Benchmarking Reinforcement Learning Algorithms with Search and Rescue Tasks

Abdullah Khaled[1,2], Zijian Guo[2], Wenchao Li[2]

Wakeland High School, 10700 Legacy Dr, Frisco, TX 75034[1]; Boston University Photonics Center, 8 St. Mary's St, Boston, MA 02215[2]

Background

- Reinforcement learning (RL) models struggle with long-horizon tasks: ones where there are multiple goals the agent must reach in a predetermined order.
- Specification-guided RL[1] uses language such as Linear Temporal Reasoning (e.g. LTL) to specify the goals and constraints of a task. For example, Equation 1:

$$!B \cup (A \,\&\, (!B \cup C)) \quad (1)$$

correlates to "Do not (!) touch B until (U) you reach A and then (&) do not touch B until you reach C." A valid path, therefore, could look like:

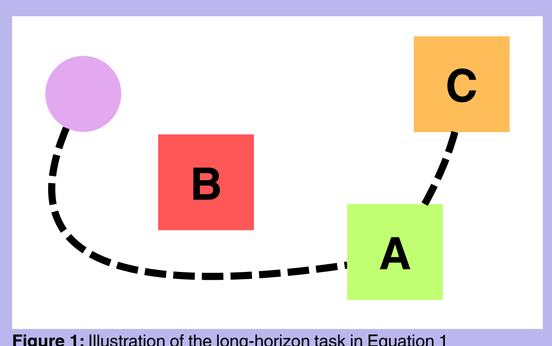

Figure 1: Illustration of the long-horizon task in Equation 1

- Search and Rescue (SAR) environments offer realistic long-horizon tasks.
- Benchmark environment(s) to evaluate specification-guided RL models exist[2], but not for SAR applications.

Method: Environment

Key Features:

- **Agent:** What the model is controlling. The model controls the forward/backward movement and rotation of the agent.
- **Border**: A box around the environment so the agent cannot leave the bounds. The episode terminates if an agent hits the border.
- **Surface casualties (SC):** Goals agent can detect with lidar or vision unless an object occludes it (e.g. other agents, buildings, walls, etc.).
- **Entrapped casualties (EC)**: Another goal that cannot be seen by lidar or vision unless the agent enters a building and an entrapped casualty is in that building.
- **Buildings**: An object that is globally visible by lidar and vision; i.e. the agent can always see buildings.
- **Walls**: Objects that obstruct the agent's vision/lidar and terminate the episode if contacted.
- **Specifications**: Evaluation uses user-specified arbitrary specifications.
- Once all casualties are rescued (surface and entrapped), the episode ends prematurely. If the episode times out, the episode truncates.

Variations:

- **Multi-Agent:** The environment supports variations of tasks with more than one agent.
- **Complexity Scaling:** We created three environments with increasing task difficulty:
  - Level 0: One surface and entrapped casualty per agent.
  - Level 1: Same as Level 0, but with ten walls scattered throughout the environment.
  - Level 2: Same as Level 1, but with two surface and entrapped casualties per agent.
- **Configurability:** Users can create custom SAR environments with adjustable settings.

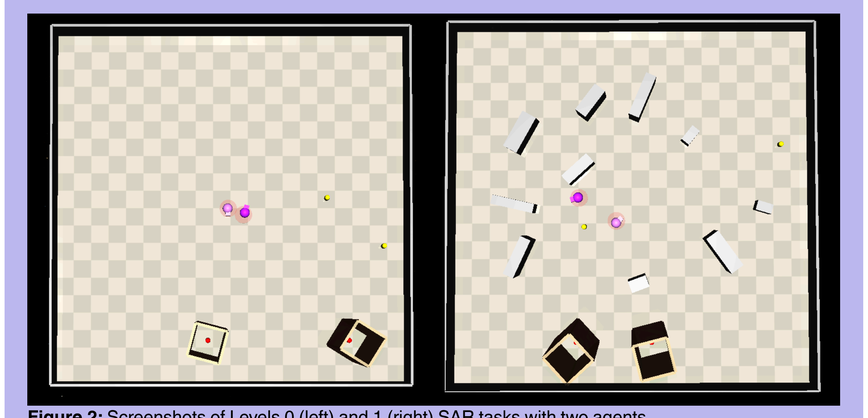

Figure 2: Screenshots of Levels 0 (left) and 1 (right) SAR tasks with two agents.

Results

Evaluated Model Categories

- **Unconstrained (PPO[3])**: Only tries to maximize reward (i.e. casualties rescued) and does not consider costs from collisions with the border or interior walls.
- **Constrained (PPO Lagrangian[4])**: Tries to maximize reward and minimize cost (max-min optimization) using a penalty coefficient λ.
- **Specification-guided (GenZ-LTL[5]):** Trains on individual subgoals (e.g. reach any entrapped casualty); evaluation uses LTL.

Training

- We trained the models on Levels 0 and 1 of the SAR environments.
- Training was done in single-agent setting.

Evaluation

- We evaluated using the specification *"Avoid all SC and walls until all EC are rescued, then avoid all walls until all SC are rescued."*
- Evaluation was done in multi-agent (N=2) variation with time out after 2500 timesteps.

	PPO			PPO-Lagrangian			GenZ-LTL		
	$\eta_s \uparrow$	$\eta_v \downarrow$	$\mu \downarrow$	$\eta_s \uparrow$	$\eta_v \downarrow$	$\mu \downarrow$	$\eta_s \uparrow$	$\eta_v \downarrow$	$\mu \downarrow$
Level 0	0.070 ± 0.102	0.34 ± 0.31	1157.83 ± 181.63	0.272 ± 0.139	0.16 ± 0.06	905.78 ± 72.78	**0.778 ± 0.045**	**0.09 ± 0.07**	**805.24 ± 54.03**
Level 1	0.004 ± 0.005	**0.39 ± 0.30**	**631.00 ± 8.49**	0.008 ± 0.013	0.62 ± 0.14	1714.35 ± 770.25	**0.144 ± 0.063**	0.85 ± 0.07	766.49 ± 77.00

Table 1: Evaluation result metrics. Symbols explained below. Bold represents best for that level.

Metrics

- For all metrics, (↑) indicates a higher value is better, while (↓) indicates the inverse.
- Success Rate (η_s) is the ratio of episodes where all casualties were rescued.
- Violation Rate (η_v) is the ratio of episodes where agent violated specification.
- Success Episode Length (μ) is the average length of successful episodes.

Visualization

- Pictured to the right are samples of successful and violated runs from GenZ-LTL
- **Indecision**: When presented with two buildings equidistant from one another, the agent struggles to choose.
- **Competition**: Since policies run independently, agents do not coordinate, which can result in avoidable failures.

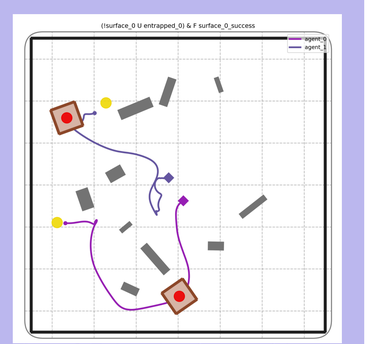

Figure 3: Trajectories of successful run

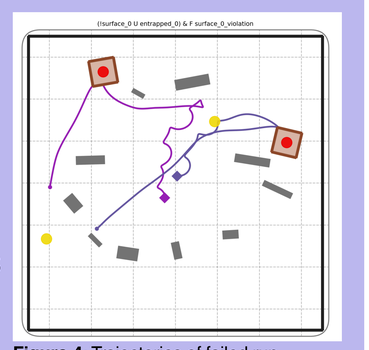

Figure 4: Trajectories of failed run

Conclusion

- GenZ-LTL significantly outperforms traditional models in both environments, though all models struggle in Level 1.
- GenZ-LTL had higher successful mean episode length than PPO.
- GenZ-LTL has lower violation rate on Level 0 but highest on Level 1.
- Violation rate and mean episode success length are likely skewed due to highly infrequent successful episodes.

References

(1) Icarte, R. T.; Klassen, T.; Valenzano, R.; McIlraith, S. Using Reward Machines for High-Level Task Specification and Decomposition in Reinforcement Learning. Proceedings of the 35th International Conference on Machine Learning 2018, 80, 2107–2116.
(2) Guo, Z.; Işık, İ.; Ahmad, H. M.; Li, W. SpecRLBench: A Benchmark for Generalization in Specification-Guided Reinforcement Learning. arXiv preprint 2026.
(3) Guo, Z.; Işık, İ.; Ahmad, H. M.; Li, W. One Subgoal at a Time: Zero-Shot Generalization to Arbitrary Linear Temporal Logic Requirements in Multi-Task Reinforcement Learning. Advances in Neural Information Processing Systems 2026, 38, 77500–77529.
(4) Schulman, J.; Wolski, F.; Dhariwal, P.; Radford, A.; Klimov, O. Proximal Policy Optimization Algorithms. arXiv preprint 2017.
(5) Ray, A.; Achiam, J.; Amodei, D. Benchmarking Safe Exploration in Deep Reinforcement Learning. arXiv preprint arXiv:1910.01708 2019.

Acknowledgement

I would like to thank Professor Wenchao Li and graduate student Zijian Guo for offering me their time and guidance through my project. I also want to thank my family for supporting me throughout RISE and my academic career. Finally, I want to thank Boston University for offering me the opportunity to perform research as part of the RISE program.