Goal Space Abstraction in Hierarchical Reinforcement Learning via Reachability Analysis

Mehdi Zadem, Sergio Mover, Sao Mai Nguyen

Mehdi Zadem, Sergio Mover, Sao Mai Nguyen. Goal Space Abstraction in Hierarchical Reinforcement Learning via Reachability Analysis. Intrinsically Motivated Open-ended Learning IMOL 2023, Sep 2023, Paris, France.

Goal Space Abstraction in Hierarchical Reinforcement Learning via Reachability Analysis

Mehdi Zadem*†, Sergio Mover*, Sao Mai Nguyen†
*LIX, CNRS, École Polytechnique, Institute Polytechnique de Paris, Palaiseau, France
†Flowers Team, U2IS, ENSTA Paris, Institut Polytechnique de Paris & Inria, Palaiseau, France

***Abstract*—Open-ended learning benefits immensely from the use of symbolic methods for goal representation as they offer ways to structure knowledge for efficient and transferable learning. However, the existing Hierarchical Reinforcement Learning (HRL) approaches relying on symbolic reasoning are often limited as they require a manual goal representation. The challenge in autonomously discovering a symbolic goal representation is that it must preserve critical information, such as the environment dynamics. In this work, we propose a developmental mechanism for subgoal discovery via an emergent representation that abstracts (i.e., groups together) sets of environment states that have similar roles in the task. We create a HRL algorithm that gradually learns this representation along with the policies and evaluate it on navigation tasks to show the learned representation is interpretable and results in data efficiency.**

I. Introduction

Symbol emergence is key for developmental learning to tackle the curse of dimensionality and scale up to open-ended high-dimensional sensorimotor space, by allowing symbolic reasoning, compositionality, hierarchical organisation of the knowledge, etc. While symbol emergence has been recently investigated for the sensor data, action symbolization can lead to a repertoire of various movement patterns by bottom-up processes, which can be used by top-down processes such as composition to form an action sequence [1], planning and reasoning for more efficient learning, as reviewed in [2]. Sensorimotor symbol emergence thus is key to scaling up primitive actions into complex actions for open-ended learning, using compositionality [3] and hierarchy [4].

Action hierarchies are the core idea of Hierarchical Reinforcement Learning (HRL) that decomposes a task into easier subtasks. In particular, in Feudal HRL [5] a high-level agent selects subgoals that a low-level agent learns to achieve. The performance of Feudal HRL depends on the "hierarchical division of the available state space" [5], the representation of the goals that the high level agent uses to decompose a task. Yet, only few algorithms learn it automatically [6], while others either use directly the state space [7] or manually provide a representation [8], [9]. In this research, we tackle the problem of learning automatically, while learning the policy, a discrete interpretable goal representation from continuous observations that expresses the task structure for data-efficiency.

We introduce a novel goal space representation and a feudal HRL algorithm, GARA (Goal Abstraction via Reachability Analysis), that develops such a representation while simultaneously learning a hierarchical policy from exploration data. The representation emerges through a developmental process, gradually gaining precision from a bottom-up manner, by leveraging data acquired from exploration. This discretisation of the environment is used to orient top-down process of the goal-directed exploration, that in turn helps improving policies and this representation.

II. Formulation

The goal space $\mathcal{G}$ is formulated as a partition of the state space $\mathcal{S}$ into n disjoint sets of states $\mathcal{G} = \{G_0, \ldots, G_n\}$ s.t $\bigcup_{G \in \mathcal{G}} G = \mathcal{S}$ and and $\forall G, G' \in \mathcal{G}$, $G \cap G' = \emptyset$ if $G \neq G'$. We define $R_k(G, G')$ as the set of states reached when starting from a state in G and applying the low-level policy $\pi^{Low}(s \in G, G')$ targeting G' for k steps. This goal space should satisfy the reachability property: $\forall\ G, G' \in \mathcal{G}, R_k(G, G') \subseteq G'$ or $R_k(G, G') \cap G' = \emptyset$. Intuitively, this property expresses that each goal G would group together states with a similar role in the task in terms of their ability to reach other goals. Inversely, if only some states in G manage to reach the target G' then G contains states having different roles. This means that environment dynamics are not completely captured. In the following section we present GARA (Goal Abstraction via Reachability Analysis) that concurrently learns a hierarchical policy and the abstract goal space.

III. Methodology

GARA is a Feudal HRL algorithm that learns two policies; a high-level policy $\pi^{High} : \mathcal{S} \to \mathcal{G}$ selects goals $G_i \sim \pi^{High}(s)$, and a low-level goal-conditioned policy $\pi^{Low} : \mathcal{S} \times \mathcal{G} \to \mathcal{A}$ that learns how to best achieve these goals by choosing actions in the action space $\mathcal{A}$ s.t $a_t \sim \pi^{Low}(s, G_i)$. π^{High} is rewarded by the environment reward, while π^{Low} is rewarded with respect to its ability to reach the selected goal.

Learning the goal space comes down to identifying which states in each goal exhibit similar reachability behaviours. To this end, GARA trains a neural network from data acquired during exploration after each learning episode. This network is called the forward model $\mathcal{F}_k : \mathcal{S} \times \mathcal{G} \to \mathcal{S}$ such that $\mathcal{F}_k(s_t, G')$ predicts the state s_{t+k} reached after applying $\pi^{Low}(s, G')$ for k steps. A core idea of GARA, is that the reachability relations are computed over sets of states. To derive this from $\mathcal{F}_k$, we resort to a formal verification tool Ai2 [10] that can compute the output of a neural network given a set of inputs. More precisely, if the input to $\mathcal{F}_k$ is the set of states G, then the output should be an approximation of the

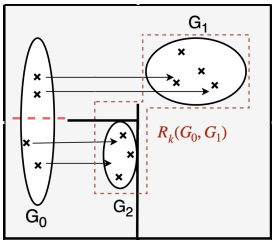


Fig. 1: In this maze, the transition starting from a state in G_0 with the policy "go right" may reach both G_1 and G_2. G_0 is thus split into two regions where all the states in the upper one reach G_1 and the states in the bottom one don't.

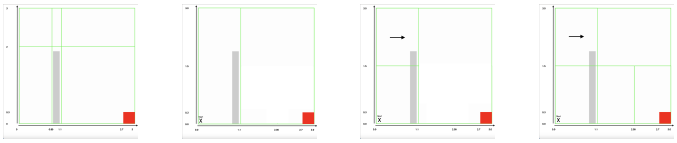
reached set of states $\tilde{R}_k(G, G')$. For each explored transition from start set G_s to destination goal G_d, Ai2 computes $\tilde{R}_k(G_s, G_d)$. If $\tilde{R}_k(G_s, G_d) \subseteq G_d$ or $\tilde{R}_k(G_s, G_d) \cap G_d = \emptyset$ then the reachability property is respected and G_s would not be refined as the behaviour is similar across its states. Otherwise if $\tilde{R}_k(G_s, G_d) \not\subseteq G_d$ and $\tilde{R}_k(G_s, G_d) \cap G_d \neq \emptyset$, G_s would be split in two sets G' and G'' on which the reachability analysis is re-conducted (we compute $\tilde{R}_k(G', G_d)$ and $\tilde{R}(G'', G_d)$). This process continues recursively until the reachability relation is decidable. G would thus be refined into two new sets G'_s and G''_s where $R_k(G'_s, G_d) \subseteq G_d$ and $\tilde{R}_k(G''_s, G_d) \cap G_d = \emptyset$. Fig.1 illustrates this process. The emerging regions constitute learning targets that are easily reachable and together would compose an abstract model for the task.

IV. Results

We focus on one experimental evaluation from our study which seeks to determine if an interpretable representation for the goal space can be learned from exploration, and if it helps the hierarchical policy to be more data-efficient. We conduct the evaluation on a U-shaped maze with a continuous state space, discrete actions controlling the agent's acceleration in 4 directions and a sparse reward is only attributed when reaching the exit. We compare GARA against some of the state-of-art approaches:

- *Feudal HRL with* **Handcrafted** *representation:* inspired by hDQN [8], this algorithm is similar in structure to GARA in using a discrete set-based goal space. This represntation is however handcrafted and fixed.
- **HIRO***:* also a feudal HRL algorithm, it relies on raw states to act as goals $\mathcal{G} = \mathcal{S}$. Additionally, it uses a goal interpolation mechanism along with hindsight experience.

a) Representation learning: Focusing first on the learned representation by GARA, Fig. 2b, Fig. 2c, and Fig. 2d show the evolution of the goal space throughout the learning at 0, 10^3, and 3×10^4 steps (for a randomly selected run of the algorithm). Initially, GARA identifies the region at the top-left corner of the maze with positive velocity which provides a good starting point to learn policies that efficiently manage to reach the right half of the maze. Later, GARA refines the right half region, which allows it to focus on the exit point. Our intuition is that such final partition results in easier to reach goals, prompting the agent to select successful behaviours.



(a) Handcrafted representation (b) Initial $\mathcal{G}$ given to GARA (c) $\mathcal{G}$ learned by GARA after 10^3 steps (d) Final $\mathcal{G}$ learned by GARA

Fig. 2: Representation of the goal space $\mathcal{G}$ in the U-shaped maze for one run of algorithm. The exit is marked in red. Green boxes show intervals for x, y and the horizontal and vertical arrows indicate the sign of the velocities v_x and v_y, respectively. No arrows indicate there are no split across v_x or v_y.

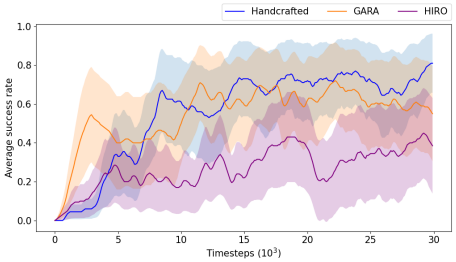


Fig. 3: Average success rate on the U-shaped Maze (20 runs).

Overall, Fig. 2 shows that GARA learns an interpretable representation from data collected during the HRL exploration.

b) Data efficiency: Fig. 3 shows that our approach manages to learn a successful hierarchical policy with a performance approaching the handcrafted representation, whereas HIRO cannot learn to solve the task within the same time frame. We attribute this to the better sample efficiency associated with the learned abstraction, as the agents successfully decompose the task into simple-to-achieve goals.

References

[1] S. M. Nguyen, N. Duminy, A. Manoury, D. Duhaut, and C. Buche, "Robots learn increasingly complex tasks with intrinsic motivation and automatic curriculum learning," *Künstliche Intelligenz*, vol. 35, 2021.
[2] T. Taniguchi, E. Ugur, M. Hoffmann, L. Jamone, T. Nagai, B. Rosman, T. Matsuka, N. Iwahashi, E. Oztop, J. Piater, and F. Wörgötter, "Symbol emergence in cognitive developmental systems: A survey," *IEEE TCDS*, vol. 11, no. 4, pp. 494–516, 2019.
[3] A. Manoury, S. M. Nguyen, and C. Buche, "Hierarchical affordance discovery using intrinsic motivation," in *HAI*. ACM, 2019.
[4] N. Duminy, S. M. Nguyen, and D. Duhaut, "Learning a set of interrelated tasks by using sequences of motor policies for a strategic intrinsically motivated learner," in *Proceedings of IEEE International Conference on Robotic Computing*, 2018.
[5] P. Dayan and G. E. Hinton, "Feudal reinforcement learning," in *NeurIPS*, vol. 5, 1992.
[6] A. S. Vezhnevets, S. Osindero, T. Schaul, N. Heess, M. Jaderberg, D. Silver, and K. Kavukcuoglu, "Feudal networks for hierarchical reinforcement learning," *CoRR*, vol. abs/1703.01161, 2017.
[7] O. Nachum, S. Gu, H. Lee, and S. Levine, "Data-efficient hierarchical reinforcement learning," in *NeurIPS 2018*, 2018.
[8] T. D. Kulkarni, K. Narasimhan, A. Saeedi, and J. Tenenbaum, "Hierarchical deep reinforcement learning: Integrating temporal abstraction and intrinsic motivation," in *NeurIPS*, vol. 29, 2016.
[9] T. Zhang, S. Guo, T. Tan, X. Hu, and F. Chen, "Generating adjacency-constrained subgoals in hierarchical reinforcement learning," in *NeurIPS*, 2020.
[10] T. Gehr, M. Mirman, D. Drachsler-Cohen, P. Tsankov, S. Chaudhuri, and M. T. Vechev, "AI2: safety and robustness certification of neural networks with abstract interpretation," in *IEEE Symposium on Security and Privacy*, 2018.