior to consider if an expected hole disappears and allow the OSV to confirm the repair of a fishnet.

V. CONCLUSION AND FUTURE WORK

In this paper, we discussed an extension to our previous paper on using neural network fish hole detection by using an HTN planner to generate plans on when to closely inspect the fish net and when to survey the fish net from a distance. The proposed approach may also incorporate new behaviors easily in the future and can be extended to scenarios where multiple OSVs need to balance surveying multiple fishnets.

In future work, we plan to verify our proposed approach by collecting experimental results and incorporating other behaviors that make the scenario more realistic, such as including a necessary battery constraint or considering the environmental uncertainty (i.e., condition of the weather or tidal flow). In addition, we may consider the effects of repair divers that have one-way communication with the OSV, i.e., they are able to learn of the locations of holes from the OSVs but do not inform the OSV that a repair has occurred. This might occur

if the OSV feeds information back to a central server from which a diver may use to specifically target hole repairs.

REFERENCES

- [1] "The State of World's Fisheries and Aquaculture" 2016. Food and Agriculture Organization of the United Nations.
- [2] S. Potyagaylo, C. C. Constantinou, G. Georgiades, and S. G. Loizou, "Asynchronous ukf-based localization of an underwater robotic vehicle for aquaculture inspection operations," in OCEANS 2015 - MTS/IEEE Washington, Oct 2015, pp. 1–6.
- [3] Ø. Jensen, T. Dempster, E. Thorstad, I. Uglem, and A. Fredheim, "Escapes of fishes from norwegian sea-cage aquaculture: causes, consequences and prevention," Aquaculture Environment Interactions, vol. 1, no. 1, pp. 71–83, 2010.
- [4] S. Potyagaylo and S. G. Loizou, "Online adaptive geometry predictor of aquaculture fish-nets," in 22nd Mediterranean Conference on Control and Automation, June 2014, pp. 1002–1007.
- [5] H. V. Bjelland, M. Føre, P. Lader, D. Kristiansen, I. M. Holmen, A. Fredheim, E. I. Grøtli, D. E. Fathi, F. Oppedal, I. B. Utne, and I. Schjølberg, "Exposed aquaculture in norway," in OCEANS 2015 -MTS/IEEE Washington, Oct 2015, pp. 1–10.
- [6] R. Capocci, G. Dooly, E. Omerdi'c, J. Coleman, T. Newe, and D. Toal, "Inspection-class remotely operated vehicles - a review," Journal of Marine Science and Engineering, vol. 5, no. 1, 2017.
- [7] Solutions for Offshore Aquaculture, Teledyne Marine, Market Brochure, 2016.
- [8] "Aquaculture, inspect interior and predator nets, moorings, perform feeding assessments, and quickly retrieve morts," https://www.deeptrekker.com/aquaculture/, accessed: 2018-08-15.
- [9] M. N. Haval, "Modelling and control of underwater inspection vehicle for aquaculture sites," Master's thesis, Norwegian University of Science and Technology, 2012.
- [10] P. Rundtop and K. Frank, "Experimental evaluation of hydroacoustic instruments for rov navigation along aquaculture net pens," Aquacultural Engineering, vol. 74, pp. 143 – 156, 2016.
- [11] V. Chalkiadakis, N. Papandroulakis, G. Livanos, K. Moirogiorgou, G. Giakos, and M. Zervakis, "Designing a small-sized autonomous underwater vehicle architecture for regular periodic fish-cage net inspection," in 2017 IEEE International Conference on Imaging Systems and Techniques (IST), Oct 2017, pp. 1–6.
- [12] Q. Tao, K. Huang, C. Qin, B. Guo, R. Lam and F. Zhang, "Omnidirectional Surface Vehicle for Fish Cage Inspection," OCEANS 2018 MTS/IEEE Charleston, Charleston, SC, 2018, pp. 1-6
- [13] Obst, Oliver, and Joschka Boedecker. "Flexible coordination of multiagent team behavior using HTN planning." Robot Soccer World Cup. Springer, Berlin, Heidelberg, 2005.
- [14] Hayes, Bradley, and Brian Scassellati. "Autonomously constructing hierarchical task networks for planning and human-robot collaboration." 2016 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2016.
- [15] Kaelbling, Leslie Pack, and Tomás Lozano-Pérez. "Integrated task and motion planning in belief space." The International Journal of Robotics Research 32.9-10 (2013): 1194-1227.
- [16] Patrón, Pedro, et al. "Semantic knowledge-based representation for improving situation awareness in service oriented agents of autonomous underwater vehicles." OCEANS 2008. IEEE, 2008.
- [17] Buksz, Dorian, et al. "Strategic-tactical planning for autonomous underwater vehicles over long horizons." 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 2018.
- [18] Q. Tao, J. Lobley, Y. Yu, Y. M. Aung, Y.Zhou, F. Tong, A. Song and F. Zhang, "Omnidirectional Surface Vehicle for Evaluating Underwater Acoustic Communication Performance in Confined Space," 2019 International Conference on Underwater Networks & Systems (WUWNET'19).
- [19] J. Redmon, "Yolo: Real-time object detection." [Online]. Available: https://pjreddie.com/darknet/yolov1/
- [20] Norvig, P. R., & Intelligence, S. A. (2002). A modern approach. Prentice Hall.
- [21] Nau, D., et al. "SHOP: Simple hierarchical ordered planner." Proceedings of the 16th international joint conference on Artificial intelligence-Volume 2. 1999.