Curiosity-Driven Development of Tool Use Precursors: a Robotic Model

Sébastien Forestier (sebastien.forestier@inria.fr)
Inria Bordeaux Sud-Ouest and Ensta Paristech, France

Pierre-Yves Oudeyer (pierre-yves.oudeyer@inria.fr)
Inria Bordeaux Sud-Ouest and Ensta Paristech, France

Abstract

This is the abstract.
This is the abstract.
This is the abstract.
This is the abstract.
This is the abstract.
This is the abstract.
This is the abstract.

Keywords: curiosity-driven learning; tool use; goal babbling; overlapping waves; developmental trajectory; hierarchical skill learning; HACOB model

Introduction

The understanding of tool use development in young children is one of the key question for the more general understanding of the ontogeny of human cognition. Indeed, a series of abilities are progressively developed from the simplest reaching movements of the arms through more dexterous manipulation of a spoon, towards advanced control of multiple interacting objects. The latter shows an understanding of shapes, forces and other physical properties that can be recruited for mental transformations and planning operations which are pillars of human cognition. Children's development has first been described as staircase-like successive stages in which all children go through (Piaget, Cook, & Norton, 1952). More recently, different views were developed describing variability among children's developmental paths. Siegler's overlapping waves theory (Siegler, 1996) also describes variability in a child's set of current methods to solve a problem, and documents their evolution. In particular, the development of tool use precursors can be described as three consecutive and overlapping stages of behaviours (Guerin, Kruger, & Kraft, 2013): body babbling, behaviours with a single object, and behaviours with several interacting objects. A study of free play (Zelazo & Kearsley, 1980) shows that at $9\frac{1}{2}$ months play is mostly composed of tactile examination, waving, banging and mouthing of a single object, but that simple relational acts of banging two objects together is already common. Later at $13\frac{1}{2}$ months, the study reveals that most children instead prefer to explore the relationships among objects, but still show behaviors of the previous phase. Furthermore, they show that this overlapping phases pattern averaged across children is also present in a longitudinal study of a single child.

In this paper we focus on the study of this progression between overlapping phases of behaviours with objects in a robotic model and in particular on the use of concurrent methods to solve a problem. We hypothesize that several mechanisms play a role in the structure of this behavioural progression and in particular 1) the intrinsic motivation to explore as a self-regulation of the learning growth of complexity, and 2) the structure of the representation used to encode sensorimotor experience. Active learning is a paradigm where the learning agent chooses which actions to perform in its environment. To make this choice the agent can optimize some extrinsic rewards given by the environment, as in standard Reinfocement Learning tasks. These extrinsic rewards are not suited for an open-ended autonomous exploration of the environment that could reflect children learning by free play. Based on infant's developmental psychology, different types of intrinsic motivations have been studied (Santucci, Baldassarre, & Mirolli, 2013). One way to define an intrinsic motivation is to autonomously search for situations where the learning progress is high, so as to avoid unreachable or unlearnable situations. Computational models have shown that developmental trajectories could emerge from the curiosity-driven learning of sensorimotor mappings, in very different settings. In the Playground Experiment (Oudeyer, 2007), a quadruped robot learned how to use its motor primitives to interact with the items of an infant play mat and a robot peer. Also, in a study of the self-organization of vocalizations (Moulin-Frier, Nguyen, & Oudeyer, 2014), an agent had to learn how to use a vocal synthesizer by self-exploration or with the help of humans' demonstrations of phonetic items. This model reproduces accurately major phases of infant vocal development until 6 months. In both studies, developmental trajectories of increasing complexity are emerging from learning, with both regularities in developmental steps and diversity. The diversity comes from different mechanisms: stochasticity in the algorithms, variability in the environment, and the multiples attractors of the dynamic learning system. In existing models, the agent learns only one mapping that relates a motor space to a single task space. However, in the perspective of an open-ended development of reusable skills, and specifically in the development of tool use, multiple interdependent and hierarchically organized task spaces should be available to the agent as for instance using a tool to act upon an object could make use of previously explored parameterized interaction with the tool.

We study aspects of those hypothesis leveraging previous models of curiosity-driven learning and extending them to the active exploration of hierarchical sensorimotor and task spaces. We define hierarchies of sensorimotor models that structure the sensory space to reflect the interaction of the different items of the environment. In such hierarchies of models to explore, different exploration choices are available to the agent at each learning iteration: which model to explore, and how to explore that model. The problem of finding an efficient active choice strategy is an instance of strategic learning (Nguyen & Oudeyer, 2012), where different outcomes and strategies are available and the agent has to learn which strategies are useful for which outcomes. This can be viewed as a generalization of active learning methods in machine learning. We define the HACOB (Hierarchical Active Curiosity-driven mOdel Babbling) architecture and compare several possible strategies to study the role of active learning and hierarchical representation in the structuration of developmental trajectories. We compare the different learning conditions in a 2D environment where a simulated arm with three joints plus a gripper can grab one of two available tools to move an out-of-reach object. We measure the different phases of behaviour during exploration and compare the different structures of behavioural evolution.

To our knowledge, HACOB is the first model of the curiosity-driven development of tool use, and the first to show the autonomous emergence of overlapping phases of behaviour in the development of object manipulation skills. Here we define tool use as the ability to perform different effects on an object with the help of an intermediate object, using some sort of learned inverse mapping. Our model is also the first to allow the intrinsically-motivated parallel exploration of different tools to reach a same goal, in line with Siegler's overlapping waves theory. Other models predefine successive learning phases in the learning of object affordances (Ugur, Nagai, Sahin, & Oztop, 2015), or do not study the role of intrinsic motivation in the learning of tool affordances (Stoytchev, 2005).

However, here we do not study some other important factors in the development of tool use. We consider the hierarchy of sensorimotor models to learn as given to the agent and do not study its autonomous building and evolution. Also, social guidance with imitation and mimicry is of central importance for the development of tool use in infants but we do not address the question of its modeling in this paper nor the interplay and tradeoff between social learning and self-exploration. Another important feature of young infants' learning of tool use is the need to adapt to a developing body and to the maturation of vision during ontogenetic development, but here we consider motor control and sensory perception steady over the simulated learning time span.

Along with this paper we provide open-source Python code[1] with iPython/Jupyter notebooks that explain how to reproduce the experiments and analysis.

Methods

Environment

We simulate a 2D robotic arm that can grasp tools that can be used to move an object into different boxes in the environment. In each trial, the agent executes a motor trajectory, we evaluate its consequences on the sensory dimensions and we give him this sensory feedback. Finally the arm, tools and objects are resetted to their intial state. The next sections precisely describe the items of the environment and their interactions. See Fig.1 for an exemple state of the environment.

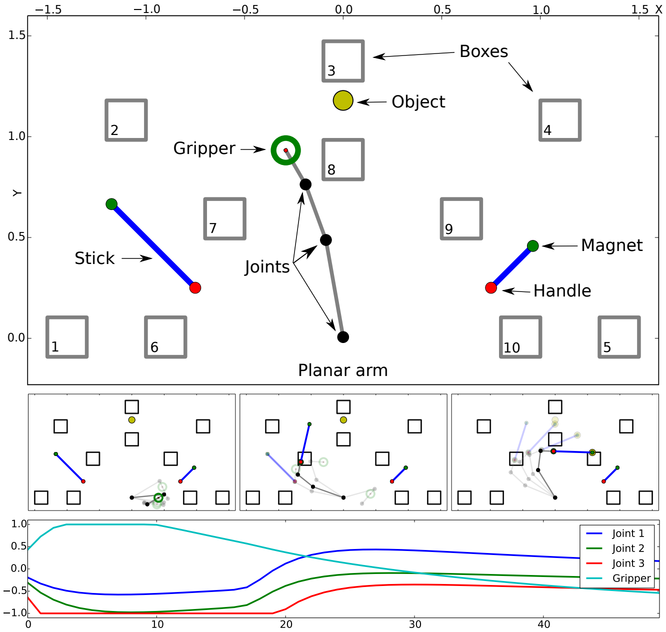


Figure 1: Top: a state of the environment. Middle: position of the arm at time steps 17, 33 and 50, with some intermediate positions, along the 50 steps movement. Bottom: trajectory of each of the four virtual motors, generated by a DMP.

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 π *rad* around its resting position, mapped to a standard interval of $[-1,1]$. The length of the 3 segments of the arm are 0.5, 0.3 and 0.2 so the length of the arm is 1 unit. The resting position of the arm is vertical with joints 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 resting position is *open* (with $g = 0$). The robotic arm 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.

Motor control We use Dynamical Movement Primitives (Ijspeert, Nakanishi, Hoffmann, Pastor, & Schaal, 2013) 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 degrees-of-freedom (DOF) is controlled by a DMP starting at the rest position of the joint. Each DMP is parameterized by one weight on each of 2 basis functions and one weight specifying the end position of the movement. The weights are bounded in the interval $[-1,1]$ and allow each joint to fairly cover the 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

[1] Source code and notebooks available as a Github repository at `https://github.com/sebastien-forestier/CogSci2016`

movement. The arm's movement is thus parameterized with 12 weights, represented by the motor space $M = [-1,1]^{12}$.

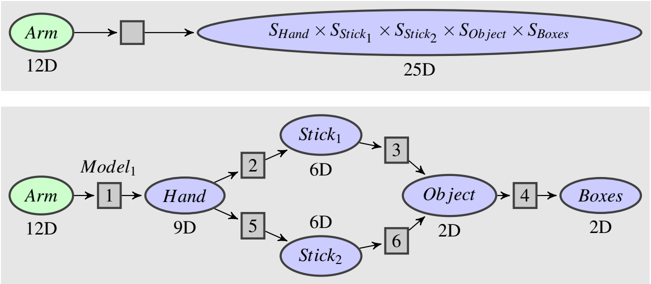
Objects and tools Two sticks can be grasped by the handle side in order to catch an out-of-reached object. A small stick of length 0.3 is located at position $(0.75, 0.25)$ with initial angle $\frac{\pi}{4}$ from the horizontal line. A long stick of length 0.6 is located at position $(-0.75, 0.25)$ with initial angle $\frac{3\pi}{4}$ as in Fig. 1. A yellow sphere can be caught by the magnetic side of one of the two sticks, moved and possibly placed into one of ten fixed squared boxes. The initial position of the sphere is $(0, 1.2)$ and is thus unreachable directly with the gripper. If the gripper is closed near the handle of one stick (closer than 0.2), it is considered grasped and will follow the gripper's position and the angle of the arm's last segment until the gripper opens. Similarly, if the other end of a stick reaches the sphere (within 0.1), the sphere will follow the end of the stick. The ten boxes (identified from 1 to 10) are static and have size 0.2. Boxes 1 to 5 can only be reached with the long stick, and the other five boxes can be reached with both sticks. At the end of the movement, the object is considered to be in one of the box if its center is in the box.

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 the gripper (S_{Hand}, 9D), the trajectory of the end of the sticks (S_{Stick_1},$6D$ and S_{Stick_2},$6D$), the position of the object at the end of the mouvement (S_{Object},$2D$), and whether the object is in a box at the end of the mouvement and the distance between the object and the nearest box (S_{Boxes},$2D$). The trajectory of the gripper is represented as the x and y positions and the aperture (1 or -1) of the gripper at 3 time points: steps 17, 33, 50 during the movement of 50 steps (9D). Similarly, the trajectories of the end points of the sticks are 3-points sequences of x and y positions (6D for each stick). The agent receives the identifier (from 1 to 10) of the reached box if one of them has been reached by the sphere, 0 otherwise. He also gets the minimal distance of the object (at the end of the movement) to the center of a box, even if none have been reached. The sensory information thus contains 9 values for the trajectory of the gripper, 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 S has 25 dimensions.

Learning architectures

The problem settings for the learning agent is to explore its sensorimotor space and collect data so as to generate a diversity of effects and to learn an inverse model to be able to reproduce those effects. In this section we describe the learning architectures that we will compare in the experiments.

Flat architectures We define a flat architecture as directly learning a mapping between the motor space M (12D) and the sensory space S (25D). To do so, the agent needs a sensorimotor model that learns the mapping and provides inverse inference of a probable m to reach a given s. The sensorimotor model stores new information of the form (m,s) with $m \in M$ being the experimented motor parameters and $s \in S$ the associated sensory feedback. It computes the inverse inference with the nearest neighbor algorithm: it gets the motor part of the nearest neighbor of the given s in S, and adds exploration noise (gaussian with $\sigma = 0.01$) to allow new motor parameters to be explored.



Arm
12D
$S_{Hand} \times S_{Stick_1} \times S_{Stick_2} \times S_{Object} \times S_{Boxes}$
25D
$Model_1$
1
Hand
9D
2
5
Stick$_1$
6D
6D
Stick$_2$
3
6
Object
2D
4
Boxes
2D

Figure 2: Architectures. Top: Flat. Bottom: Hierarchical.

The agent also needs an interest model that chooses goals in the sensory space. The control condition is a random motor babbling condition (F-RmB) that randomly chooses new motor parameters m to try at each iteration. In the other conditions, the agent performs Goal Babbling, a method by which it self-generates a goal in the sensory space at each iteration and trying to reach it. To generate those goals, different strategies have been studied (Baranes & Oudeyer, 2013). It was shown that estimating the learning progress in different regions of the sensory space and generating the goals where the progress is high leads to a fast learning. However, this idea cannot be applied in a 25D sensory space as a learning progress signal cannot be properly estimated in this volume. Thus we use a simpler random generation of goals in the sensory space as an interest model in the flat random goal babbling condition (F-RGB), which was nevertheless proven to be highly efficient in complex sensorimotor spaces (Rolf, Steil, & Gienger, 2010). We use the Explauto autonomous exploration library (Moulin-Frier, Rouanet, Oudeyer, & others, 2014) to easily define those sensorimotor and interest models.

Hierarchical architectures The 25D sensory space can be structured to reflect the interaction of the different items of the environment. Indeed, the arm motor position influence the gripper, which influence one of the tools (but not both at the same time), which influence the position of the object and the filling of the boxes. We thus study here learning architectures that could make use of this sensorimotor structure, and we call them hierarchical. Those architecture learn 6 models at the same time (see Fig. 2: gray squares are models). Each of those models functions in the same way as the random goal babbling flat architecture (F-RGB). Each model has a motor space (e.g. motor space of model 2 is S_{Hand}), a sensory space (respectively S_{Stick_1}, see arrows in Fig. 2, and can choose goals randomly in this sensory space. At each iteration, the architecture first have to choose the model in which to choose a goal, a procedure that we call Model Babbling. Once a model is chosen, it finds a goal in its sensory space, and infer motor parameters (that can be in the sensory space of a lower-level model) to reach that goal. Then, it passes those parameters as a goal to be reached by a lower-level model, which similarly infers motor parameters and passes those ones until the actual *Arm* motor space gets parameters to try in the environment (with the same exploration noise as in Flat architectures). Model 4 have also to choose which lower-level model to use in order to reach an object end position s_o in S_{Object}, as two models (3 and 6) have S_{Object} as sensory space. Model 4 chooses the tool that have allowed to reached s_o as close as possible in the past, e.g. if model 3 has in its history a reached sensory point s closer to s_o than any reached point with model 6, then model 3 is chosen. Finally, when motor parameters m have been tested in the environment and feedback s received, the mappings of all models are updated, but if model 4 have chosen a given tool, then the mapping of the other tool is not.

Random vs Active Model Babbling A first condition is to randomly choose the model that will find a goal, this is Random Model Babbling (H-RMB). The problem of Model Babbling is an instance of strategic learning (Nguyen & Oudeyer, 2012), where different outcomes and strategies to learn them are available and the agent learns which strategies are useful for which outcomes. In that paper, they show that an active choice of the outcomes and strategies based on the learning progress on each of them increase learning efficiency compared to random choice. To develop active learning strategies, we first define a measure of learning progress for each of the 6 models. When a model has been chosen to babble, it draws a random goal s_g, and finds motor parameters m to reach it using the lower-level models. The actual outcome s in the sensory space of the model, associated to m might be very different from s_g as this goal might be unreachable, or because lower-level models are not mature enough for that goal. We define the competence associated to a goal s_g as minus the distance between the goal and the reached point, divided by the maximal distance in this space, to scale this measure across different spaces:

$$C(s_g) = -\frac{||s_g - s||}{max_{s'}||s' - s||} \quad (1)$$

and the interest $I(s_g)$ associated to this goal as

$$I(s_g) = |C(s_g) - mean_{kNN} C(s)| \quad (2)$$

the absolute difference between $C(s_g)$ and the mean competence of the ($k = 20$) nearest previous goals. The interest of a model is initialized at 0 and updated to follow the interest of the goals (with rate $n = 200$):

$$I_{model} = \frac{n-1}{n} I_{model} + \frac{1}{n} I(s_g) \quad (3)$$

In condition H-P-AMB, the choice of model is probabilistic and proportional to their interest (but with $\varepsilon = 10\%$ of random choice). In condition H-GR-AMB, the choice of model is greedy (model with maximum interest) but also with $\varepsilon = 10\%$ of random choice. Finally, condition H-P-AMB-ATC is the same as H-P-AMB but the choice of the tool to use (model 3 or 6) is with probabilities proportional to the interest of the two models, instead of being based on the more competent tool for the given object goal position. We call HACOB this Hierarchical Active Curiosity-driven mOdel Babbling algorithmic architecture with the algorithms H-P-AMB and H-P-AMB-ATC being two variants of the architecture.

Results

We perform 100 simulations per condition. Each simulation starts with 100 iterations of motor babbling and then runs for 100000 iterations of the condition. In this section we provide results for different types of measures. We define a behavioural measure to categorize the types of behaviours with objects and study the structure of their evolution. We also give a measure of the total exploration of the different spaces during simulations. Finally, we compare the structure of tool choice made to reach object goal position during exploration in the two conditions for which only this choice differs.

Fig. 3 shows details about one trial of the condition H-P-AMB. We can see the interest of each model during the whole experiment. The interests of models 2 and 5 increase abruptly once the arm succeeded in grabbing the corresponding stick. Following that, the interests of models 3 and 6 increase abruptly once the object has been touched by the corresponding stick. An exemple of exploration of the 2D space of the object is also provided in Fig. 3(b) for the same condition.

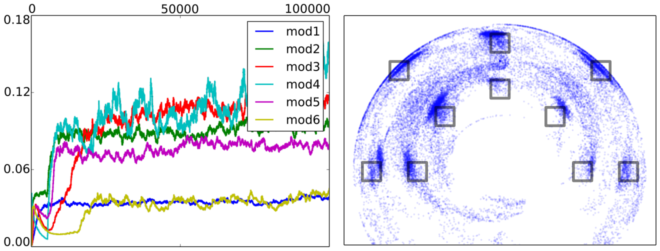


Figure 3: Condition H-P-AMB. Left: Interests of each model. Right: Exploration of the object space: each dot is one point reached with the object at the end of a movement.

Structure of the evolution of behaviours We provide a measure of the different types of behaviours with the sticks and the object during exploration. We categorize the behaviours into three types. In the first category (*hand*) are mouvements of the arm that did not grab any stick and thus did not move the out-of-reached object. The second category (*stick*) are mouvements that did grab one of the two sticks but did not touch the object with it. The third category (*object*) contains the mouvements where both a stick was grabbed and the object was moved by the stick. Fig. 4 shows the prototypical evolution of the proportion of the three categories of behaviour along the 100000 iterations for conditions H-GR-AMB and H-P-AMB. We performed a more detailed analysis by counting the trials where the evolution of the three types of behaviours were similar to the one of Fig. 4(b). A structure was considered similar if it validated each of the following criteria: behaviours of categories *stick* and *object* increase quikly from 0 to more than 10% (potentially after an initial phase with a steady low value), and are followed by a plateau (steady curve with small slope) with no abrupt changes, and behaviours of category *object* start to raise at least 1000 iterations after *stick* started to raise. See Table 1 for the results of this analysis per condition. Also, the median number of abrupt changes across trials for each condition are reported in the same table (as the sum of steady changes of more than 10% in the three behaviours). with a significant difference between condition H-GR-AMB and the others (Mann-Whitney U tests, $p < 0.0001$).

Table 1: Behavioural analysis

Condition	Number of Trials validating criteria	Median number of Abrupt changes
F-RmB	0	0
F-RGB	0	1
H-RMB	60	2
H-P-AMB	70	2
H-GR-AMB	7	6
H-P-AMB-ATC	79	1

Exploration efficiency Also, for each condition we measure the total exploration of the different sensory spaces during training. The exploration of the hand, sticks and object spaces is defined as the number of reached cells in a 100×100 discretization of the (X,Y) space of the position at timestep 50 (end of movement). The exploration of the boxes is the number of boxes that have been filled with the object during training. Fig. 5 shows the total exploration of the different sensory spaces for each condition. We provide statistical Mann-Whitney U test results of comparisons of the exploration in different pairs of conditions. Firstly, the Motor Babbling condition (F-RmB) have more explored S_{Hand} and less S_{Object} and S_{Boxes} compared to the other conditions ($p < 0.0001$). Then, F-RGB explores all spaces less than H-RMB condition ($p < 0.01$). H-RMB explores more S_{Hand} ($p < 0.001$) and less S_{Object} ($p < 0.05$) than H-P-AMB. Also, H-GR-AMB shows lower exploration all spaces than H-P-AMB ($p < 0.01$). Condition H-P-AMB-ATC explores more S_{Stick_2} ($p < 0.05$) than condition H-P-AMB, and difference is not significant in other spaces.

Structure of tool choice Fig. 6 shows a comparison of the choice of tool to reach a given object goal position in the conditions H-P-AMB and H-P-AMB-ATC. In those conditions, model 4 learns a mapping between S_{Object} and S_{Boxes}. When this model is babbling, it chooses a random goal $s_b \in S_{Boxes}$ and infers the best object position s_o to reach s_b. To reach s_o, one of the tools ($Stick_1$ with model 3 or $Stick_2$ with model 6) is chosen. We plot all those choices model 4 made during exploration, at position s_o on a 2D space, with color blue if $Stick_1$ was chosen and red if $Stick_2$ was chosen, in one figure for each of the two conditions. We can see two very different choice structures. However, goal that can be reached with both tools are more often chosen to be explored with the long stick in the interest-based choice of condition H-P-AMB-ATC than in competence-based choice of condition H-P-AMB.

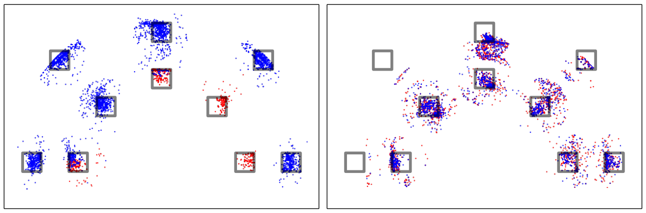


Figure 6: Chosen tool depending on object goal position. Blue points: long stick choice. Red points: small stick choice. Left: condition H-P-AMB, strong boundaries between tool choice regions. Right: condition H-P-AMB-ATC, parallel exploration of both tools for the same goals.

Discussion

Structure of the evolution of behaviours Results show different structures of behaviour evolution in the different conditions. H-GR-AMB shows successive behavioural steps with abrupt changes. Random model babbling and active model babbling both show overlapping waves structure in the evolution of the three behaviours, but random model babbling explores slightly less the object position. In this setup, the difference is more on the cognitive or intentional level, as active model babbling monitors the progress on each model whereas random model babbling do not, than on a quantitative level, because here all models are still useful to explore. However, in others setups where some tasks are learned much faster than others and where at some point it become useless to explore a mastered task, then active model babbling would make a key difference on a quantitative exploration point of view, and on the structure of the evolution of the measured behaviours.

Variability of strategies to reach goals Different structure of tool choice: overlapping waves of strategies to fulfill given goals in condition H-P-AMB-ATC.

References

Baranes, A., & Oudeyer, P.-Y. (2013). Active learning of inverse models with intrinsically motivated goal exploration in robots. *Robotics and Autonomous Systems*, *61*(1).

Guerin, F., Kruger, N., & Kraft, D. (2013). A survey of the ontogeny of tool use: from sensorimotor experience to planning. *Autonomous Mental Development, IEEE Transactions on*, *5*(1).

Ijspeert, A. J., Nakanishi, J., Hoffmann, H., Pastor, P., & Schaal, S. (2013). Dynamical movement primitives: learn-

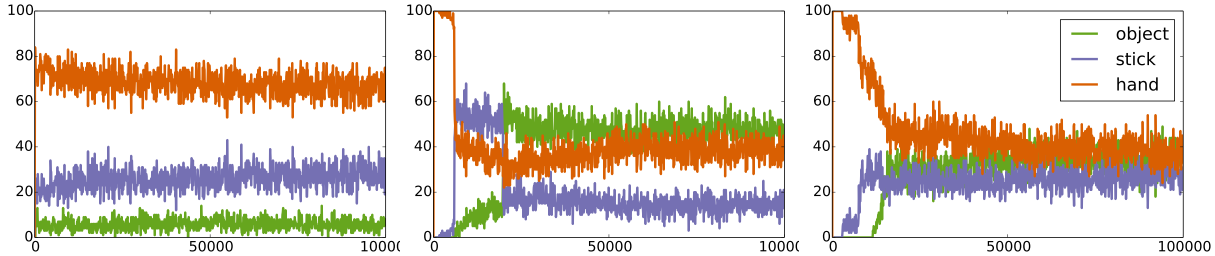
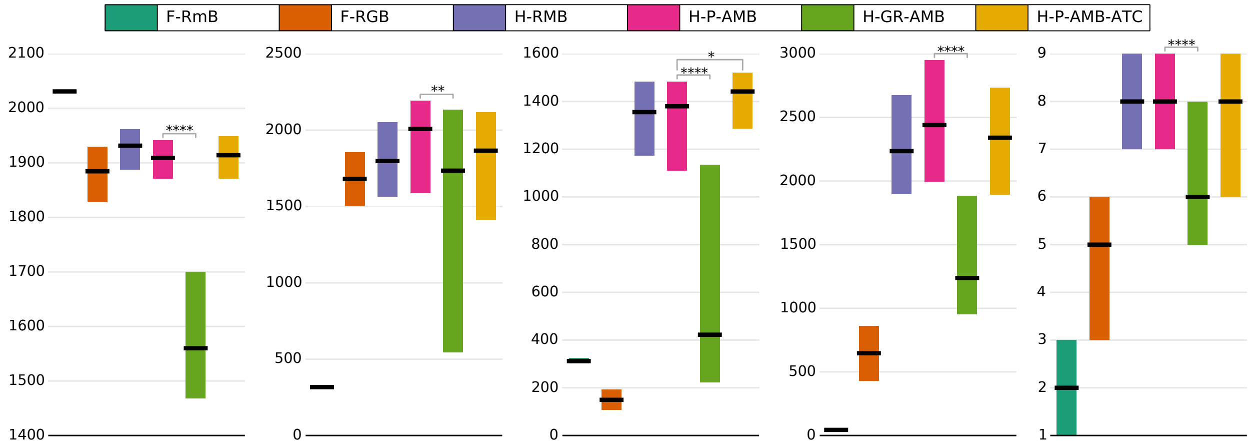


Figure 4: Behavioural phases typical exemples. Left: F-RGB. Middle: H-GR-AMB, Right: H-P-AMB. Only hierarchical active condition H-P-AMB shows overlapping phases of behaviour similar to infant development.



(a) *Hand* (b) *Stick*$_1$ (c) *Stick*$_2$ (d) *Object* (e) *Boxes*

Figure 5: Exploration of sensory spaces. Box plots show for each condition and space the median and quartiles of the 100 trials.

ing attractor models for motor behaviors. *Neural computation*, *25*(2).

Moulin-Frier, C., Nguyen, S. M., & Oudeyer, P.-Y. (2014). Self-organization of early vocal development in infants and machines: the role of intrinsic motivation. *Frontiers in Psychology*, *4*.

Moulin-Frier, C., Rouanet, P., Oudeyer, P.-Y., & others. (2014). Explauto: an open-source Python library to study autonomous exploration in developmental robotics. In *ICDL-Epirob-International Conference on Development and Learning, Epirob.*

Nguyen, S., & Oudeyer, P.-Y. (2012). Active choice of teachers, learning strategies and goals for a socially guided intrinsic motivation learner. *Paladyn*, *3*(3).

Oudeyer, P.-Y. (2007). What is intrinsic motivation? A typology of computational approaches. *Frontiers in Neurorobotics*, *1*.

Piaget, J., Cook, M., & Norton, W. (1952). *The origins of intelligence in children* (Vol. 8) (No. 5). International Universities Press New York.

Rolf, M., Steil, J., & Gienger, M. (2010). Goal babbling permits direct learning of inverse kinematics. *Autonomous Mental Development, IEEE Transactions on*, *2*(3).

Santucci, V. G., Baldassarre, G., & Mirolli, M. (2013). Which is the best intrinsic motivation signal for learning multiple skills? *Frontiers in Neurorobotics*, *7*.

Siegler, R. S. (1996). *Emerging minds: The process of change in children's thinking*. Oxford University Press.

Stoytchev, A. (2005). Behavior-grounded representation of tool affordances. In *Proceedings of the 2005 ieee international conference on robotics and automation.*

Ugur, E., Nagai, Y., Sahin, E., & Oztop, E. (2015). Staged development of robot skills: Behavior formation, affordance learning and imitation with motionese. *IEEE Transactions on Autonomous Mental Development*, *7*(2).

Zelazo, P. R., & Kearsley, R. B. (1980). The emergence of functional play in infants: Evidence for a major cognitive transition. *Journal of Applied Developmental Psychology*, *1*(2).