Extended Abstract: Safe Learning from Hallucination for Navigation in the Wild

Xuesu Xiao[1], **Bo Liu**[1], **Garrett Warnell**[2], **Peter Stone**[1, 3]
[1]University of Texas at Austin, [2]Army Research Laboratory, [3]Sony AI
xiao@cs.utexas.edu, bliu@cs.utexas.edu,
garrett.a.warnell.civ@mail.mil, pstone@cs.utexas.edu

Introduction

One conundrum of using machine learning to find safe navigation systems that can be deployed in the wild is that, in order to produce safe motions in obstacle-occupied spaces, a robot needs to first gather experience in those dangerous spaces *before* it has learned how to generate safe motions. This conundrum becomes even more severe when the goal is to navigate in challenging or even adversarial real-world scenarios. One solution to learn navigation in the wild is to learn from pre-supplied, good demonstrations (e.g., from a human expert (Xiao et al. 2020; Wang et al. 2021)) or perform exploration based on trial-and-error (Xu et al. 2021) in the deployment environment (or one very similar to it), but both of these approaches become costly in dangerous spaces in the wild.

Although learning to find an optimal plan among the variety of maneuvers the robot can perform is difficult without many trial-and-error attempts or an expert who is already capable of doing so, given a plan performed in obstacle-free space, it is relatively easy to find an obstacle configuration for which that plan is optimal.

Based on this observation, instead of finding the optimal motion plan for a specific obstacle configuration, we consider this "dual" problem of classical motion planning and seek to find the obstacle configuration(s) where a specific motion plan is guaranteed to be optimal. We name this process *hallucination*. Solving this problem gives us the freedom to allow random exploration in a completely safe obstacle-free space and collect an extensive amount of motion plans, whose optimally will be assured by a class of *hallucination* techniques. In this work, we introduce two of those techniques: to hallucinate (1) the (unique) *most constrained* and (2) a (not unique) *minimal* obstacle configuration. We then train an end-to-end motion planner that can produce motions to navigate through realistic obstacles during deployment in the wild. Both methods are tested on a physical mobile robot in real-world cluttered environments.

Safe Learning from Hallucination

Given a robot's configuration space (C-space) partitioned by unreachable (obstacle) and reachable (free) configurations, $C = C_{obst} \cup C_{free}$, we define the classical motion planning problem as to find a function $f(\cdot)$ that can be used to produce optimal plans $p = f(C_{obst} \mid c_c, c_g)$ that result in the robot moving from the robot's current configuration c_c to a specified goal configuration c_g without intersecting (the interior of) C_{obst}. Here, a plan $p \in \mathscr{P}$ is a sequence of low-level actions $\{u_i\}_{i=1}^{t}$. We introduce two hallucination methods to approach the "dual" problem of finding optimal $f(\cdot)$.

Hallucinating the Most Constrained Obstacle Space (Xiao et al. 2021)

Since different C_{obst} can lead to the same plan, the left inverse of f, f^{-1}, is not well defined (see Fig. 1 left). However, we can instead define a similar function $g(\cdot)$ such that $C^*_{obst} = g(p \mid c_c, c_g)$, where C^*_{obst} denotes the C-space's *most constrained* unreachable set corresponding to p.[1] Formally, given a plan p and the set of all unreachable sets $\mathscr{C}_{obst}$, we say

$$\begin{aligned} C^*_{obst} = g(p \mid c_c, c_g) \;\text{ iff }\; \forall C_{obst} \in \mathscr{C}_{obst}, \\ f(C_{obst} \mid c_c, c_g) = p \implies C_{obst} \subseteq C^*_{obst}, \end{aligned} \tag{1}$$

We denote the corresponding reachable set of C as $C^*_{free} = C \setminus C^*_{obst}$. We call $g(\cdot)$ the *most constrained* hallucination function and the output of $g(\cdot)$ a *most constrained* hallucination. This hallucination can be projected onto the robot's sensors. For example, for a LiDAR sensor, we perform ray casting from the sensor to the boundary between C^*_{obst} and C^*_{free} in order to project the hallucination onto the range readings (Fig. 1 right). Given the hallucination C^*_{obst} for p, the only viable (and therefore optimal) plan is $p = g^{-1}(C^*_{obst} \mid c_c, c_g)$. Note that $g(\cdot)$ is bijective and its inverse $g^{-1}(\cdot)$ is well defined. Leveraging machine learning, $g^{-1}(\cdot)$ is represented using a function approximator $g_\theta^{-1}(\cdot)$. Note that we aim to approximate $g_\theta^{-1}(\cdot)$ instead of the original $f(\cdot)$ due to the vastly different domain size: the *most constrained* ($\mathscr{C}^*_{obst}$) vs. *all* ($\mathscr{C}_{obst}$) unreachable sets.

During deployment, we use a smoothed coarse global path from a global planner to generate runtime hallucination so $g_\theta^{-1}(\cdot)$ does not need to generalize to unseen scenarios. Other components, including a Turn in Place, Recovery

[1]Technically, c_g can be uniquely determined by p and c_c, but we include it as an input to $g(\cdot)$ for notational symmetry with $f(\cdot)$.

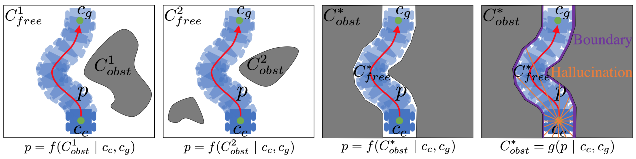

Figure 1: $p = f(C_{obst} \mid c_c, c_g)$ and $C^*_{obst} = g(p \mid c_c, c_g)$

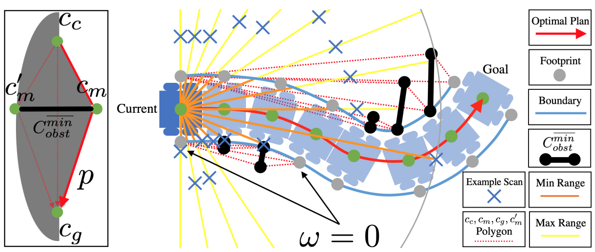

Figure 2: Left: $\overline{C^{min}_{obst}}$ defined by three consecutive configurations with c_c, c_m, c_g, and symmetry point c'_m. Right: LiDAR reading randomly sampled between min and max range.

Behavior, and Speed Modulation modules, are used in conjunction with $g_\theta^{-1}(\cdot)$ to address inevitable out-of-distribution scenarios and adapt to the real C-space.

Hallucinating the Minimal Obstacle Space (Xiao, Liu, and Stone 2021)

Learning from hallucinated C^*_{obst} can efficiently reduce input space, and therefore learning complexity, but requires runtime hallucination and other components during deployment. Hallucination of a *minimal* obstacle space generates C^{min}_{obst}, which is a minimal set of obstacle configurations required to cause the plan p to be optimal, We then randomly samples augmentations to the minimal unreachable set. Formally, we define the set of C^{min}_{obst} as:

$$\mathscr{C}^{min}_{obst} = \{C^{min}_{obst} \mid \forall c \in C^{min}_{obst}, \\ f(C^{min}_{obst} \setminus \{c\} \mid c_c, c_g) \neq f(C^{min}_{obst} \mid c_c, c_g)\} \quad (2)$$

We use a special $\overline{C^{min}_{obst}}$ to approximate any $C^{min}_{obst} \in \mathscr{C}^{min}_{obst}$ (Fig. 2 left). This approximation is sufficient when the robot trajectory is composed of a dense sequence of configurations and $\overline{C^{min}_{obst}}$ is instantiated on discrete LiDAR beams, which will be shown empirically. As shown in Fig. 2 right, the max range of a LiDAR beam is determined by $\overline{C^{min}_{obst}}$ (if the beam intersects $\overline{C^{min}_{obst}}$) or the sensor's physical limit (if not), while the min range for each beam is determined by the boundary of the robot path. A random range is sampled between the min and max values, considering possible continuity among neighboring beams and being offset for uncertainty/safety induced by the optimal plan p. Therefore, many C_{obst} can be augmented based on $\overline{C^{min}_{obst}}$. We then train a parameterized policy $f_\theta(\cdot)$ to approximate classical motion planner $f(\cdot)$.

The advantage of augmenting $\overline{C^{min}_{obst}}$ and generating many C_{obst} is, during deployment, no runtime hallucination with the help of a global path and other extra components are required. The learned $f_\theta(\cdot)$ can plan in response to the real perception and adapt to the actual scenarios on its own.

Physical Experiments[2]

Two datasets are collected by two random exploration policies in an obstacle-free space in simulation: one with mostly constant 0.4m/s linear velocity ($v \approx 0.4$m/s) and varying angular velocity ($\omega \in [-1.57, 1.57]$rad/s), the other with varying $v \in [0, 1.0]$m/s and $\omega \in [-1.57, 1.57]$rad/s. If trained on the first dataset, the speed of the planner output is modulated by a Model Predictive Control based collision probability checker, achieving a max $v = 0.6m/s$. Four neural network based planners are trained using the two datasets and two hallucination techniques. Simulated (Perille et al. 2020) and physical experiments are performed. While the *minimal* hallucination works well on both datasets and outperforms all other variants, and even the classical DWA motion planner, the *most constrained* hallucination only performs well on the 0.4m/s dataset, because learning from varying speed while hallucinating only the most constrained space causes ambiguity for the learner.

Conclusions

Instead of seeking an optimal motion plan for an obstacle configuration, we present two learning from hallucination techniques that approach the classical motion planning problem from the opposite direction: find the obstacle configuration(s), where a motion plan is optimal. Video links of the physical experiments of the hallucination methods are provided, along with references to the detailed papers.

References

Perille, D.; Truong, A.; Xiao, X.; and Stone, P. 2020. Benchmarking metric ground navigation. In *2020 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, 116–121. IEEE.

Wang, Z.; Xiao, X.; Liu, B.; Warnell, G.; and Stone, P. 2021. Appli: Adaptive planner parameter learning from interventions. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE.

Xiao, X.; Liu, B.; and Stone, P. 2021. Agile Robot Navigation through Hallucinated Learning and Sober Deployment. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE.

Xiao, X.; Liu, B.; Warnell, G.; Fink, J.; and Stone, P. 2020. APPLD: Adaptive Planner Parameter Learning From Demonstration. *IEEE Robotics and Automation Letters* 5(3): 4541–4547.

Xiao, X.; Liu, B.; Warnell, G.; and Stone, P. 2021. Toward Agile Maneuvers in Highly Constrained Spaces: Learning from Hallucination. *IEEE Robotics and Automation Letters* 1503–1510.

Xu, Z.; Dhamankar, G.; Nair, A.; Xiao, X.; Warnell, G.; Liu, B.; Wang, Z.; and Stone, P. 2021. Applr: Adaptive planner parameter learning from reinforcement. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE.

[2] Videos: https://www.youtube.com/watch?v=AE-KgxJS-iE&t=48s and https://www.youtube.com/watch?v=LZcBN9zgtXg&t=50s.