Model-Free Reinforcement Learning for Static Point Source Localization in a 3D Simulation.

Connor Tate
Intelligent Systems and Robotics
University of West Florida
Pensacola, USA
cct21@students.uwf.edu

Kevin Francis
Intelligent Systems and Robotics
University of West Florida
Huntsville, USA
kjf13@students.uwf.edu

Timothy L.J. Stewart
Intelligent Systems and Robotics
University of West Florida
Pensacola, USA
tls54@students.uwf.edu

***Abstract*—The development of effective source localization techniques has broad reaching implications across multiple domains and environments. Whether it be natural resource detection, environmental monitoring and conservation, search and rescue, navigation, mine countermeasures or endless additional applications it is an interesting and complex problem that can be solved by the implementation of machine learning algorithms. In this paper, we show that an artificial intelligent agent trained with Reinforcement Learning in a Q-learning algorithm can reliably locate a source of pollution in deterministic and stochastic environments.**

***Index Terms*—reinforcement learning, q-learning, source localization, model and simulation, pollution**

I. Introduction

Source localization is a technical challenge spanning multiple domains from military mine counter measures, commercial offshore oil detection and environmental monitoring of point source pollution. Traditional methods for detecting and localizing potential contaminants require extensive administrative oversight, time consuming laboratory analyses and are dependent on the time and spatial distribution of field teams. The environmental problems that field teams face in the ocean are dynamic, complex and time sensitive requiring rapid response and are often beyond the reach of human teams. In circumstances such as oil leaks, chemical plumes or harmful algal blooms a person may perceive a problem at the sea surface which originates miles below. These circumstances often call for quick action and ability to reach extreme environments that may not be accessible to field teams but are accessible to autonomous agents. For this reason autonomous underwater vehicles (AUVs) have been developed to extend field sensing capabilities.

The vehicles used today are "autonomous" in the sense they can be set with a mission and sent out, however, they are not intelligent [1]. When faced with a problem that does not have the goal state associated with coordinates, many systems fall back on basic lawnmower, yo-yo or spiral patterns in order to cover a broad space within the region of interest. These brute force methods are effective for mapping a space, however when detecting and locating a threat is time sensitive, the ability to set out with a mission and adapt to sensory information is essential. Oftentimes AUVs are equipped with sensor suites that allow for high resolution data throughput and post analysis to tackle an array of monitoring mission objectives. Many of these sensors can be directly linked into the control board to provide rich sensory information such as underwater mass spectroscopy (UMS), dissolved oxygen (DO2) fluorescence, backscatter and other optical measurements. By utilizing these sensor suites the AUV has increased capacity to perceive its environment and learn about the conditions in a framework that is related to the mission objective.

In this paper we will apply a Reinforcement Learning approach using a Q-learning algorithm to teach a single agent how to locate a source based on sensory inputs from the marine environment. We will simulate this agent in a 3 dimensional space in which pollution is distributed as a diffuse gradient representative of a plume as well as a stochastic distribution similar to the distribution pattern of an oil spill. The objective is to train the agent to detect pollution levels and learn to guide itself to the source based on these percepts. This paper will be organized in the following manner. Section 1 which you have already read will introduce the problem. Section 2 we will discuss the background of source localization, the various approaches, related works and a brief overview of Q-learning. Section 3 we will review methods for developing the simulation environment and the configuration of the agent goal and learning parameters as well as the experiment

II. Related Works

A. Source Localization

Historically the problem of source localization has been tackled from stationary sensing nodes, physical sample collection by field teams, towed sensor arrays behind ships or more recently with the use of autonomous vehicles(AV). Autonomous vehicles provide the most flexibility of all the detection and localization modalities due to their ability to reach remote and extreme environments as well as perform long term deployments ranging from weeks to a year depending on the vehicle architecture. Whereas the other methods for detecting and localizing require extensive manpower, resources and time succumbing to spatial and temporal limitations. The variety of vehicles available for autonomous monitoring and sensing range from gliders, surface vehicles, lagrangian drifters and

Source code at https://github.com/tljstewart/openai-rl-findit/tree/main torpedo shaped submersibles; this array of platforms enables multiple approaches toward solving the source localization problem. However current deployment of these vehicles is constrained by the inefficient predefined trajectories and limitations in operator-vehicle communications that prevent trajectory modification [2]. In the marine environment communication is bandwidth limited and in some instances such as localization for mine counter measures, communication and human intervention for trajectory modification is restricted therefore a need for increased autonomy and higher levels of system intelligence are essential [3]. The importance of expanding the functionality and applicability of these vehicles in low communication environments with varying levels of uncertainty has been identified as a key research area by multiple institutions including the National Defense Research Institution, Academic research groups and environmental research institutions such as Woods Hole Oceanographic Institution.

B. Approaches

In the past 20 years research in the area of autonomous source localization and vehicle control by means of chemical profiling and on board sensors has increased significantly. Evidence supporting onboard chemical profiling for plume tracing and source localization can be found in many field studies including the work of [4]–[8]. However, of the field demonstrations, many are trajectory limited to the traditional lawnmower and yo-yo patterns designed to cover a large area. Although effective for mapping large swaths these methods of trajectory planning are inefficient when considering the precisely localized problem of source localization [2]. Recent work integrating on board sampling for adaptive trajectory control focuses on one of three areas: targeted features of interest (TFOI), objectives of sampling mission (OSM), and multi-vehicle networking. Some related efforts involve adaptive sampling techniques which include informative path planning strategies such as those employed by [9], [10] integrating sensory information with the vehicles working environment model to update the trajectory. One such approach is that of [2] in which they use real-time information-seeking algorithms to optimize source localization tasks by having the agent autonomously decide sampling locations. Of the works reviewed [2] was the only paper that showed promising simulation results which translated to actual field performance.

Outside of limited field demonstrations, much of the current research is simulated with the majority providing simulations in a 2D grid [7], [11]–[13] [14]. This limitation is typically due to computational constraints and the increase in complexity and computational demands of simulating an environment in 3D.

In the works reviewed methods for source source localization include: supervised learning methods [15], chemotaxis and Fisher Information Matrix [11], and a range of reinforcement learning approaches (RL) [7], [12], [13], [16] [14]

In [15] Source Localization in an Ocean Wave-guide Using Supervised Machine Learning they compared Support Vector Machines (SVM), Feed-forward Neural Networks and Random Forests have been explored for the solving of acoustic source localization [15]. However the attenuation behavior of acoustics and chemical contaminants in water are incomparable and the state space is exponentially larger for chemical dispersants making supervised learning methods an inefficient approach to the localization of contaminants.

[11] work in optimal search strategies developed a probabilistic representation that adaptively account for source location in uncertainty using the Fisher Information Matrix and compared their results to systems using chemotaxis and information taxis [11]. But their simulations were limited to 2D and results indicated the method was subject to pitfalls of local minima and maxima.

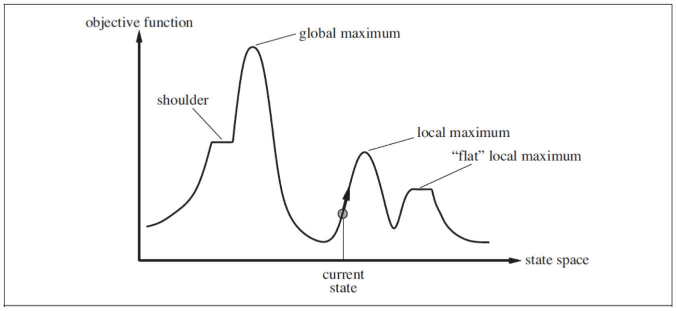

Fig. 1. Global, local maxima, minima and shoulders that can stump an agent.

A number of RL methods have been proposed [12], [13]. [7], [14], [16] Of the papers reviewed three contributed to overall vehicle control and spatial navigation related to matters of depth control, obstacle avoidance and propulsion. As it is related to this work, the papers by Weidemann and Wang implemented and supported the use of RL for the subject of source localization. [17] used a model based method which is dependent on a priori knowledge of the environment and high fidelity of the model.

C. Reinforcement Learning Approach

As it stands, there are limited applications of RL in the domain of source localization and of those that do employ an RL method many provide only 2D simulations. Those who have researched 3D simulation or field work rely on model-based learning which are subject to high fidelity modeling during agent training, which is notoriously difficult due to the complexity of the environment. Additionally, the constraints introduced by vehicle communications and the proven need for increased autonomy means that online mission reconfiguration is limited. [17] [3].

III. Methods

A python interface was implemented to simulate an agent and a three-dimensional environment. A uniform concentration gradient in a deterministic environment was applied around a source point which the agents goal was to locate, later a stochastic pollutant environment was applied based on a pseudo-random gaussian distribution for pollution concentrations. The agent was trained using reinforcement learning, with a Q-Learning Algorithm.

A. Q-Learning

Q-Learning is a Reinforcement Learning (RL) algorithm based on the Q-Value Iteration Algorithm first developed by Richard Bellman in the form of the Bellman Optimality Equation, that utilizes a Markov Decision Process which are directed graphs to find an optimal reward defined by a state and action probabilities one may take in that particular state to acquire an associated reward with an agents decision. Q-Learning is said to be an off-policy algorithm as the policy that is being trained is not the final policy being utilized.

$$\underbrace{\text{New}Q(s,a)}_{\text{New Q-Value}} = \underbrace{Q(s,a)}_{\text{Current Q-Value}} + \underset{\text{Learning Rate}}{\alpha}[\underbrace{R(s,a)}_{\text{Reward}} + \underset{\text{Discount rate}}{\gamma}\overbrace{\max Q'(s',a')}^{\substack{\text{Maximum predicted reward, given}\\ \text{new state and all possible actions}}} - Q(s,a)] \tag{1}$$

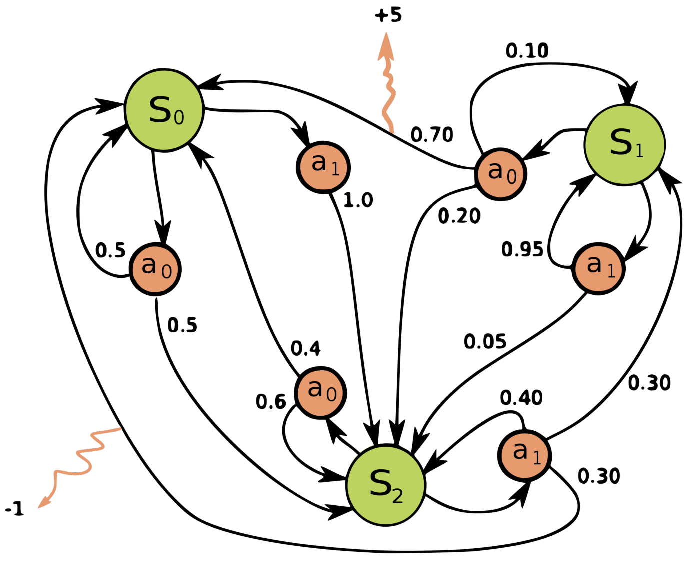

Fig. 2. Simple MDP with three states (green circles) and two actions (orange circles), with two rewards (orange arrows).

B. $\epsilon - Greedy$

One can invoke an exploration behavior for the RL agent by providing pure randomness to the algorithms decision, however we implement a $\epsilon - Greedy$ policy where varies between 0 and 1. The closer to 1 epsilon is the greater the exploration of the RL agent, while the closer to 0 the greater the exploitation of RL agent with prefer which leads to always selecting the maximum learned reward at that state. However, one must be aware that the maximum learned reward may not be a global maximum, that is why initial training should start with a high exploration to discover global maximum and gradually reduce epsilon to exploit this learned maximums.

C. Learning Rate

The *learning rate* is a step size that determines the extent newly acquired information overrides old information. A factor of 0 makes the agent learn nothing forcing the agent to exclusively exploit prior learned knowledge, while a factor of 1 makes the agent consider only the most recent information, ignoring prior knowledge to explore and hopefully to discover globally maximum rewards. In fully deterministic environments, a learning rate of $\alpha_t = 1$ is optimal, and while the problem is stochastic and variable learning rate may be useful. In practice, often a constant learning rate is used, such as $\alpha_t = 0.1$ for all t.

D. Discount Factor

The *discount factor* γ is a parameter value between 0 and 1 that determines the importance of future rewards. A factor of 0 will make the agent "greedy" by only considering current rewardsr_t while a factor approaching 1 will make it strive for more long-term high rewards. γ , starting with a lower discount factor and increasing it towards its final value accelerates learning.

E. Objective

Demonstrated methods of source localization such as local search and optimization problems guided by chemotaxis are not efficient for large search environments or remote isolated sources due to their affinity for local minima and maxima. The contribution of this work aims to implement an RL method that balances the trade-off between exploration and exploitation to efficiently locate and navigate toward the "source" in a relatively large static environment

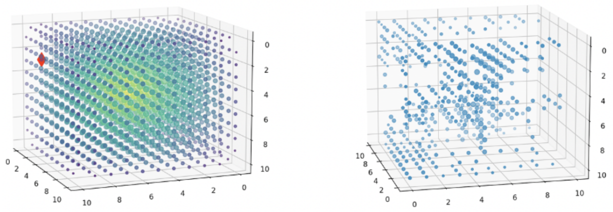

Fig. 3. Left shows the agent moving in the environment and sensing the states. Right shows the 3D representation of the policy containing the Q-values associated to the location within the environment.

F. Environmental Sensors and Percepts

The agent's mission is to find a pollution source, this task lends itself to multiple potential sensory inputs ranging from pH, ctd, fluorometry, optical backscatter and dissolved oxygen sensors. In earlier renditions the agent receives information regarding pollution level as a non specific measure. This is left intentionally vague as 'pollution' can refer to anything from anthropogenic runoff to oil spills and beyond. However, many types of marine pollution influence the level of dissolved oxygen available in the water. This is due to pollution's impact on the microbial community, in particular phytoplankton. These organisms are responsible for oxygen production in the photic zone and can be choked out by oil spills blocking sunlight or overbloom and die off in harmful algae blooms. Either effect results in a net loss of dissolved oxygen in the surface layer of the ocean. For this reason dissolved oxygen (DO2) sensor input is provided to the agent in the form of an immutable tuple correlated to a location and a pollution level at that location.

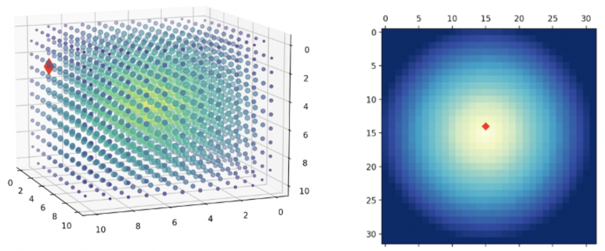

Fig. 4. Left 3D and Right aerial representation of the environment in which the pollution is diffused over a gradient with the Goal state being located in the center of the space. The goal is the state with 100% pollution which is represented in bright yellow. As the pollution level decreases the "water" is visualized with a bluer coloration.

Q-learning was employed to allow the agent the ability to explore and build upon its knowledge of the surrounding environment through "experiences". Experiences were set as a tuple of three ((coordinates), DO2, pollution) and were appended for every action the agent took. The quality of moving to a new location was estimated based on the reward for being at the current location + the utility of all of its successor states. This was calculated by iterating over a list of possible moves the agent could make and appending a utility table the agent kept track of in an internal data structure. For the 1st trial the agent would return mostly zeros until it reached the goal, this is because utility is a function of reward and discount factor which is representative of preference over immediate vs distant reward. Once the source location was found the surrounding locations could be appended in the agent's memory and become more precise to their true quality with every iteration. The RL agent was trained on a 12x12x12 search space where each state had a reward value of 0, while the source state reward was set to 100. γ was set to 0.9 in order to encourage the agent to reach the goal rather than accumulate reward during exploration.

Exploration was an important component of the agent's utility memory and learning determining whether or not the agent experienced all the states in the space or preferred taking a direct path. This direct path seems like the obvious approach, however, if the agent does not spend enough time exploring the utility estimates are inaccurate making it difficult to navigate efficiently. To encourage exploration random actions were chosen in the beginning with a rate of decay ϵ of 0.999. Decay rate determines how long the agent prioritizes random movement over exploiting learned utility policy. Because the state space was so large a slow decay allowed the agent to explore more of the space in the beginning. Exploitation occurs once utility exceeds decay, at which point the agent prioritizes its following actions based on the utilities it has learned along the way.

Algorithm 1: Q-Learning

```
States $\mathcal{X} = \{1, \dots, n_x\}$
Actions $\mathcal{A} = \{1, \dots, n_a\}$,    $A : \mathcal{X} \Rightarrow \mathcal{A}$
Reward function $R : \mathcal{X} \times \mathcal{A} \to \mathbb{R}$
Black-box (probabilistic) transition function
$T : \mathcal{X} \times \mathcal{A} \to \mathcal{X}$
Learning rate $\alpha \in [0, 1]$, typically $\alpha = 0.1$
Discounting factor $\gamma \in [0, 1]$
Procedure $QLearning(\mathcal{X}, A, R, T, \alpha, \gamma)$
Initialize: $Q : \mathcal{X} \times \mathcal{A} \to \mathbb{R}$ arbitrarily
while $Q$ is not converged do
    Start in state $s \in \mathcal{X}$
    while $s$ is not terminal do
        Calculate $\pi$ according to $Q$ and exploration
        strategy (e.g. $\pi(x) \leftarrow \arg\max_a Q(x, a)$)
        $a \leftarrow \pi(s)$
        $r \leftarrow R(s, a)$ ;    ▷ Receive the reward
        $s' \leftarrow T(s, a)$ ;       ▷ Receive the new
         state
        $Q(s', a) \leftarrow$
        $(1 - \alpha) \cdot Q(s, a) + \alpha \cdot (r + \gamma \cdot \max_{a'} Q(s', a'))$
        $s \leftarrow s'$
    return $Q$
```

IV. Experiments and Results

A. Environment

The marine environment is dynamic, but for the purposes of this development the worlds generated will remain static and will vary between deterministic and stochastic. This is relevant when sampling for pollution as there are multiple different pollution types: chemical plumes, oil spills, harmful algal blooms (HABs) and anthropogenic runoff. These pollution types can distribute differently in the environment especially given changing environmental conditions such as sea states, wind, currents, solar radiation and multiple physical and chemical factors. To simplify for the purposes of this environmental model we categorize based on how stochastic or deterministic each type of pollution typically disperses.

Both chemical plumes and anthropogenic runoff disperse in a more predictable manner (without accounting for hydrodynamic conditions) diffusing the further away from the source, for this reason they are categorized as deterministic. In contrast oil spills and harmful algal blooms appear in patches interspersed with open relatively 'clean' water, or in other words with a more stochastic distribution. The level of determinism and stochasticism was controlled by a gradient and noise function in the environment generation. The equation for calculating pollution is as follows and is modified with noise amplitude if stochastic.

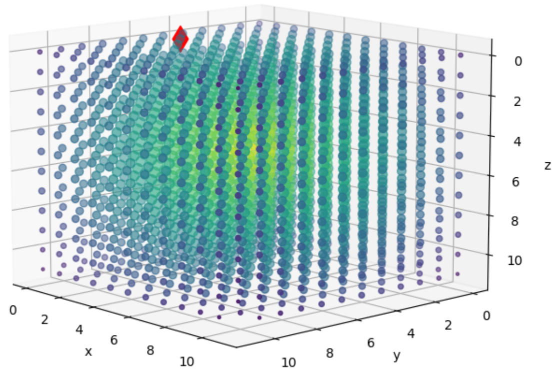

Fig. 5. Deterministic pollution environment in which the pollution is diffused over a gradient with the goal state being located in the center of the space. The goal is the state with 100 percent pollution which is represented in bright yellow.

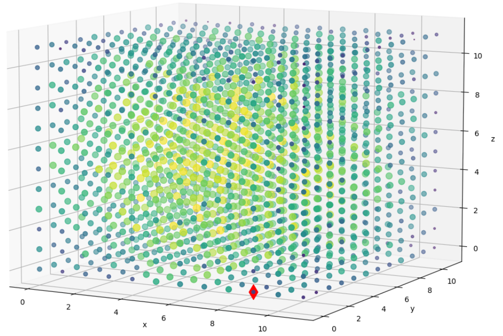

Fig. 6. Stochastic pollution environment in which the pollution is diffused over a gradient with the goal state being located in the center of the space. The goal is the state with 100 percent pollution which is represented in bright yellow.

B. Results

1) Deterministic: The first environment the model was tested against was a deterministic one. That is, little noise or distortion within the data. This was done with an epsilon value of 0.95, gamma value of 0.9, and an alpha of 0.5. In figure 7 the training results can be seen. Around the 500th iteration the q-values started to provide an optimal path to the source at which point the average steps per run to achieve the goal begins to decline rapidly. Figure 8 represents how the trained model performed on a source at the same location as the training data. It can be seen that the agent has learned from the training and is much more efficient when using the generated q-values.

```
Algorithm 2: Environment Generation
  for $dimensions = x, y, z,$ do
  begin
      $\triangle x = source_x - x$
      $\triangle y = source_y - y$
      $\triangle z = source_z - z$
      $distance = \sqrt{\triangle x^2 + \triangle y^2 + \triangle z^2}$
      $pollutionat(x, y, z) = 1 - distance$
  end
```

Moving the source changed how well the model performed, that can be seen in figure 9. This is because the training is specific to a certain source location and not adaptive. This topic will be addressed in future work.

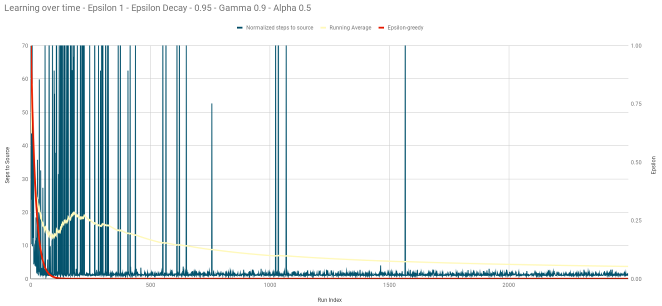

Fig. 7. Training results from deterministic pollution environment

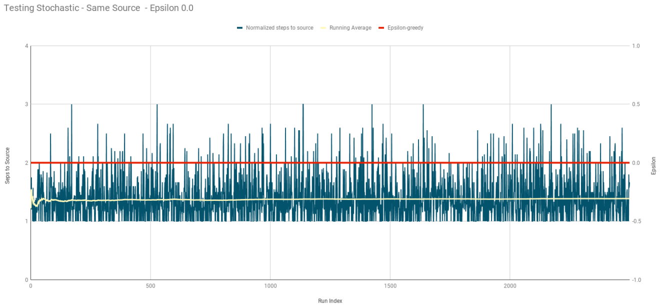

Fig. 8. Testing results from deterministic pollution environment with the same source location as training

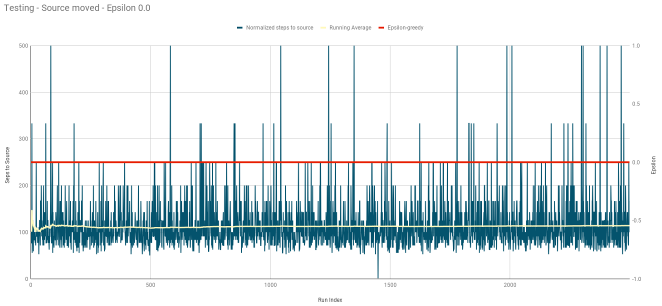

Fig. 9. Testing results from deterministic pollution environment with a source location that is different from training

2) Stochastic: Noise has been added to the environment generation to make the pollution distribution more stochastic. The agent was then trained in this new environment and then tested using the same source location as training. The training results can be seen in figure 10 and testing in figure 11. Notice that the running average is not too different between training and testing. This implies that the learned q-values are no better than exploration in this environment. This is also a problem with will discussed in future work.

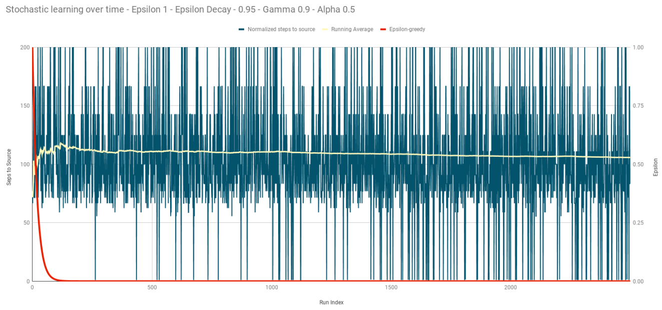

Fig. 10. Training results from stochastic pollution environment

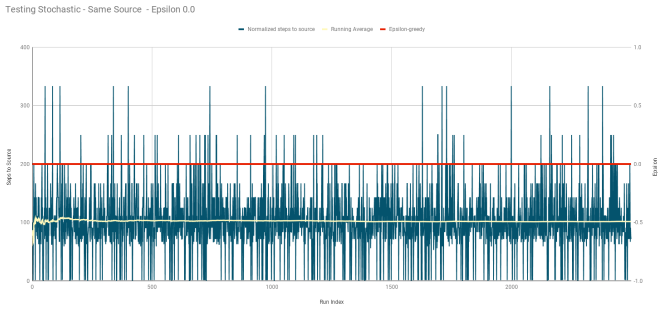

Fig. 11. Testing results from stochastic pollution environment with the same source location as training

V. Conclusion

Traditional environmental monitoring methods for assessing ocean health or tracking ocean pollution require extensive administrative oversight, time consuming laboratory analyses and are dependent on the time and spatial distribution of field teams. The environmental problems that field teams face in the ocean are dynamic, complex and time sensitive requiring rapid response. In this paper we have shown how reinforcement learning can be applied to an autonomous underwater agent using Q-learning to not only seek out the source of pollutant but identify the optimal path to the source.

VI. Future Work

Applying a Deep Q-Learning algorithm which utilizes a Deep Neural Network to estimate the Q-values opposed to using the Temporal Difference equation to find the optimal Q-values may result in higher fidelity action decision making based on the sensory input, resulting in actions with more accurate associated Q-values. Another area of interest is investigating the generality of Reinforcement learning, and maximizing this transfer to similar source location task, for instance, changing the pollutant to CO2, oil leakage, methane or other chemical signatures that can be identified via UMS and have higher spatial variability. Furthermore, to more accurately reflect water currents in an underwater environment the distribution of discretized pollutant nodes should reflect a position and velocity vector on each time step as to resemble a dynamic environment, and investigating these dynamic effects on reinforcement learning for autonomous source localization.

Addressing the noticed issues in the results provides an additional path for advancement. Pursuing Deep Q-Learning may correct the noted issues along the way. However another option is to look at a form of adaptive learning with testing. That is, to test the model, but if needed slightly adjust the q-values to account for a moved source. The distortion in the environment may require other techniques be applied when calculating the q-values to account for the non-uniformity.

References

[1] Jimin Hwang, Neil Bose, and Shuangshuang Fan. Auv adaptive sampling methods: A review, 8 2019.

[2] Paul Stankiewicz, Yew T. Tan, and Marin Kobilarov. Adaptive sampling with an autonomous underwater vehicle in static marine environments. *Journal of Field Robotics*, 2020.

[3] Robert W Button, John Kamp, Thomas B Curtin, and James Dryden. A survey of missions for unmanned undersea vehicles. Technical report, RAND NATIONAL DEFENSE RESEARCH INST SANTA MONICA CA, 2009.

[4] D P Fries, R T Short, L L Langebrake, J T Patten, M L Kerr, G Kibelka, D C Burwell, and J C Jalbert. In-water field analytical technology: Underwater mass spectrometry, mobile robots, and remote intelligence for wide and local area chemical profiling, 2001.

[5] R T Short, D P Fries, M L Kerr, C E Lembke, S K Toler, P G Wenner, and R H Byrne. Focus: Field-portable and miniature ms underwater mass spectrometers for in situ chemical analysis of the hydrosphere, 2001.

[6] R Camilli, B Bingham, M Jakuba, and H Singh. Integrating in-situ chemical sampling with auv control systems, 2004.

[7] Lingxiao Wang, Shuo Pang, and Jinlong Li. Olfactory-based navigation via model-based reinforcement learning and fuzzy inference methods. *IEEE Transactions on Fuzzy Systems*, pages 1–1, 7 2020.

[8] Kanae Komaki, Mayumi Hatta, Kei Okamura, and Takuroh Noguchi. Development and application of chemical sensors mounting on underwater vehicles to detect hydrothermal plumes. Institute of Electrical and Electronics Engineers Inc., 5 2015.

[9] Kian Hsiang Low, John Dolan, and Pradeep Khosla. Information-theoretic approach to efficient adaptive path planning for mobile robotic environmental sensing. In *Proceedings of the International conference on automated planning and scheduling*, volume 19, 2009.

[10] Muhammad F Mysorewala, Dan O Popa, and Frank L Lewis. Multi-scale adaptive sampling with mobile agents for mapping of forest fires. *Journal of Intelligent and Robotic Systems*, 54(4):535–565, 2009.

[11] Behzad Bayat, Naveena Crasta, Howard Li, and Auke Ijspeert. Optimal search strategies for pollutant source localization*.

[12] Chris Gaskett, David Wettergreen, and Alexander Zelinsky. Reinforcement learning applied to the control of an autonomous underwater vehicle, 1999.

[13] Xing Wu, Haolei Chen, Changgu Chen, Mingyu Zhong, Shaorong Xie, Yike Guo, and Hamido Fujita. The autonomous navigation and obstacle avoidance for usvs with anoa deep reinforcement learning method. *Knowledge-Based Systems*, 196, 5 2020.

[14] Thomas Wiedemann, Cosmin Vlaicu, Josip Josifovski, and Alberto Viseras. Robotic information gathering with reinforcement learning assisted by domain knowledge: An application to gas source localization. *IEEE Access*, 9:13159–13172, 2021.

[15] Haiqiang Niu, Emma Reeves, and Peter Gerstoft. Source localization in an ocean waveguide using supervised machine learning, 1 2017.

[16] Hui Wu, Shiji Song, Keyou You, and Cheng Wu. Depth control of model-free auvs via reinforcement learning. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 49:2499–2510, 12 2019.

[17] Yannick Allard and Elisa Shahbazian. Unmanned underwater vehicle (uuv) information study, 2014.