Towards hierarchical curiosity-driven exploration of sensorimotor models
Sébastien Forestier, Pierre-Yves Oudeyer

To cite this version:
Towards Hierarchical Curiosity-Driven Exploration of Sensorimotor Models

Sébastien Forestier, Pierre-Yves Oudeyer
Flowers Team — INRIA Bordeaux

Introduction

Curiosity-driven exploration mechanisms have been proposed to allow robots to actively explore high-dimensional sensorimotor spaces in an open-ended manner [1]. In such setups, intrinsic motivations based on competence progress show good results — the learner explores its sensory space with a bias toward regions which are predicted to yield a high competence progress. However, throughout its life, a developmental robot has to incrementally explore skills that add up to the hierarchy of previously learned skills, with a constraint being the cost of experimentation. We rely on the SAGG-RAC architecture (Self Adaptive Goal Generation - Robust Intelligent Active Curiosity) [2] with an intrinsic motivation that pushes the agent to explore regions where the progress of the competence to reach self-generated goals in the sensory space is higher.

Curiosity-Driven Exploration

We use and extend the Explauto library [3] that aims at studying autonomous exploration. In the Explauto framework, a sensorimotor model is learned together with an interest model that guides future exploration. These models are then used by a goal babbling control engine.

Sensorimotor model

This model stores the experimented motor commands and their associated sensory experience and builds a mapping between the motor space and the sensory space. We use the fast nearest neighbor algorithm to build this mapping but more powerful regression methods could be used instead.

Interest model

The interest model estimates how interesting it is to explore given parts of the sensory space. We use the SAGG-RAC architecture (Self Adaptive Goal Generation - Robust Intelligent Active Curiosity) [2] with an intrinsic motivation that pushes the agent to explore regions where the progress of the competence to reach self-generated goals in the sensory space is higher.

Exploration

The agent selects a goal in its sensory space according to the interest model and infers motor parameters to get close to this point with the sensorimotor model. It adds some exploration noise to discover new motor configurations, executes the command and observes the sensory experience. Finally, it updates the sensorimotor model with the new association, and the interest model with the competence to reach the goal.

Hierarchical Exploration

To study hierarchical exploration, we design a setup where the agent learns two sensorimotor models, with the second one that reuse the first one. In our setup, the first sensorimotor model in the hierarchy is a relation between 20 motor parameters of a robotic arm and the 9 parameters of the 3D trajectory of the robotic hand. The second sensorimotor model is a relation between the trajectory of the hand and the 2D position of a block at the end of the movement. Control architectures will have to learn directly a relation between the motor parameters and the position of the block.

Motor Bubbling Control

Exploration of a mapping between $M$ and $S_1$ with random motor actions.

Goal Bubbling Control

Exploration of a mapping between $M$ and $S_1$ with a competence-based intrinsic motivation (SAGG-RAC).

Simplest First

Exploration of a mapping between $M$ and $S_2$, for the first half of the trials and a mapping between $S_2$ and $S_3$ for the second half. Both explorations are made with SAGG-RAC.

Top-Down Guidance

Exploration of a mapping between $S_1$ and $S_2$. When asking to explore a hand’s movement representation, this goal $S_2$ is used to guide the exploration of a mapping between $M$ and $S_1$ which is given a certain amount of iterations to try motor configurations to reach that goal (with a black-box optimisation technique [4]).

Learning Architectures

Motor Control

The robotic setup is the left arm of the Poppy robot [5], with 1 degrees of freedom, simulated in the V-REP simulator based on the Bullet physics engine.

We use Dynamical Movement Primitive [6] to control the arm’s movement. This framework allows the production of a diversity of arm’s trajectories with few parameters. Each arm’s movement is controlled by a DMP with a starting and a goal position equal to the rest position of the motor. Each DMP is parameterized by one weight on each of 7 basis functions whose centers are distributed homogenously in the interval $[0, 1]$. The length of duration of the movement is fixed. $X$ is the 2D space of the motor parameters. $S_1$ is a 2D space representing the 3D trajectory of the hand. We also use the DMP framework to project each of the $X$ and $Y$ movements on a sensory DMP with 1 basis functions.

A zero wide block is located near the robot’s hand and can be moved in different complex ways, e.g. with the hand pushing on the top of the block or on a side. $S_2$ is the 3D space representing the position of the block at the end of the simulation. The initial position of the block is $X = (0.25, 0.15, 0.15)$. $X$ axis points in front of the robot, $Y$ axis is on its left.

Experiments

We ran different trials for each conditions (i.e., 5 trials in the first condition, 5 trials in the second condition, etc.). Each trial is made of 300 learning iterations. Each iteration takes about 5s on an Intel Core i5 Xeon E5507 node. Exploration is evaluated each 500 iterations with 2 different measures. Statistical tests are performed after 5000 iterations to compare conditions.

References


Measures

Exploration measure

This measures how diverse outcomes have been found by the agent in its goal space. It makes no assumptions about the interest of exploring different regions of the goal space from the viewpoint of the engineer. To compute the quantity of exploration we consider the goal space as an unbounded grid where we count the cells that have been reached during training.

Low Robustness

In our setup, the environment is highly stochastic in the sense that when the agent tries the same motor command different times, the variability in the generated arm movement and the collisions between the arm and the block leads to high variances in the end position of the block. Thus the agent might not succeed in reaching a previously reached position, and in other words, a high exploration does not mean a high learning.

Competent Exploration Measure

This measures how competent the robot is on the explored part of its goal space. We give the centers (black dots) of explored cells as goals to the agent, and we count the cells where the agent manages to get close to the goal (with a 0.2 cm threshold).

Results

Exploration measure

Competent Exploration measure

Discussion

Results show that exploration is better in the Top-Down Guidance condition than in the Simplest First condition, with both measures. This implies that the guidance from the top-level model is useful to drive the exploration of the lower-level model on parts of the task space that are more interesting for the exploration of the higher one.

Results also show comparable exploration for the Top-Down Guidance and the Goal Babbling control condition, whereas the competent exploration measure shows that the architectures learning an intermediate hand movement representation allow the agent to put again the object on much more diverse locations.

An intuition for this result is that when exploring around motor parameters that lead to a movement of the block, the control architecture modifies directly joint’s trajectories leading to less accurate collisions with the block than hierarchical architectures that try to modify hand’s Cartesian trajectory. For this reason, hierarchical architectures would produce less diverse but more reproducible interactions than control architectures.

In this setup, the intermediate hand’s movement representation is reused by only one higher-level model, but learning intermediate representations should be even more beneficial for exploration in complex learning hierarchies where more than one models are reusing the learned representations.

The next step is to integrate social interaction with a human peer that will give demonstrations on the task space and the hierarchy of models to explore.