Curiosity-driven Exploration of Skill Hierarchies

Sébastien Forestier
INRIA Bordeaux Sud-Ouest
Bordeaux, France
Email: sebastien.forestier@inria.fr

Pierre-Yves Oudeyer
INRIA Bordeaux Sud-Ouest
Bordeaux, France
Email: pierre-yves.oudeyer@inria.fr

***Abstract*—The abstract goes here.**

I. Introduction

Curiosity-driven exploration and developmental trajectories...

The study of the control of manipulation actions in humans has revealed a modular representation of actions either in the cerebral cortex and in the spinal cord with compositionality: an infinite number of movements can be expressed through combination of simple primitives, and generalization: certain neurons (higher in the hierarchy) can represent actions independently of the effectors used [1].

Like a developing child, a developmental robot will have to incrementally explore skills that add up to the hierarchy of previously learned skills throughout its life, with a constraint being the cost and time of experimentation. We will seek to define curiosity-driven hierarchical learning architectures that could reuse the sensorimotor contingencies previously learned and to combine them to explore more efficiently new complex sensorimotor models.

Here, we will rely on the SAGG-Random architecture [2]. This architecture learns a single mapping between a continuous motor space and and continuous sensory (or task) space by randomly selecting goals in the sensory space. In our hierarchy of sensorimotor models, each model will be explored using the SAGG-Random procedure, but it could be replaced by any other exploration architecture.

We want to study and compare our exploration strategies in a high-dimensional continuous environment where primitive actions can be reused to influence different perceptual spaces. We will use a simulated 2D robotic arm that can interact with tools and objects. The robot will have the possibility to use different tools and to ask for the help of a pair to reach certain objects. Section II shows the details of that environment.

Section III explains in detail the SAGG-Random exploration architecture and shows how it behaves on the designed environment.

A. Questions of the study

- Learning a structured hierarchy is more efficient than directly from M to S.?
- Which task should be explored now ?
- How to choose between different means to explore a given space ?
- How can high-level tasks guide the exploration of lower-level ones ?
- How can the system cope with perturbations of some of the forward models ?

B. Related work

Explain [3]–[5].

Different computational models have the possibility to learn skill hierarchies. In finite environments represented by a factored Markov Decision Process [6], an intrinsic motivation towards actions maximizing Dynamic Bayesian Networks' structure has been shown to allow the learning of the environment's structure.

In continuous environments but with discrete actions, Metzen et al. [7] use the framework of options [8] to learn skill hierarchies. An intrinsic motivation rewards positively the novelty of the states encountered and negatively the prediction error of the learned skill model.

The model from Fabisch et al. [9] learns a discrete task space (tasks are called contexts). It uses an intrinsic motivation for learning progress, and a Multi-Armed Bandit algorithm (D-UCB) to choose on which context the agent should train for. The Upper Confidence Bound algorithm chooses between contexts given their estimated learning progress and the uncertainty of these estimations by picking the context with the maximum upper confidence bound. This algorithm supposes a stationary learning progress on each context so the authors use an adaptation [10] to encompass non-stationary learning progress.

In a fully continuous setting, Mugan et al. [11] have developed an algorithm that first learns a qualitative representation of environment states and actions in order to then learn the structure of Dynamic Bayesian Networks representing the temporal contingencies of those states and actions. In order to choose which action to practice, the authors use the Intelligent-Adaptive-Curiosity algorithm [12] where the agent is intrinsically motivated to choose actions that are estimated to yield high prediction error progress.

II. Environment

We simulate a 2D robotic arm using tools to push an object in its environment with the help of an experienced pair. The agent can either try to push objects into boxes on its own, or produce a vocal signal that might engage the experienced pair to help move the object to reach a box.

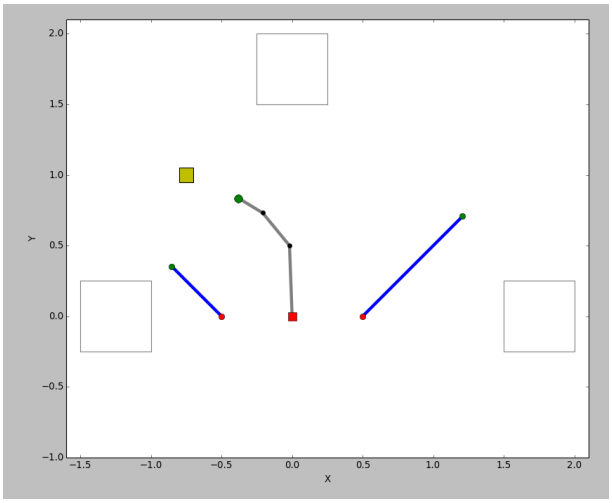


Fig. 1. Play Environment

The environment is designed such that each box is more easily reached by one different mean. Box 1 is best reached with the small stick, box 2 with the long stick, and box 3 with the small stick and the help of the pair.

A simpler version of that environment will also be considered, where only the small stick is provided to the robot (environment E1) instead of both (environment E2).

An iteration consists of the evaluation of a motor command given by the agent which gives sensory information back to him, and finally the environment is resetted to its intial state.

The next sections precisely describe the different items of the environment and their interactions. See Fig. 1 for an exemple of the state of the environment.

A. Robotic arm

The 2D robotic arm has 3 joints plus a gripper located at the end-effector. Each joint can rotate from $-\pi\ rad$ to $\pi\ rad$ around its initial position, mapped to a standard interval of $[-1, 1]$. The length of the 3 parts of the arm are 0.5, 0.3 and 0.2 so the total length of the arm is 1 unit. The initial position of the arm is vertical with each joint at $0\ rad$ and its base is fixed at position $[0, 0]$. The gripper g has 2 possible positions: *open* ($g \geq 0$) and *closed* ($g < 0$) and its initial position is *open* (with $g = 0$). The robotic arm thus has 4 degrees of freedom represented by a vector in $[-1, 1]^4$. A trajectory of the arm will be represented as a sequence of such vectors.

B. Objects and tools

A yellow squared object can be moved into one of the 3 fixed squared boxes. The initial position of the yellow square is $(-0.75, 1)$ and is thus unreachable with directly with the gripper. One of 2 sticks can be grasped in order to reach the object. A small stick of length 0.5 is located on the left of the arm, with initial position $(-0.5, 0)$ and initial angle $\frac{3\pi}{4}$ from the horizontal line. A long stick of length $1.$ is located on the right of the arm, with initial position $(0.5, 0)$ and initial angle $\frac{\pi}{4}$ from the horizontal line as in Fig. 1.

If the gripper closes near the end of one of the sticks (closer than 0.2), it is considered grasped and will follow the gripper's position and the angle (with some noise) of the arm's last part until the gripper opens. The grasped stick will have its angle equal to arm's last part plus a gaussian noise (of size 0.02 for the small sitck and 0.1 for the long one), updated at each step of the movement.

Similarly, if the other end of a stick reaches the yellow squared object (within 0.25), the object will follow the end of the stick. Three boxes are fixed at positions $(-1.25, 0)$, $(0, 1.75)$ and $(-1.75, 0)$ and have size 0.5. At the end of the trial, the object is considered to be in one of the box if its center is in the box.

C. Help from an experienced pair

A pair sitting at the right of the robot will help him put the yellow square into the closer box. It will wait for the robot to move the object on its own, and if the robot also produces the good vocal signal, will move the object to the closest box.

However, as the long stick is long enough to reach the any of the 3 boxes but not the small one, the pair will help the robot only when it will use the small stick. Also, as the pair is sitting on the right side of the robot, it will be unable to help reach box 1 (on the left). The pair will help reach box 2 (on the front) but with a bad precision as it is far, and box 3 (on the right) with a good precision. The pair will put the object at the center of box 2 (with a gaussian noise of size 0.2 on x and y dimensions thus missing the box quite often), only if the object is located near box 2 at a distance from the base of the arm greater than $1.$ and an angle from the horizontal line between 45 and 105. If the angle is between -15 and 45, then the pair moves it towards the center of box 3 with a gaussian noise of size 0.05, thus rarely missing the box.

We simulate a simple vocal signal controled by pitch and intensity between -1 and 1. The pair will engage in helping the robot only if pitch and intensity are sufficiently high (> 0).

D. Motor control

We use Dynamical Movement Primitive [13] to control the arm's movement as this framework permits the production of a diversity of arm's trajectories with few parameters. Each of the 4 arm's degree of freedom (DOF) is controlled by a DMP with a starting and a goal position equal to the rest position of the joint. Each DMP is parameterized by one weight on each of 3 basis functions whose centers are distributed homogeneously throughout the movement. The weights are bounded in the interval $[-200, 200]$ (mapped to the standard interval $[-1, 1]$) which allow each joint to cover its standard interval $[-1, 1]$ during the movement. Each DMP outputs a series of 50 positions that represents a sampling of the trajectory of one joint during the movement.

The arm's movement is thus parameterized by 12 weights, and the static vocal signal by 2 weights. Let M_a be the $12D$ space of arm's commands $[-1, 1]^{12}$, M_v be the $2D$ vocal space, and M the $14D$ global motor space.

E. Sensory feedback

At the end of the movement, the robot gets sensory feedback from the different items of the environment. It gets the trajectory of its hand and gripper, whether the vocal signal engaged the pair or not, the trajectory of the end of the sticks, the end position of the object, and whether the object is in each box and at which distance.

The trajectory of the hand and of the end point of the sticks is the sequence of x and y positions at different time points: steps 12, 25, 37 during the movement of 50 steps. The trajectory of the gripper is a sequence of 1 or -1 depending whether the gripper is open or not. The pair understanding of the vocal signal is represented as 1 if intensity and pitch were correct, -1 otherwise. The sensory information about the boxes is composed of 2 values. The first one tells which box has the object inside (-1 if no one, -0.25 if box 1, 0.25 if box 2, and 0.75 if box 3). The second value is the minimal distance between a box and the object at the end of the movement and after the pair helped.

The sensory information thus contains 6 dimensions for the trajectory of the hand, 1 for the pair help, 6 for the trajectory of the end of each stick, 2 for the end position of the object, and 2 for the boxes. The total sensory space has 34 dimensions if only the small stick is provided to the robot (environment E1), and 40 dimensions if both are (environment E2).

III. Random Goal Babbling

Explain SAGG-Random, NN, show results on environment for GB vs MB. Explain exploration in SAGG-Random, show competence and progress with different measures (and why its important to measure that). The possible perturbation of the standard measure by exploration noise, and a possible fix with the following:

We have

$$||s_g - s|| = ||s_g - s_{NN} + s_{NN} - s_p + s_p - s||$$

where s_{NN} is the nearest neighbor of s_g in the sensorimotor mapping, and is also the sensory prediction of m, and s_p is the sensory prediction of $m + \epsilon$.

When there is no exploration, $\epsilon = 0$, so $s_{NN} = s_p$ and

$$||s_g - s|| = ||s_g - s_{NN} + s_p - s||$$

The expression $s_g - s_{NN}$ corresponds to the novelty with respect to the already reached points, and $s_p - s$ to the forward model's prediction error.

We could also use $||s_g - s_{NN} + s_p - s||$ when $\epsilon > 0$ as the basis of the computation of competence and interest.

IV. Hierarchically structured exploration

A. Experiment 1: Methods

Here we want to compare the learning of a structured hierarchy of SAGG-Random modules with the control condition where one sensorimotor model learns directly from the whole motor space M to the whole sensori space S. For the hierarchical conditions, we randomly choose the module of the hierarchy that will explore at each iteration. We also compare Motor Babbling and Goal Babbling in the hierarchical and in the flat hierarchies. We use hierarchy H1 (Fig. 2), and environmnent E1. Exploration: only the babbling module will add exploration noise.

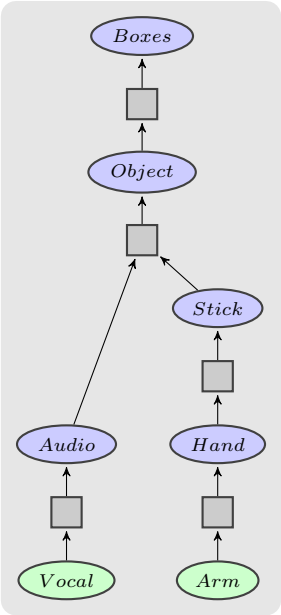


Fig. 2. H1

We measure the number of times the agent manages to grasp the tool, push the object, and reach a box. TODO: exploration of all spaces (hand, tools, object, boxes), competence to reach random goals.

B. Experiment 1: Results

See Fig. 3, 4 and 5.

C. Experiment 1: Discussion

V. Choice of module to explore

A. Experiment 2: Methods

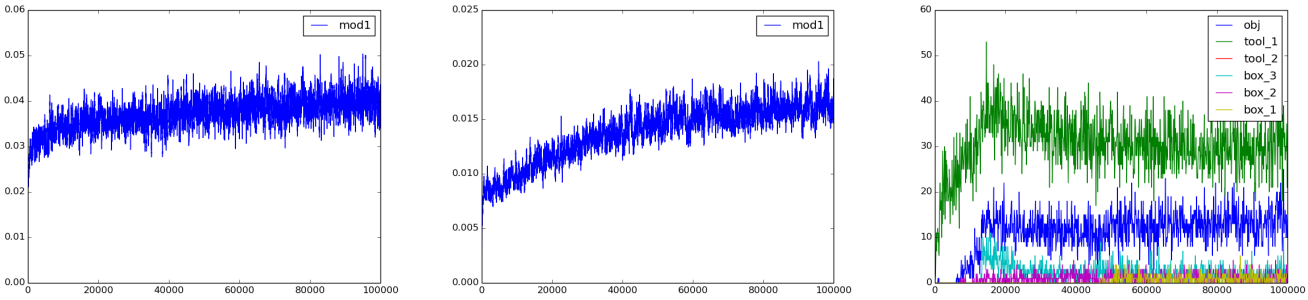
Here we want to compare the different possibilities to choose the module to explore in the hierarchy: choosing the one with the max progress (proba proportional to interest), maximizing with a bias towards lower-level modules, or use ZPDES, with the same hierarchy H1 (Fig. 2), and environmnent E1.

Conditions: Random module (RD), choose with proba proportional to interest (I), the same with bias toward lower levels (IB), or ZPDES. Also try to make all modules add exploration noise (EXPLOA) or only the babbling module (EXPLOB).

We measure the number of times the agent manages to grasp the tool, push the object, and reach a box. TODO: exploration of all spaces (hand, tools, object, boxes), competence to reach random goals.

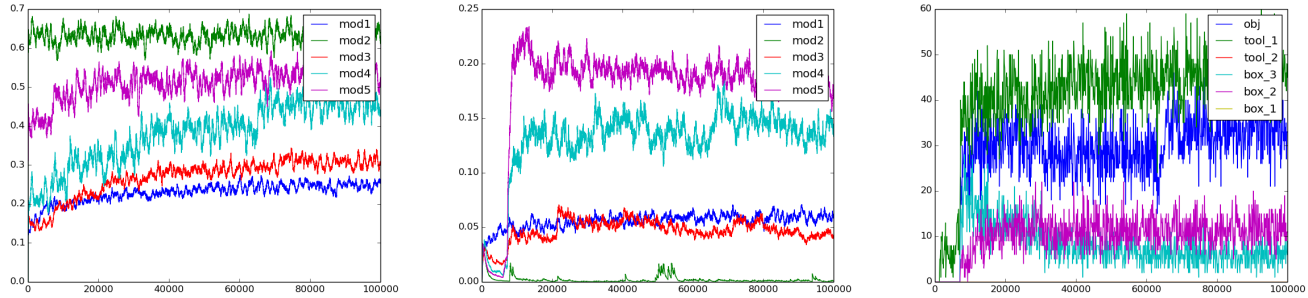
B. Experiment 2: Results

See Fig. 6 and 7.



(a) Competences (b) Interests (c) Events

Fig. 3. Competences, interests and events using H0 with Random Goal Babbling



(a) Competences (b) Interests (c) Events

Fig. 4. Competences, interests and events using H1 with Random Goal Babbling

C. Experiment 2: Discussion

VI. Choice of tool to use

A. Experiment 3: Methods

- Idea: to explain how we can choose between different means (e.g. different tools) using the one with the maximal competence, or maximal progress. We can use the hierarchy (b) of Fig. **??**, in order to have 2 different tools to move the object.
- Conditions: maximize competence vs progress, noise on each tool, reachability (e.g. size) of each tool.
- Features: Hierarchy (b), MAB on all modules, SAGG-Random, NN, No TDD.
- Measures: exploration of intermediate spaces (hand, tools), exploration of top spaces (object). Competence to reach random goals in reachable parts of intermediate and top spaces. Statistics on multiple runs to see regularity/diversity in developmental trajectories.

B. Experiment 3: Results

C. Experiment 3: Discussion

VII. Top-Down Guidance

A. Experiment 4: Methods

- Idea: to compare different possibilities of Top-Down Guidance, with hierarchy (c) of Fig. **??** in order to have TD guidance at different levels: arm with one higher model, hand with 2 higher models, tool1 with one higher model, or tool2 with 2 higher models.
- Conditions: No TDD, Just add noise to motor command of each module (pb: interferes with competence estimation) vs explore n points and returns the best (warning: exponential) vs only one layer below the babbling module explores n points.
- Features: Hierarchy (c), MAB on all modules with bias (or no bias?), SAGG-Random, NN.
- Measures: exploration of intermediate spaces (hands, tools), exploration of top spaces (objects). Competence to reach random goals in reachable parts of intermediate and top spaces. Statistics on multiple runs to see regularity/diversity in developmental trajectories.

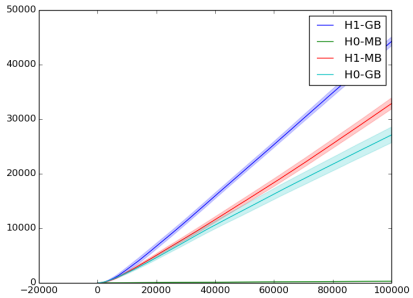
B. Experiment 4: Results

C. Experiment 4: Discussion

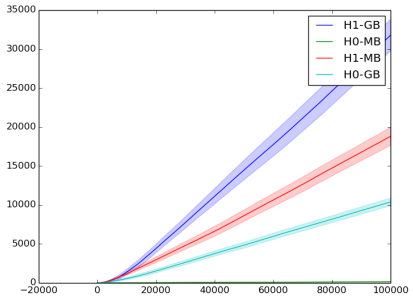
VIII. Robustness to perturbations

A. Experiment 5: Methods

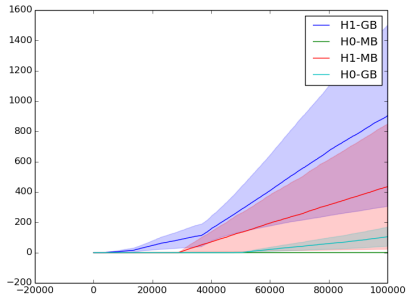
- Idea: to apply perturbations to one of the possible forward models, either blocking, shifting or randomizing one dimension. We can use the hierarchy (d) of Fig. **??** in order to see an adaptation at 2 levels: the use of one



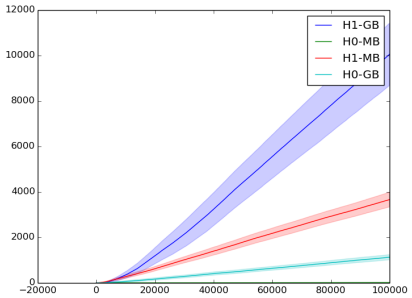
(a) Tool



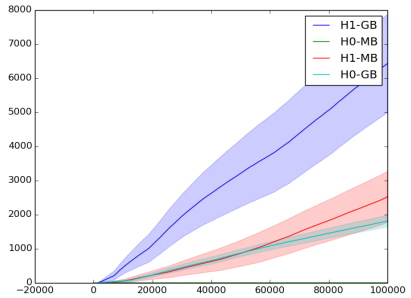
(b) Object



(c) Box 1

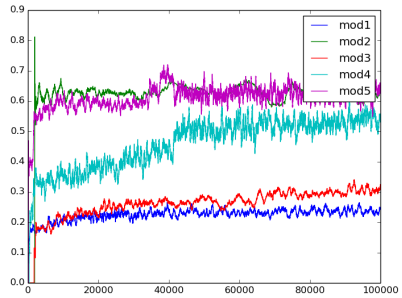


(d) Box 2

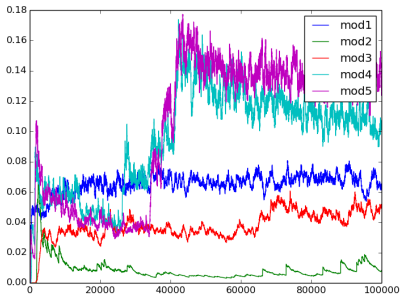


(e) Box 3

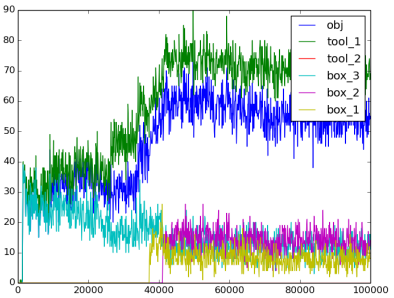
Fig. 5. Number of touch of tool, object and boxes for each 100 iterations' bin.



(a) Competences



(b) Interests



(c) Events

Fig. 6. Competences, interests and events using H1 with I-EXPLOB

hand to use one tool, the use of both tools to move the object even if only one is perturbated.

- Conditions: No perturbations, which model is perturbated (arm, tool), type of perturbation (blocking, shifting, random).
- Features: Hierarchy (d), MAB on all modules, SAGG-Random, NN, best TDD.
- Measures: exploration of intermediate spaces (hands, tools) and top spaces (objects) before and after perturbations. Competence to reach random goals in reachable parts of intermediate and top spaces before and after perturbations. Statistics on multiple runs to see regularity/diversity in developmental trajectories.

B. Experiment 5: Results

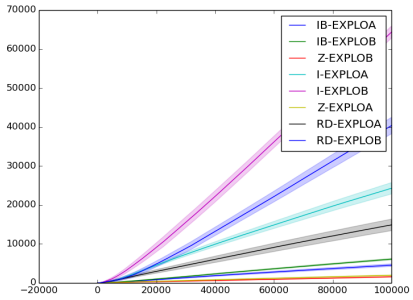
C. Experiment 5: Discussion

IX. General Discussion

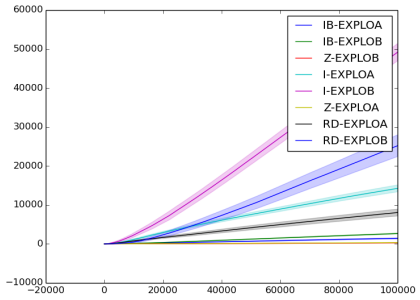
References

[1] A. Cangelosi, G. Metta, G. Sagerer, S. Nolfi, C. Nehaniv, K. Fischer, J. Tani, T. Belpaeme, G. Sandini, F. Nori *et al.*, "Integration of action and language knowledge: A roadmap for developmental robotics," *Autonomous Mental Development, IEEE Transactions on*, vol. 2, no. 3, pp. 167–195, 2010.

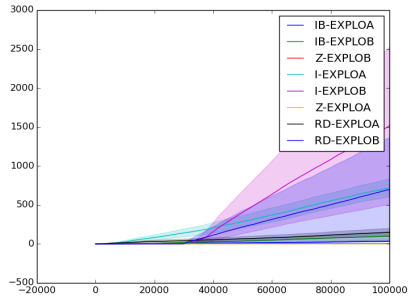
[2] A. Baranes and P.-Y. Oudeyer, "Intrinsically motivated goal exploration for active motor learning in robots: A case study," in *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*. IEEE, 2010, pp. 1766–1773.



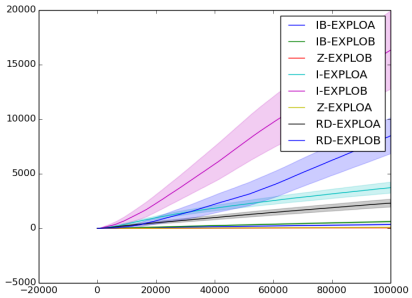
(a) Tool



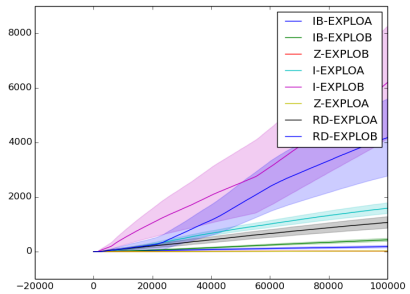
(b) Object



(c) Box 1



(d) Box 2



(e) Box 3

Fig. 7. Number of touch of tool, object and boxes for each 100 iterations' bin.

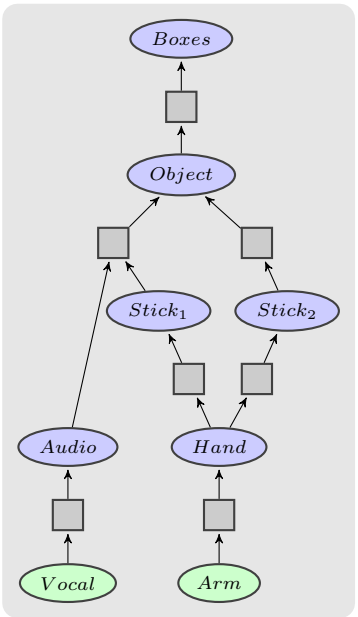


Fig. 8. H2

[3] E. Ugur and J. Piater, "Emergent structuring of interdependent affordance learning tasks," in *Development and Learning and Epigenetic Robotics (ICDL-Epirob), 2014 Joint IEEE International Conferences on*. IEEE, 2014, pp. 489–494.

[4] E. Ugur, Y. Nagai, E. Sahin, and E. Oztop, "Staged development of robot skills: Behavior formation, affordance learning and imitation with motionese," *Autonomous Mental Development, IEEE Transactions on*, vol. 7, no. 2, pp. 119–139, June 2015.

[5] M. Schmerling, G. Schillaci, and V. V. Hafner, "Goal-directed learning of hand-eye coordination in a humanoid robot," in *5th Joint IEEE International Conferences on Development and Learning and on Epigenetic Robotics (ICDL-Epirob)*, 2015.

[6] C. M. Vigorito and A. G. Barto, "Intrinsically motivated hierarchical skill learning in structured environments," *Autonomous Mental Development, IEEE Transactions on*, vol. 2, no. 2, pp. 132–143, 2010.

[7] J. H. Metzen and F. Kirchner, "Incremental learning of skill collections based on intrinsic motivation," *Frontiers in Neurorobotics*, vol. 7, 2013.

[8] R. S. Sutton, D. Precup, and S. Singh, "Between mdps and semi-mdps: A framework for temporal abstraction in reinforcement learning," *Artificial intelligence*, vol. 112, no. 1, pp. 181–211, 1999.

[9] A. Fabisch and J. H. Metzen, "Active contextual policy search," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 3371–3399, 2014.

[10] L. Kocsis and C. Szepesvári, "Discounted ucb." 2nd PASCAL Challenges Workshop, 2006.

[11] J. Mugan and B. Kuipers, "Autonomously learning an action hierarchy using a learned qualitative state representation," 2009.

[12] P.-Y. Oudeyer, F. Kaplan, and V. V. Hafner, "Intrinsic Motivation Systems for Autonomous Mental Development," *IEEE Transactions on Evolutionary Computation*, vol. 11, no. 2, pp. 265–286, Apr. 2007.

[13] A. J. Ijspeert, J. Nakanishi, H. Hoffmann, P. Pastor, and S. Schaal, "Dynamical movement primitives: learning attractor models for motor behaviors," *Neural computation*, vol. 25, no. 2, pp. 328–373, 2013.