DSSE: An environment for simulation of reinforcement learning-empowered drone swarm maritime search and rescue missions

Renato Laffranchi Falcão [1*¶], **Jorás Custódio Campos de Oliveira** [1*], **Pedro Henrique Britto Aragão Andrade** [1*], **Ricardo Ribeiro Rodrigues** [1*], **Fabrício Jailson Barth** [1*], and **José Fernando Basso Brancalion** [2*]

1 Insper, Brazil **2** Embraer, Brazil **¶** Corresponding author ***** These authors contributed equally.

Summary

The goal of this project is to advance research in maritime search and rescue missions using Reinforcement Learning techniques. The software provides researchers with two distinct environments: one simulates shipwrecked people drifting with maritime currents, creating a stochastic setting for training and evaluating autonomous agents; the other features a realistic particle simulation for mapping and optimizing search area coverage by autonomous agents.

Both environments adhere to open-source standards and offer extensive customization options, allowing users to tailor them to specific research needs. These tools enable Reinforcement Learning agents to learn efficient policies for locating shipwrecked individuals or maximizing search area coverage, thereby enhancing the effectiveness of maritime rescue operations.

Statement of need

Maritime navigation plays a crucial role across various domains, including leisure activities and commercial fishing. However, maritime transportation is particularly significant as it accounts for 80% to 90% of global trade (*Safety and Shipping Review*, 2023). While maritime navigation is essential for global trade, it also poses significant safety risks, as evidenced by the World Health Organization's report (*Drowning*, 2023) of approximately 236,000 annual drowning deaths worldwide. Therefore, maritime safety is essential, demanding significant enhancements in search and rescue (SAR) missions. It is imperative that SAR missions minimize the search area and maximize the chances of locating the search object.

To achieve this objective, traditional SAR operations have utilized path planning algorithms such as parallel sweep, expanding square, and sector searches ("Chapter 5. Search Techniques and Operations," 2022). However, these methods have not been optimal. Trummel & Weisinger (Trummel & Weisinger, 1986) demonstrated that finding an optimal search path, where the agent must search all sub-areas using the shortest possible path, is NP-complete. Recent research, however, proposes a different approach using Reinforcement Learning (RL) algorithms instead of pre-determined search patterns (Ai et al., 2021; Wu et al., 2024). This is based on the belief that RL can develop new, more efficient search patterns tailored to specific applications. The hypothesis is that maximizing reward fosters generalization abilities, thereby creating powerful agents (Silver et al., 2021). Such advancements could potentially save more lives.

The two primary metrics for evaluating an efficient search are coverage rate and time to detection. Coverage rate is the proportion of the search area covered by the search units over a

specific period. Higher coverage rates typically indicate more effective search strategies. Time to detection is the time taken from the start of the search operation to the successful detection of the target. Minimizing this time is often a critical objective in SAR missions.

Expanding on the state-of-the-art research presented by (Ai et al., 2021; Wu et al., 2024), this project introduces a unique simulation environment that has not been made available by other researchers. Additionally, this new environment enables experiments on search areas that are significantly larger than those used in existing research.

Functionality

In order to contribute to research on the effectiveness of integrating RL techniques into SAR path planning, the Drone Swarm Search Environment (DSSE), distributed as a Python package, was designed to provide a training environment using the PettingZoo (J. Terry et al., 2021) interface. Its purpose is to facilitate the training and evaluation of single or multi-agent RL algorithms. Additionally, it has been included as a third-party environment in the official PettingZoo documentation (Jordan Terry et al., 2021).
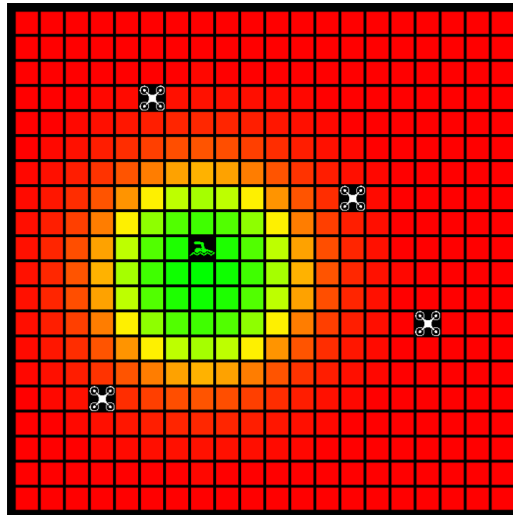

Figure 1: Simulation environment showcasing the algorithm's execution.

The environment depicted in Figure 1 comprises a grid, a probability matrix, drones, and an arbitrary number of persons-in-water (PIW). The movement of the PIW is influenced by, but not identical to, the dynamics of the probability matrix, which models the drift of sea currents impacting the PIW (Wu et al., 2023). The probability matrix itself is defined using a two-dimensional Gaussian distribution, which expands over time, thus broadening the potential search area. This expansion simulates the diffusion of the PIW, approximating the zone where drones are most likely to detect them. Moreover, the environment employs a reward function that incentivizes the speed of the search, rewarding the agents for shorter successful search durations.

The package also includes a second environment option. Similar to the first, this alternative setup is designed for training agents, but with key differences in its objectives and mechanics. Unlike the first environment, which rewards agents for speed in their searches, this second option rewards agents that cover the largest area without repetition. It incorporates a trade-off by using a stationary probability matrix, but enhances the simulation with a more advanced Lagrangian particle model (Dagestad et al., 2018) for pinpointing the PIW's position. Moreover, this environment omits the inclusion of shipwrecked individuals, focusing instead on promoting

research into how agents can learn to efficiently expand their search coverage over broader areas.

Using this environment, any researcher or practitioner can write code and execute an agent's training, such as the source code presented below.

```python
from DSSE import DroneSwarmSearch

env = DroneSwarmSearch()

observations, info = env.reset()

rewards = 0
done = False
while not done:
    actions = policy(observations, env.get_agents())
    observations, rewards, terminations, truncations, infos = env.step(actions)
    done = any(terminations.values()) or any(truncations.values())
```

The grid is divided into square cells, each representing a quadrant with sides measuring 130 meters in the real world. This correlation with real-world dimensions is crucial for developing agents capable of learning from realistic motion patterns. The drones, which are controlled by RL algorithms, serve as these agents. During the environment's instantiation, users define the drones' nominal speeds. These drones can move both orthogonally and diagonally across the grid, and they are equipped to search each cell for the presence of the PIW.

Several works have been developed over the past few years to define better algorithms for the search and rescue of shipwrecks (Ai et al., 2021; Wu et al., 2024). However, no environment for agent training is made available publicly. For this reason, the development and provision of this environment as a Python library and open-source project are expected to have significant relevance to the machine learning community and ocean safety.

This new library makes it possible to implement and evaluate new RL algorithms, such as Deep Q-Networks (DQN) (Mnih et al., 2015) and Proximal Policy Optimization (PPO) (Schulman et al., 2017), with little effort. Additionally, several state-of-the-art RL algorithms have already been implemented and are available (Rodrigues et al., 2024). An earlier iteration of this software was utilized in research that compared the Reinforce algorithm with the parallel sweep path planning algorithm (Abreu et al., 2023).

References

Abreu, L. D. M. de, Carrete, L. F. S., Castanares, M., Damiani, E. F., Brancalion, J. F. B., & Barth, F. J. (2023). *Exploration and rescue of shipwreck survivors using reinforcement learning-empowered drone swarms* (pp. 64–69). Simpósio de Aplicações Operacionais em Áreas de Defesa (SIGE).

Ai, B., Jia, M., Xu, H., Xu, J., Wen, Z., Li, B., & Zhang, D. (2021). Coverage path planning for maritime search and rescue using reinforcement learning. *Ocean Engineering*, *241*, 110098. https://doi.org/10.1016/j.oceaneng.2021.110098

Chapter 5. Search techniques and operations. (2022). In *International aeronautical and maritime search and rescue manual: Vol. II*. International Maritime Organization; International Civil Aviation Organization. ISBN: 9789280117356

Dagestad, K.-F., Röhrs, J., Breivik, Ø., & Ådlandsvik, B. (2018). OpenDrift v1.0: A generic framework for trajectory modelling. *Geoscientific Model Development*, *11*(4), 1405–1420. https://doi.org/10.5194/gmd-11-1405-2018

Drowning. (2023). World Health Organization. https://www.who.int/news-room/fact-sheets/detail/drowning

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., Petersen, S., Beattie, C., Sadik, A., Antonoglou, I., King, H., Kumaran, D., Wierstra, D., Legg, S., & Hassabis, D. (2015). Human-level control through deep reinforcement learning. *Nature*, *518*(7540), 529–533. https://doi.org/10.1038/nature14236

Rodrigues, R. R., Oliveira, J. C. C. de, Andrade, P. H. B. A., & Falcão, R. L. (2024). *Algorithms for drone swarm search (DSSE)*. https://github.com/pfeinsper/drone-swarm-search-algorithms

Safety and shipping review (p. 4). (2023). Allianz Global Corporate & Specialty. https://commercial.allianz.com/news-and-insights/reports/shipping-safety.html

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., & Klimov, O. (2017). *Proximal policy optimization algorithms*. https://doi.org/10.48550/arXiv.1707.06347

Silver, D., Singh, S., Precup, D., & Sutton, R. S. (2021). Reward is enough. *Artificial Intelligence*, *299*, 103535. https://doi.org/10.1016/j.artint.2021.103535

Terry, J., Black, B., Grammel, N., Jayakumar, M., Hari, A., Sullivan, R., Santos, L. S., Dieffendahl, C., Horsch, C., Perez-Vicente, R., Williams, N., Lokesh, Y., & Ravi, P. (2021). PettingZoo: Gym for multi-agent reinforcement learning. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P. S. Liang, & J. W. Vaughan (Eds.), *Advances in neural information processing systems* (Vol. 34, pp. 15032–15043). Curran Associates, Inc. https://proceedings.neurips.cc/paper_files/paper/2021/file/7ed2d3454c5eea71148b11d0c25104ff-Paper.pdf

Terry, Jordan, Black, B., Grammel, N., Jayakumar, M., Hari, A., Sullivan, R., Santos, L., Perez, R., Horsch, C., Dieffendahl, C., Williams, N., & Lokesh, Y. (2021). *PettingZoo: Gym for multi-agent reinforcement learning*. https://github.com/Farama-Foundation/PettingZoo

Trummel, K., & Weisinger, J. (1986). The complexity of the optimal searcher path problem. *Operations Research*, *34*(2), 324–327.

Wu, J., Cheng, L., & Chu, S. (2023). Modeling the leeway drift characteristics of persons-in-water at a sea-area scale in the seas of china. *Ocean Engineering*, *270*, 113444. https://doi.org/10.1016/j.oceaneng.2022.113444

Wu, J., Cheng, L., Chu, S., & Song, Y. (2024). An autonomous coverage path planning algorithm for maritime search and rescue of persons-in-water based on deep reinforcement learning. *Ocean Engineering*, *291*, 116403. https://doi.org/10.1016/j.oceaneng.2023.116403