Taming an Autonomous Surface Vehicle for Path Following and Collision Avoidance Using Deep Reinforcement Learning

Tatukuluri Manisha[1*] and Gadi Nirmala[2]

[1*] *M.Tech Student, Department of CSE, Sir C R Reddy College of Engineering, Eluru.*
[2] *Associate Professor, Department of CSE, Sir C R Reddy College of Engineering, Eluru.*
[1*] tmanisha396@gmail.com and [2] nirmala.gadi@gmail.com

Abstract— In this article, we explore the feasibility of applying proximal policy optimization, a state-of-the-art deep reinforcement learning algorithm for continuous control tasks, on the dual-objective problem of controlling an under actuated autonomous surface vehicle to follow an a priori known path while avoiding collisions with non-moving obstacles along the way. The AI agent, which is equipped with multiple rangefinder sensors for obstacle detection, is trained and evaluated in a challenging, stochastically generated simulation environment based on the OpenAI gym Python toolkit. Notably, the agent is provided with real-time insight into its own reward function, allowing it to dynamically adapt its guidance strategy. Depending on its strategy, which ranges from radical path-adherence to radical obstacle avoidance, the trained agent achieves an episodic success rate close to 100%.

Keywords: Collision, Reinforcement, Python, Transportation.

I. INTRODUCTION

Autonomy offers surface vehicles the opportunity to improve the efficiency of transportation while still cutting down on greenhouse emissions. However, for safe and reliable autonomous surface vehicles (ASV), effective path planning is a pre-requisite which should cater to the two important tasks of path following and collision avoidance (COLAV). In the literature, a distinction is typically made between reactive and deliberate COLAV methods. In short, reactive approaches, most notably artificial potential field method, dynamic window methods, velocity obstacle methods and optimal control-based methods, base their guidance decisions on sensor readings from the local environment, whereas deliberate methods, among them popular graph-search algorithms such as A* and Voronoi graphs as well as randomized approaches such as rapidly-exploring random tree and probabilistic roadmap, exploit a priori known characteristics of the global environment in order to construct an optimal path in advance, which is to be followed using a low-level steering controller. By utilizing more data than just the current perception of the local neighborhood surrounding the agent, deliberate methods are generally more likely to converge to the intended goal, and less likely to suggest guidance strategies leading to dead ends, which is frequently observed with reactive methods due to local minima. However, in the case where the environment is not perfectly known, as a result of either incomplete or uncertain mapping data or due to the environment having dynamic features, purely deliberate methods often fall short. The block diagram for Autonomous boat driving system is shown in figure 1.

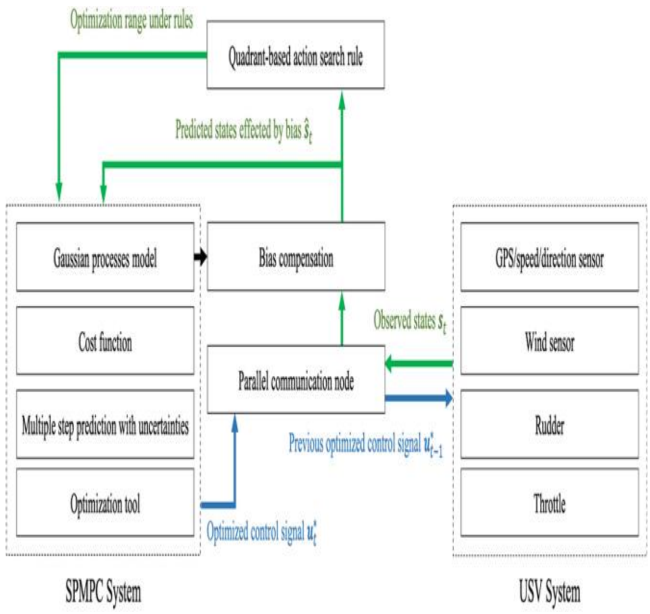

Figure.1: Example diagram for Autonomous boat driving system.

II. RELATED WORK

2.1 The vector field histogram-fast obstacle avoidance for mobile robots:

A new real-time obstacle avoidance method for mobile robots has been developed and implemented. This method, named the vector field histogram (VFH), permits the detection of unknown obstacles and avoids collisions while simultaneously steering the mobile robot toward the target. The VFH method uses a two-dimensional Cartesian histogram grid as a world model. This world model is updated continuously with range data sampled by on-board range sensors. The VFH method subsequently employs a two-stage data reduction process in order to compute the desired control commands for the vehicle. In the first stage the histogram grid is reduced to a one-dimensional polar histogram that is constructed around the robot's momentary location. Each sector in the polar histogram contains a value representing the polar obstacle density in that direction. In the second stage, the algorithm selects the most suitable sector from among all polar histogram sectors with a low polar obstacle density, and the steering of the robot is aligned with that direction. Experimental results from a mobile robot traversing densely cluttered obstacle courses in smooth and continuous motion and at an average speed of 0.6-0.7 m/s demonstrate the power of the VFH method.

2.2 Motion planning and collision avoidance using navigation vector fields:

This paper presents a novel method on the motion and path planning for unicycle robots in environments with static circular obstacles. The method employs a family of 2-dimensional analytic vector fields, which have singular points of high-order type and whose integral curves exhibit various patterns depending on the value of a parameter λ. More specifically, for a known value of λ the vector field has a unique singular point of dipole type and its integral curves are suitable for steering the unicycle to a goal configuration. Furthermore, for the value of λ that the vector field has a continuum of singular points, the integral curves can be used to define flows around circular obstacles. An almost global feedback motion plan is then constructed by suitably blending attractive and repulsive vector fields in a static obstacle environment. The proposed motion planning and control design is also extended to the multi-agent case, where each agent needs to converge to a desired configuration while avoiding collisions with other agents. The efficacy of the approach is demonstrated via simulation results.

2.3 A modified dynamic window algorithm for horizontal collision avoidance for AUVs:

Much research has been done on the subject of collision avoidance (COLAV). However, few results are presented that consider vehicles with second-order nonholonomic constraints, such as autonomous underwater vehicles (AUVs). This paper considers the dynamic window (DW) algorithm for reactive horizontal COLAV for AUVs, and uses the HUGIN 1000 AUV in a case study. The DW algorithm is originally developed for vehicles with first-order nonholonomic constraints and is hence not directly applicable for AUVs without resulting in degraded performance. This paper suggests further developments of the DW algorithm to make it better suited for use with AUVs. In particular, a new method for predicting AUV trajectories using a linear approximation which accounts for second-order nonholonomic constraints is developed. The new prediction method, together with a modified search space, reduces the mean square prediction error to about one percent of the original algorithm. The performance and robustness of the modified DW algorithm is evaluated through simulations using a nonlinear model of the HUGIN 1000 AUV.

III. FRAMEWORK

The focus of this paper is to explore how RL, given the recent advances in the field, can be applied to the guidance and control of ASV. Specifically, we look at the dual objectives of achieving the ability to follow a path constructed from a priori known way-points, while avoiding collision with obstacles along the way. In an end-to-end fashion, control signals for a simulated vessel are generated by a RL agent which, based on the readings from a rangefinder sensor suite which is attached to the vessel as well as rewards received from the environment, learns how to intelligently control the vessel in challenging obstacle avoidance scenarios. The resulting interplay between the environment, which incorporates the dynamics of the vessel itself, and the autonomous RL agent is illustrated in figure 2. For simplicity, we limit the scope of this work to non-moving obstacles of circular shapes. As RL methods are, model-free approaches, by their very nature, a positive result can bring significant value to the robotics and autonomous system field, where implementing a guidance system typically requires knowledge of the vessel dynamics, in the form of non-linear first-principle models with parameters that can only be determined experimentally at great cost.

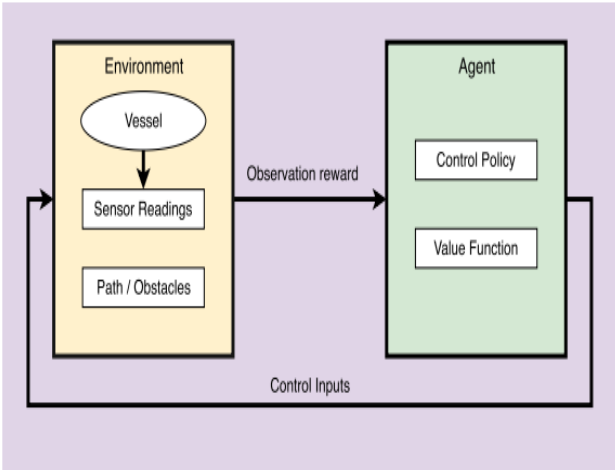

Figure.2: System model

3.1 Reinforcement learning Algorithm:

Reinforcement learning is an area of Machine Learning. It is about taking suitable action to maximize reward in a particular situation. It is employed by various software and machines to find the best possible behavior or path it should take in a specific situation. Reinforcement learning differs from the supervised learning in a way that in supervised learning the training data has the answer key with it so the model is trained with the correct answer itself whereas in reinforcement learning, there is no answer but the reinforcement agent decides what to do to perform the given task. In the absence of a training dataset, it is bound to learn from its experience.

Example: The problem is as follows: We have an agent and a reward, with many hurdles in between. The agent is supposed to find the best possible path to reach the reward. The following problem explains the problem more easily.

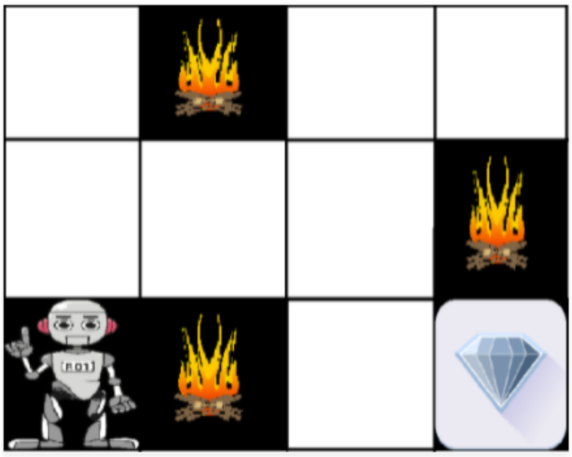

Figure.3: Reinforcement algorithm working image

The figure 3 shows the robot, diamond, and fire. The goal of the robot is to get the reward that is the diamond and avoid the hurdles that are fire. The robot learns by trying all the possible paths and then choosing the path which gives him the reward with the least hurdles. Each right step will give the robot a reward and each wrong step will subtract the reward of the robot. The total reward will be calculated when it reaches the final reward that is the diamond.

IV. EXPERIMENTAL RESULTS

To implement this project we have used OPENAI GYM tool from python language and sensors to detect obstacle. We don't have any sensors so we will use GYM simulation to find path and avoid obstacle. The results are shown in figure 4 to figure 7.

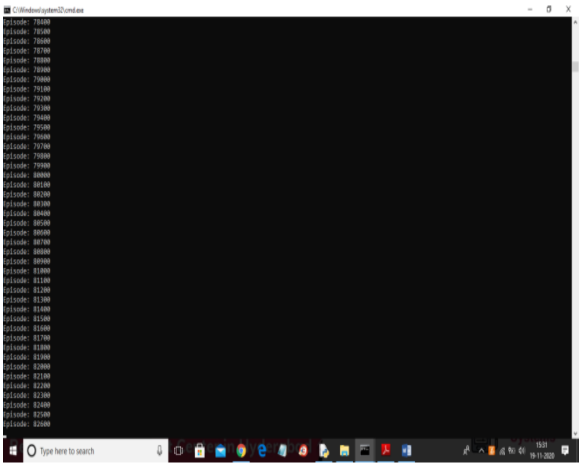

Figure.4: Find optimal path screen

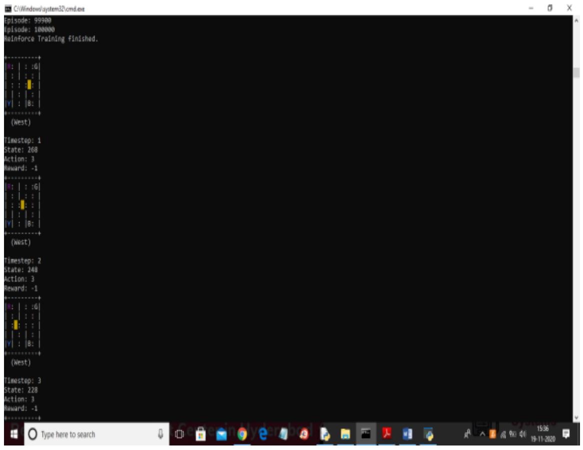

Figure.5: Yellow vehicle start moving

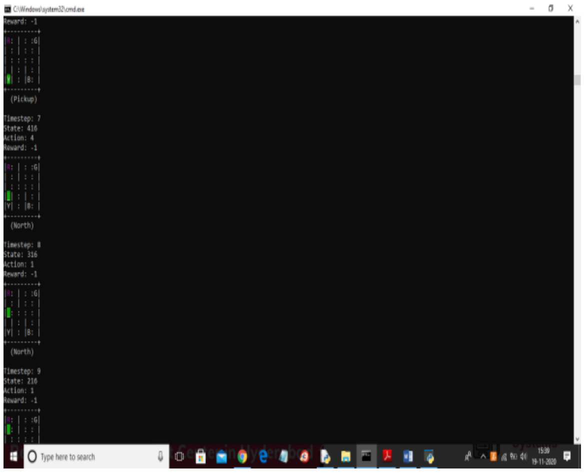

Figure.6: Vehicle starts moving towards destination

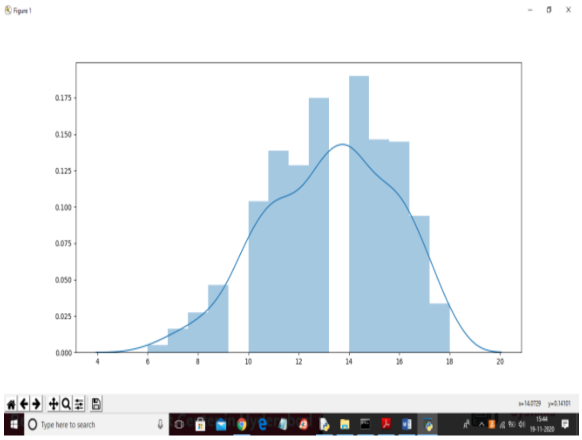

Figure.7: Rewards graph

V. CONCLUSION

In this work, we have demonstrated that RL is a viable approach to the challenging dual-objective problem of controlling a vessel to follow a path given by a priori known way-points while avoiding obstacles along the way without relying on a map. More specifically, we have shown that the state-of-the-art PPO algorithm converges to a policy that yields intelligent guidance behavior under the presence of non-moving obstacles surrounding and blocking the desired path. By means of extensive testing, we have observed that, even in challenging test environments with high obstacles densities, the agent's success rate is in the high 90s when λ is set such that it induces a strict path adherence bias, and close to 100% when a more defensive strategy is chosen.

REFERENCES

[1] R. W. Beard and T. W. McLain, Small Unmanned Aircraft: Theory and Practice. Princeton, NJ, USA: Princeton Univ. Press, 2012.
[2] O. Khatib, ''Real-time obstacle avoidance for manipulators and mobile robots,'' Int. J. Robot. Res., vol. 5, no. 1, pp. 90–98, Jul. 2016.
[3] J. Borenstein and Y. Koren, ''The vector field histogram-fast obstacle avoidance for mobile robots,'' IEEE Trans. Robot. Autom., vol. 7, no. 3, pp. 278–288, Jun. 1991.
[4] D. Panagou, ''Motion planning and collision avoidance using navigation vector fields,'' in Proc. IEEE Int. Conf. Robot. Autom. (ICRA), May 2014, pp. 2513–2518.
[5] D. Fox, W. Burgard, and S. Thrun, ''The dynamic window approach to collision avoidance,'' IEEE Robot. Autom. Mag., vol. 4, no. 1, pp. 23–33, Mar. 1997.
[6] O. Brock and O. Khatib, ''High-speed navigation using the global dynamic window approach,'' in Proc. IEEE Int. Conf. Robot. Autom., vol. 1, May 1999, pp. 341–346.
[7] B.-O.-H. Eriksen, M. Breivik, K. Y. Pettersen, and M. S. Wiig, ''A modified dynamic window algorithm for horizontal collision avoidance for AUVs,'' in Proc. IEEE Conf. Control Appl. (CCA), Sep. 2016, pp. 499–506.
[8] P. Fiorini and Z. Shiller, ''Motion planning in dynamic environments using velocity obstacles,'' Int. J. Robot. Res., vol. 17, no. 7, pp. 760–772, Jul. 1998. [Online]. Available: http://dblp.unitrier.de/db/journals/ijrr/ijrr17.html#FioriniS98
[9] D. K. M. Kufoalor, E. F. Brekke, and T. A. Johansen, ''Proactive collision avoidance for ASVs using a dynamic reciprocal velocity obstacles method,'' in Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS), Oct. 2018, pp. 2402–2409.
[10] Y. Chen, H. Peng, and J. Grizzle, ''Obstacle avoidance for low-speed autonomous vehicles with barrier function,'' IEEE Trans. Control Syst. Technol., vol. 26, no. 1, pp. 194–206, Jan. 2018.