Strategic and interactive learning of a hierarchical set of tasks by the Poppy humanoid robot

Nicolas Duminy, Sao Mai Nguyen, Dominique Duhaut

Strategic and interactive learning of a hierarchical set of tasks by the Poppy humanoid robot

Nicolas Duminy[1] Sao Mai Nguyen[2] Dominique Duhaut[1]

***Abstract*— We present an active learning architecture that allows a robot to actively learn which data collection strategy is most efficient for acquiring motor skills to achieve multiple outcomes, and generalise over its experience to achieve new outcomes for cumulative learning. In the present work, we consider the learning of tasks that are hierarchically organised, interrelated and more and more difficult.**

This paper proposes an algorithmic architecture, called Socially Guided Intrinsic Motivation with Active Choice of Task and Strategy for Cumulative Learning (SGIM-ACTSCL). It relies on hierarchical active decisions of what and how to learn, driven by empirical evaluation of learning progress for each learning strategy. Our learning agent uses both interactive learning and autonomous goal-babbling. It actively decides at the same time, which tasks to focus on, when to explore autonomously, and when and what to request for social guidance. We present experimental results on the physical humanoid robot Poppy that learns different types of motor skills, encoded by Dynamic Movement Primitives, in order to use a tablet (Fig. 1). We show that SGIM-ACTSCL learns significantly more efficiently than other algorithms. Moreover, it automatically organises its learning process focusing on easy tasks first, and difficult tasks afterwards. It coherently selects the best strategy with respect to the chosen outcome, manages to learn to associate the teacher with his competence domain in order to actively request social guidance for the appropriate tasks.

I. Strategic Interactive Learning for Life-long Learning of Multiple Tasks

In open environments, service robots would carry out not only one but multiple tasks, which is an ongoing challenge for robotic learning. These tasks can be completely independent. In other cases, they can be interrelated with simple and complex tasks, the latter needing the skill of the former. We wish to explore the possibilities for a robot to **learn cumulatively motor policies for numerous interrelated tasks** from a developmental robotics perspective [1].

Grounding our studies on behavioural psychology theories of intrinsic motivation and interactive learning, we base our work on active motor skill learning of multiple task, socially guided learning and strategic learning in order to devise an algorithm for actively learning to tackle the challenge of learning multiple interrelated tasks.

A. Active motor skill learning of multiple tasks

Approaches to robot skill learning have been widely studied for learning forward and inverse models, mapping a space of parameterised motor policies with a space of parameterised outcomes. Techniques based on optimisation and reinforcement learning have been explored in [2], [3], [4]. Nevertheless, these techniques require an engineer to provide a specific reward function associated to each new particular task to learn.

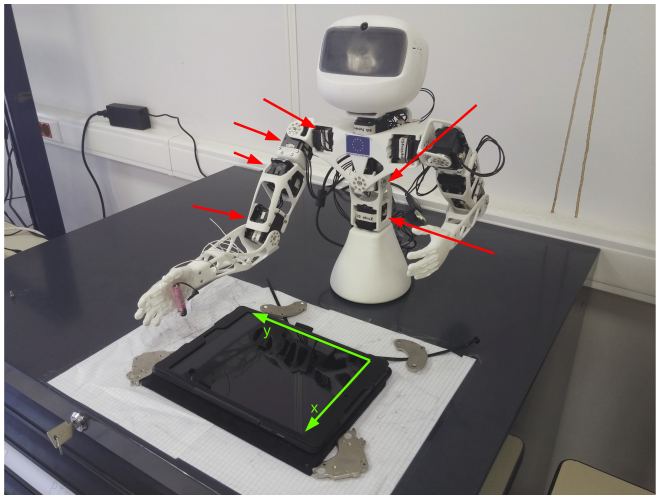

Fig. 1: Experimental setup: the Poppy robot is in front of a tactile tablet it will learn to interact with. The red arrows indicate the motors used. The green arrows represent the axes of the surface of the tablet.

In order to allow high-dimensional robots to learn more autonomously a wider diversity of tasks, methods of active exploration got inspired by *intrinsic motivation* in psychology [5] which triggers spontaneous exploration and curiosity in humans. A first family of such active learning methods is called knowledge-based approaches [6]. To tackle their limitation as sensorimotor learning increases, [7], [8] proposed competence-based approaches where parameterised outcomes were actively sampled through active goal babbling, which generates lower-level goal directed exploration.

However, such active exploration methods are only efficient in low-dimensional outcome spaces[9]. Their efficiency decreases when using real high-dimensional bodies with continuous sensorimotor channels and a large set of tasks, because of curse of dimensionality[10]. Complementary developmental mechanisms need to constrain the learning process, by driving it rapidly toward learnable subspaces. We wish to address the cumulative learning of large sets of tasks, thus examine the mechanism of social guidance for bootstrapping active exploration methods.

B. Interactive learning

To overcome these limitations of autonomous exploration, [11] has shown that human demonstration of skills can help

[1] Nicolas Duminy and Dominique Duhaut are with Université Bretagne Sud, Lorient, France. nicolas.duminy@telecom-bretagne.eu and dominique.duhaut@univ-ubs.f

[2] Sao Mai Nguyen is with Télécom Bretagne, Brest, France. nguyensmai@gmail.com

the learner to identify which the efficient subspaces in the policy and the task spaces, for more efficient exploration.

Besides, an interactive learner who not only listens to the teacher, but actively requests for the information it needs and when it needs help, has been shown to be a fundamental aspect of social learning. Under the interactive learning approach, the robot can combine programming by demonstration and learning by exploration. Several works in interactive learning have considered extra reinforcement signals [12], action requests [13], [14] or disambiguation among actions [15]. In [16], [15] the comparisons of robots that have the option to ask the user for guidance, to the passive robot, show a better accuracy and fewer demonstrations. This approach is the most beneficial both to the learner, for the information arrives as it needs it, and to the teacher, who no longer needs to monitor the learning process.

C. Strategic learning

Both active autonomous exploration and interactive learning can be formalised under the notion of self-organising the learning process. Named "strategic learning" [17], they target the autonomous discovery of the best learning strategy.

One perspective of strategic learning is learning multiple tasks. It aims at selecting which task to spend time on. In sequential problems as in robotics, producing an outcome has been modelled as a local predictive forward model [18], an option [19], or a task [7]. The learning agent decides which outcome to explore/observe next. However most studies using this perspective do not consider several strategies. Another perspective is learning how to learn, by making explicit the choice and dependence of the learning performance on the method, as in [20]. Interactive learning belongs to this perspective. However most studies using this perspective consider a single outcome.

Few works have addressed the learning of both how to learn and what to learn, to choose at the same time which outcome to spend time on, and which learning method to use. These questions were studied in [17], though they only examined a toy example with discrete and finite number of states, outcomes and strategies. To address strategic learning for different outcomes with multiple strategies, we proposed in [21] the Socially Guided Intrinsic Motivation with Active Choice of Teacher and Strategy (SGIM-ACTS) algorithm. It relies on hierarchical active decisions of what and how to learn driven by empirical evaluation of learning progress for each learning strategy. SGIM-ACTS can learn multiple tasks from several teachers by basing its decision on intrinsic motivation, to decide at each learning episode whether to explore autonomously or request for social guidance, and in the case of social guidance, to whom to request for guidance among a set of teachers. However, SGIM-ACTS has only been tested for tasks in 1D or 2D spaces, that are independent. SGIM-ACTS has only been tested in simulation and in deterministic environments.

Developing methods for strategic learning based on active goal-oriented exploration and interactive learning, we propose **Socially Guided Intrinsic Motivation with Active Choice of Task and Strategy for Cumulative Learning (SGIM-ACTSCL) to learn hierarchically organised tasks in larger number and higher-dimension spaces**. We present experimental results with a physical humanoid robot to tackle stochasticity and analyse its behaviour to learn hierarchically complex tasks.

II. Our approach

Our approach belongs to developmental robotics. We aim at adaptive robots for life-long learning that automatically structures its learning process by autonomously choosing its learning strategy for learning motor skills. Inspired by developmental psychology, we consider both autonomous goal-directed learning and imitation learning. We also base our active learning criteria on intrinsic motivation.

In this section, we formalise our learning problem and outline the principles of SGIM-ACTSCL.

A. Problem formalization

The problem is for an agent to learn motor skills. It learns to map policies π_θ, specific motor actions described using parameters $\theta \in \Pi$, and the outcomes $\omega \in \Omega$ which are their effect on the environment. Those outcomes could have different types and dimensionalities and be split in different task spaces $\Omega_i \subset \Omega$. It thus learns a forward model M to predict which outcome ω is produced by a particular policy π_θ in the task space Ω_i. A same policy can reach different outcomes in different task spaces. It builds more importantly an inverse model L to choose which action to execute in order to reach a given outcome (ideally $M(L)$ equals identity). The performance of a policy π_θ at completing an outcome ω is computed on a generic fashion by taking the normalized euclidean distance between ω and the outcome of π_θ for all task spaces. The learner focuses on building an estimate of the inverse model L to improve its performance at reaching any points in the outcome space Ω.

B. Description of SGIM-ACTSCL

SGIM-ACTSCL is a hierarchical algorithmic architecture that merges intrinsically motivated active exploration and interactive learning. The agent learns to achieve different types of outcomes by actively choosing which outcomes to focus on and set as goals, which data collection strategy to adopt and to which teacher to ask for help. It learns local inverse and forward models in complex, redundant and continuous spaces.

SGIM-ACTSCL learns by episodes during which it actively chooses simultaneously an outcome $\omega_g \in \Omega$ to reach and a learning strategy. Its choice of strategy σ is selected between intrinsically motivated exploration and imitation.

In an episode under the imitation strategy, our SGIM-ACTSCL learner actively self-generates a goal ω_g where its competence improvement is maximal. The SGIM-ACTSCL learner explores preferentially goal outcomes easy to reach and where it makes progress the fastest. The selected teacher answers its request with a demonstration $[\zeta_d, \omega_d]$ to produce an outcome ω_d that is closest to ω_g. In the case of the

present study, ω_d and ω_g can belong to different subspaces of the outcome space, and can be of different dimensionality. The robot mimics the teacher to reproduce ζ_d, for a fixed duration, by performing policies π_θ which are small variations of an approximation of ζ_d. Indeed, the demonstration trajectory might be impossible for the learner to re-execute, because of correspondance problems and of the encoding of motor primitives. At the end of the episode, SGIM-ACTSCL computes its competence at reaching the goal ω_d.

In an episode under the intrinsic motivation strategy, it explores autonomously following the SAGG-RIAC algorithm [7]. It actively self-generates a goal ω_g where its competence improvement is maximal, as in the imitation strategy. Then, it explores which policy π_θ can achieve ω_g best. It tries different policies to approach the self-determined outcome ω_g. At the end of the episode, SGIM-ACTSCL computes its competence at reaching the goal ω_g.

An extensive study of the role of these different learning strategies can be found in [24]. Thus the imitation exploration increases the learner's policy repertoire on which to build up self-exploration, while biasing the policy space exploration to interesting subspaces, that allow the robot to overcome high-dimensionality and redundancy issues and interpolate to generalise in continuous outcome spaces. Self-exploration is essential to build up on these demonstrations to overcome correspondence problems and collect more data to acquire better precision according to the embodiment of the robot.

In SGIM-ACTSCL, we adapted SGIM-ACTS [21] for cumulative learning by sharing the observables produced during an episode between all task spaces to enhance the learning process. This enables other task spaces which have been reached too to take the most of the attempt (which is particularly useful when task spaces have dimension overlaps). The teachers were modified to enable them to give a demonstration close to the requested goal for each task space. Details about each module can be read in [21].

III. Experiment

We designed an experiment for a robot to learn to use the tablet, namely to learn an infinite number of tasks, organised as 3 interrelated types of tasks. We carried out our experiment on a real robot with a high number of dimensions for policy and observable spaces. Testing the algorithm on a real platform adds the problem of stochasticity as the control of a real robot and the use of a real sensor (the tablet) add uncertainty. Fig. 2 shows that when repeating several times the same movement, the teacher's demonstration, the points sensed by the tablet are stochastic. We also decided to use the bio-inspired Dynamic Movement Primitives as our robot motion encoders.

A. Experimental setup

The learning agent of this experiment is a Poppy torso robot designed by the flowers team of INRIA Bordeaux [25]. It is equipped with a tactile stylus on its right hand. Before him lays a 10" tactile tablet, which it will learn to interact with, through the learning of 3 interrelated types of tasks described in subsection III-C.3. Each of its actions produces observables of 5 dimensions (section III-C.2).

The robot always starts an episode from the same position, as shown in Fig. 1. The learning algorithm gives a policy to the robot controller to execute. Then the tablet senses the list of points touched and returns to the robot the observables.

B. Dynamic Movement Primitives

We encode our actions as discrete joint space motions using the dynamic movement primitives (DMP) framework [22]. This framework offers many advantages (robustness, temporal and spatial invariance, and guaranteed convergence to the goal) and is thus widely used in robotics.We here use the formulation developed in [23]. Each one dimensional DMP is defined by the system:

$$\tau \dot{v} = K(g - x) - Dv - K(g - x_0)s + Kf \tag{1}$$

$$\tau \dot{x} = v \tag{2}$$

where x and v are the position and velocity of the system; x_0 and g are the starting and end position; τ is a factor used to temporally scale the system; K is like a spring constant; D is the damping term and f is a non-linear term used to shape the trajectory of the motion called the forcing term. It can be learned to fit a given trajectory using learning from demonstrations techniques [**?**] and is defined as:

$$f(s) = \frac{\sum_i \omega_i \psi_i(s) s}{\sum_i \psi_i(s)} \tag{3}$$

where $\psi_i(s) = \exp(-h_i(s - c_i)^2)$, with centres c_i; widths h_i, and weights w_i. The function f does not depend directly on time but uses a phase variable s, which will start at 1 and decrease monotonically to 0 through the motion duration following the canonical system:

$$\tau \dot{s} = -\alpha s \tag{4}$$

The realization of multi-dimensional DMPs is feasible by using one transformation system per degree of freedom (DOF) which share a common canonical system, ensuring henceforth the synchronization of the different DOF throughout the motion. The learning of their forcing term can be done successively.

C. Experiment variables

1) Policy space: We selected 6 joints on the whole robot: the right arm, one joint to rotate the spine and one to bend forward (Fig. 1).

A 6-dimensional DMP is used to encode a policy. The K, D and α parameters of eq. 1 are fixed for the whole experiment. The temporal scaling term τ of the DMP is shared for all the dimensions. The forcing term f_i of each transformation system is coded with 5 basis functions, which locations and widths are fixed for the whole experiment, leaving only their corresponding weights w_i to be parametrized. The end angle g_i of each joint is also a parameter but the starting pose is fixed, the robot always starting from the same pose. Therefore a policy π_θ is parametrized by:

$$\theta = (\omega, a_0, a_1, a_2, a_3, a_4, a_5) \in [0, 1]^{37} \tag{5}$$

where $a_i = (g_i, w_{i,0}, w_{i,1}, w_{i,2}, w_{i,3}, w_{i,4})$ represents the parameters of joint i. The policy space is thus $[0,1]^{37}$.

2) Observable spaces: The effects of the robot's actions are observed by the tablet which acts here as a sensor. The tablet sends the list of all points (x, y) touched by the robot at the end of the mouvement. Using this list, we considered the following observables:

- $M_{start} = (x_{start}, y_{start})$: the first position touched on the tablet by the learner during its attempt.
- $M_{end} = (x_{end}, y_{end})$: the last position touched on the tablet during its attempt.
- l : the length of the drawing on its whole attempt.

3) Task spaces: The tasks the agent will learn to master are normalised combinations of the previously defined observables: $\Omega_1 = \{M_{start}\} = [0,1]^2$, $\Omega_2 = \{M_{start}, M_{end}\} = [0,1]^4$ and $\Omega_3 = \{M_{start}, M_{end}, l\} = [0.1]^5$. We defined the task space as $\Omega = \Omega_1 \cup \Omega_2 \cup \Omega_3$.

These tasks have various degrees of difficulty and some will depend on each other. The idea beyond this choice of interdependent task spaces is to use tasks representing different levels of complexity (different combinations of observables) that the robot could explore progressively. The observables produced by a policy are shared to improve the skill of the robot in all the tasks at once, without restricting them to the task space initially targeted by the policy.

D. The teacher

We use a teacher who has a demonstration dataset, recorded by kinaesthetic on the robot. The dataset consists of 24 demonstrations to touch points regularly distributed on the surface of the tablet. Each demonstration corresponds to outcomes where $M_{start} = M_{end}$. So he is an expert in tasks Ω_1 only. The teacher gives a demonstration when requested for an outcome $\omega_g \in \Omega$ by the robot. For any ω_g in any subspace Ω_1, he chooses the demonstration (π_d, ω_d) which outcome ω_d is the closest to ω_g.

Moreover, due to problems during the experiment, the dataset was built using a Poppy robot different from the one used in the learning phase. The differences in the joints offsets and robot's position introduce a correspondence problem. Fig. 2 shows a shift between demonstrations and repetitions by the robot of the demonstrated policy.

During the experiment, the learner has the choice between 2 strategies: autonomous exploration using SAGG-RIAC algorithm [7] and imitation of the teacher.

E. Evaluation

1) Evaluation method: In order to evaluate our algorithm, we define beforehand a benchmark dataset of outcomes: one set per outcome space or a total of 1691 points (Fig. 3). The task space Ω_3 uses the same lines than Ω_2, except the line length l is added. This evaluation dataset is different from the teacher demonstrations, sharing no common outcomes.

To assess how well the robot can reach each of the outcomes of the evaluation dataset, we compute the closest reached outcomes. We plot the mean distance for predefined

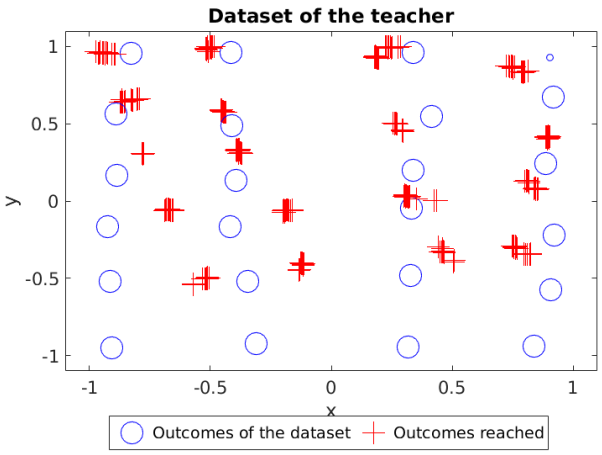

Fig. 2: 34 demonstrations in the teacher dataset (blue circles). For each demonstration, the robot repeats 20 times exactly the same demonstrated movement. the outcomes reached (red crosses) are stochastic. Overall the stylus did not touch the tablet 126 times.

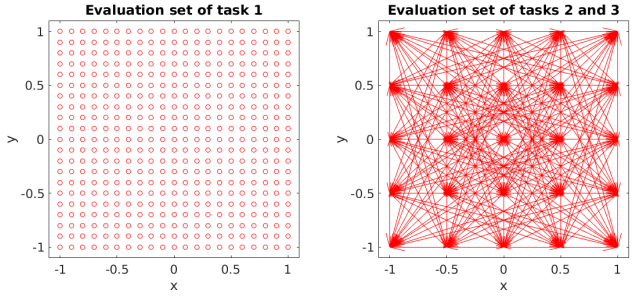

Fig. 3: Evaluation datasets: 441 points for Ω_1, 625 points for Ω_2 and Ω_3

and regularly distributed timestamps. The evaluation is carried out while freezing the learning system. Its results have no impact on the learning process.

2) Compared algorithms : To check the efficiency of our SGIM-ACTSCL algorithm in this experimental setup, we compared with 3 other learning algorithms:

- Random exploration: the robot learns by executing random policies π from the policy space.
- SAGG-RIAC: the learner autonomously explores its environment using goal-babbling without any teacher demonstrations and is driven by intrinsic motivation.
- Imitation: the learner requests a demonstration at a regular frequency, the demonstration given is among the less chosen ones. It is executed and repeated with small variations.
- SGIM-ACTSCL: interactive learning where the learner driven by intrinsic motivation chooses between autonomous exploration or imitation the teacher.

The code for those algorithms is available here.

IV. Results

Fig. 5 plots for the 4 exploration algorithms, the mean distance to outcomes of the evaluation set, through time obtained on those four experiments. It shows that SGIM-ACTSCL outperforms the three others. SAGIM-ACTSCL

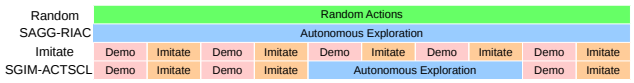

Fig. 4: Strategies of the compared algorithms

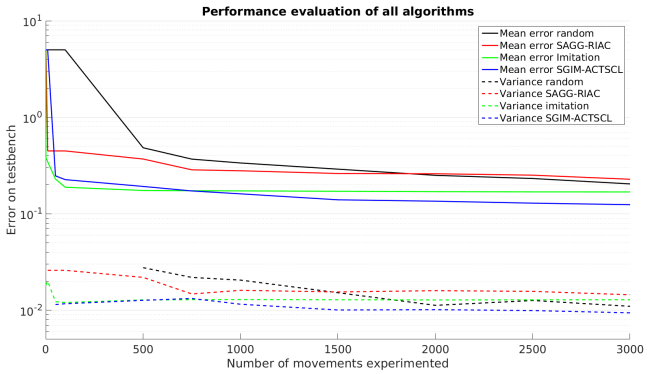

Fig. 5: Mean and variance error for reaching goal averaged on all task subspaces

outperforms Random and SAGG-RIAC from the beginning. From $t > 1000$, it outperforms imitation, owing to goal-oriented self-exploration.

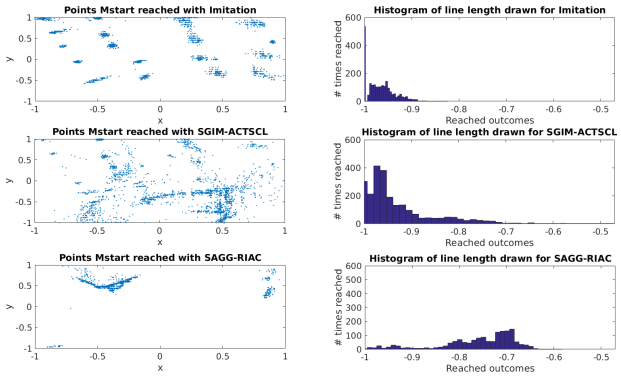

Fig. 6: Points M_{start} reached and histogram of the line length l drawn by Imitation, SGIM-ACTSCL and SAGG-RIAC

Fig. 6 analyses this difference, by plotting the outcomes reached by imitation, SGIM-ACTSCL and SAGG-RIAC. The first column shows that the outcomes in Ω_1 reached by imitation are close to the demonstrations, whereas SGIM-ACTSCL extended its exploration to cover a wider range of outcomes. SAGG-RIAC explored intensively a smaller part of the tablet. Likewise, while demonstrations correspond to outcomes in Ω_3 with only length $l = 0$, the histograms in the second column shows that imitation could increase the length of its drawings a bit, while SGIM-ACTSCL and SAGG-RIAC could draw longer lines.

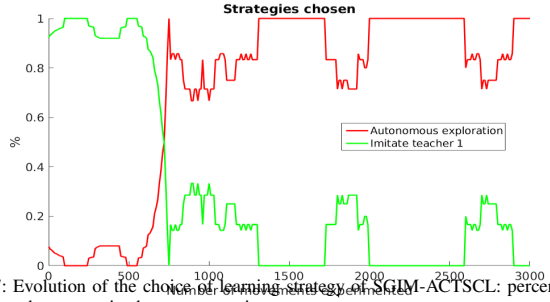

Fig. 7: Evolution of the choice of learning strategy of SGIM-ACTSCL: percentage of times each strategy is chosen across time

While SGIM-ACTSCL outperforms each of its strategy taken alone, we analyse how the SAGG-RIAC and imitation strategies were used by SGIM-ACTSCL through time. Fig 7 shows that in the beginning the robot takes advantage of the imitation strategy which overcomes the difficulty to reach the tablet at first. This difficulty is well shown by the Random algorithm results which only touched the tablet 14 times on the 3000 attempts. Imitation strategy enables it to outperform the self-exploration algorithms, but not the imitation algorithm as the latter was repeating each demonstration equally while the former was not (the demonstrations were chosen according to the robot curiosity). After more than 700 attempts using the imitation strategy, the robot had reproduced most of the teacher demonstrations and changed its strategy to keep progressing. As the teacher was only able to produce points, the learner chose the autonomous exploration strategy which enabled him to reach points farther and farther from the initial points it reached through demonstrations.

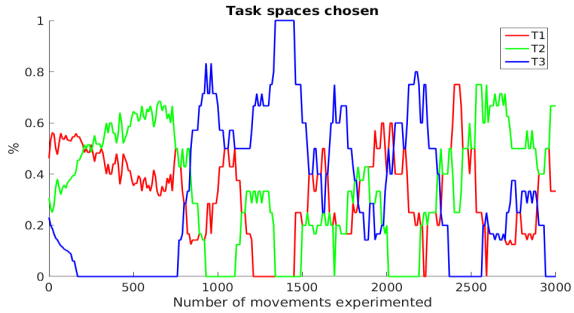

Fig. 8: Evolution of the choice of tasks of SGIM-ACTSCL: percentage of times each task is chosen across time

Detailing the different types of outcomes, Fig. 8 shows 4 phases in time. The learner focuses on type of outcomes Ω_1 in the beginning. The combined choice of the imitation strategy with the task space Ω_1 enabled the robot to progress quickly and starts choosing the task space Ω_2 in the second phase for $250 < t < 800$. After accumulating skills, from $t = 800$, it became able to tackle the most complex task space Ω_3. Finally for $t > 1500$, the SGIM-ACTSCL learner kept using autonomous exploration with the three task spaces, focusing more on the difficult tasks Ω_3 and Ω_2. This finally enabled him to overtake the imitation algorithm by keeping progressing when the latter stagnates.

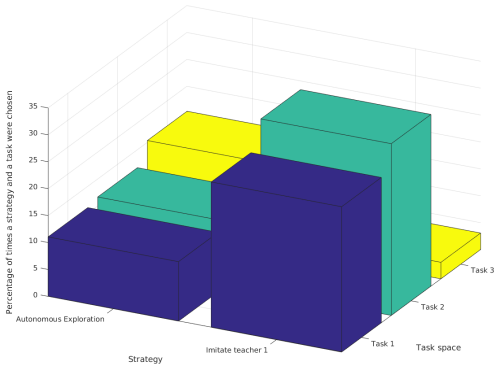

Fig. 9: Synergy between the choice of task space and the choice of learning strategy of SGIM-ACTSCL: percentage of times each strategy and task is chosen over all the learning process

Fig. 9 shows that the task space Ω_3 was mostly combined with autonomous exploration on the overall learning process when the imitation was more associated with Ω_1 and Ω_2. The learner could coherently choose the adequate exploration method for each task.

The learner showed it was capable to make wise strategic

decisions regarding the outcome spaces to tackle and the best strategy to use for it. It **started with the simplest task space** Ω_1 and also tried quickly Ω_2 and chose to **imitate** the teacher for enabling it to make the quickest progress. It was then capable to tackle the most difficult task space Ω_3 and noticed the teacher was less adapted for it. The learner successfully used its first acquired dataset of task spaces Ω_1 and Ω_2 to **autonomously explore the more complex task space** Ω_3.

V. CONCLUSION AND FUTURE WORK

With this experiment, we proved the relevance of SGIM-ACTSCL to learn hierarchically organized tasks on a real robotic platform. The algorithm could **learn a wider range of tasks** and perform with **more precision**. It was capable to use successfully imitation to bootstrap its learning process despite the correspondence problem with the teacher. Though SAGG-RIAC had a lot of issues learning in this experimental setup, its use within SGIM-ACTSCL enabled the learner to widen its range of reached outcomes. It was able to **choose the strategy that allows it to make the most progress**: imitation in the beginning to initialise a repertoire of policies to touch the tablet, then goal-oriented self-exploration to overcome the limits of the demonstrations. It was able to learn all types of tasks by organising its learning process from the easiest to the most difficult. It was also able to **relate the most efficient strategy with corresponding task, thus could recognise the domain of expertise of teachers**.

A more complete statistical analysis with more experiments should confirm these results. Furthermore, while we increased the dimensions and number of tasks with this experiment, we still need to check SGIM-ACTSCL learning capability with a higher number of tasks. The number of teachers could also be increased. Moreover, the tasks the robot had to learn were defined by hand before the experiment. The automatic discovery of task spaces by the learner through the analysis of the observables could be really helpful for using our algorithm on a robot in an unknown environment.

Most of all, this experiment tackles the cumulative learning of skills through the use of primitive actions. Introducing the possibility for the algorithm to combine action primitives in complex actions could also improve its potency as it could focus on the interesting part of the motion. In our application, it could combine a first primitive to reach the tablet and a second one to draw a line. In the long term, combination of primitive actions can lead to an infinite increase in complexity of actions.

Acknowledgment: The work presented in this paper is partially supported by the the European Regional Fund (FEDER) via the VITAAL Contrat Plan Etat Region.

REFERENCES

[1] M. Lopes and P.-Y. Oudeyer, "Active learning and intrinsically motivated exploration in robots: Advances and challenges (guest editorial)," *IEEE Trans. Aut. Mental Development*, vol. 2, no. 2, pp. 65–69, 2010.

[2] J. Peters and S. Schaal, "Reinforcement learning of motor skills with policy gradients," *Neural Networks*, vol. 21, no. 4, pp. 682–697, 2008.

[3] E. Theodorou, J. Buchli, and S. Schaal, "reinforcement learning of motor skills in high dimensions: a path integral approach," in *robotics and automation (icra), 2010 ieee international conference on*, 2010, pp. 2397–2403.

[4] F. Stulp and P.-Y. Oudeyer, "Emergent proximo-distal maturation through adaptive exploration," in *Development and Learning and Epigenetic Robotics (ICDL), 2012 IEEE International Conference on*. IEEE, 2012, pp. 1–6.

[5] E. Deci and R. M. Ryan, *Intrinsic Motivation and self-determination in human behavior*. New York: Plenum Press, 1985.

[6] P.-Y. Oudeyer and F. Kaplan, "What is intrinsic motivation? a typology of computational approaches," *Frontiers in Neurorobotics*, 2007.

[7] A. Baranes and P.-Y. Oudeyer, "Active learning of inverse models with intrinsically motivated goal exploration in robots," *Robotics and Autonomous Systems*, vol. 61, no. 1, pp. 49–73, 2013.

[8] M. Rolf, J. Steil, and M. Gienger, "Gobal babbling permits direct learning of inverse kinematics," *IEEE Transactions on Autonomous Mental Development*, vol. 2, no. 3, pp. 216–229, 09/2010 2010.

[9] P.-Y. Oudeyer, A. Baranes, and F. Kaplan, *Intrinsically Motivated Cumulative Learning in Natural and Artificial Systems*. Springer, 2013, ch. Developmental constraints on intrinsically motivated skill learning: towards addressing high-dimensions and unboundedness in the real world.

[10] C. Bishop, "Pattern recognition and machine learning," in *Information Science and Statistics*. Springer, 2007.

[11] S. M. Nguyen, A. Baranes, and P.-Y. Oudeyer, "Bootstrapping intrinsically motivated learning with human demonstrations," in *IEEE International Conference on Development and Learning*, Frankfurt, Germany, 2011.

[12] A. L. Thomaz and C. Breazeal, "Experiments in socially guided exploration: Lessons learned in building robots that learn with and without human teachers," *Connection Science*, vol. 20 Special Issue on Social Learning in Embodied Agents, no. 2,3, pp. 91–110, 2008.

[13] D. H. Grollman and O. C. Jenkins, "Incremental learning of subtasks from unsegmented demonstration," 2010.

[14] M. Lopes, F. Melo, and L. Montesano, "Active learning for reward estimation in inverse reinforcement learning," in *European Conference on Machine Learning*, 2009.

[15] S. Chernova and M. Veloso, "Interactive policy learning through confidence-based autonomy," *Journal of Artificial Intelligence Research*, vol. 34, 2009.

[16] M. Cakmak, C. Chao, and A. L. Thomaz, "Designing interactions for robot active learners," *Autonomous Mental Development, IEEE Transactions on*, vol. 2, no. 2, pp. 108–118, 2010.

[17] M. Lopes and P.-Y. Oudeyer, "The Strategic Student Approach for Life-Long Exploration and Learning," in *IEEE Conference on Development and Learning / EpiRob*, San Diego, États-Unis, Nov. 2012.

[18] P.-Y. Oudeyer, F. Kaplan, and V. Hafner, "Intrinsic motivation systems for autonomous mental development," *IEEE Transactions on Evolutionary Computation*, vol. 11, no. 2, pp. 265–286, 2007.

[19] A. G. Barto, S. Singh, and N. Chentanez, "Intrinsically motivated learning of hierarchical collections of skills," in *IEEE International Conference on Development and Learning*, 2004.

[20] Y. Baram, R. El-Yaniv, and K. Luz, "Online choice of active learning algorithms," *The Journal of Machine Learning Research,*, vol. 5, pp. 255–291, 2004.

[21] S. M. Nguyen and P.-Y. Oudeyer, "Active choice of teachers, learning strategies and goals for a socially guided intrinsic motivation learner," *Paladyn Journal of Behavioural Robotics*, vol. 3, no. 3, pp. 136–146, 2012.

[22] A. J. Ijspeert, J. Nakanishi, and S. Schaal, "Learning attractor landscapes for learning motor primitives," Tech. Rep., 2002.

[23] P. Pastor, H. Hoffmann, T. Asfour, and S. Schaal, "Learning and generalization of motor skills by learning from demonstration," in *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*. IEEE, 2009, pp. 763–768.

[24] S. M. Nguyen and P.-Y. Oudeyer, "Properties for efficient demonstrations to a socially guided intrinsically motivated learner," in *21st IEEE International Symposium on Robot and Human Interactive Communication*, 2012.

[25] M. Lapeyre, P. Rouanet, and P.-Y. Oudeyer, "Poppy Humanoid Platform: Experimental Evaluation of the Role of a Bio-inspired Thigh Shape," in *Humanoids*, Atlanta, United States, Oct. 2013.