Developmental Learning for Social Robots in Real-World Interactions

Alexandre Galdeano, Alix Gonnot, Clément Cottet, Salima Hassas, Mathieu Lefort, Amélie Cordier

Developmental Learning for Social Robots in Real-World Interactions

Alexandre Galdeano[1], Alix Gonnot[2], Clément Cottet[3], Salima Hassas[4], Mathieu Lefort[5], and Amélie Cordier[6]

***Abstract*— This paper reports preliminary research work on applying developmental learning to social robotics for making human-robot interactions more instinctive and more natural. Developmental learning is an unsupervised learning strategy relying on the fact that the learning agent is intrinsically motivated, and is able to incrementally build its own representation of the world through its experiences of interaction with it. Our claim is that using developmental learning in social robots could dramatically change the way we envision human-robot interaction, notably by giving the robot an active role in the interaction building process, and even more importantly, in the way it autonomously learns suitable behaviors over time. Developmental learning appears to be an appropriate approach to develop a form of "interactional intelligence" for social robots. In this work, our goal was to set up a common framework for implementing, experimenting and evaluating developmental learning algorithms with various social robots.**

***Index Terms*—Developmental Learning, Social Robotics, Interactions, Intrinsic Motivation, Hierarchical Learning, Enaction.**

I. Introduction

Developmental learning is an artificial intelligence approach drawing inspiration from psychology, and more specifically from the way children learn and build incrementally their own representation of the world by experimenting their environment through interactions [1], [2]. In robotics, developmental learning has been used to address problems such as reaching and grasping [3], exploration and curiosity [4], navigation [5], speech acquisition [6], [7], and adaptation to arm and leg damage [8].

In the field of social robotics, developmental learning appears to be an interesting approach to consider, notably because of its potential to bring answers to problems such as life-long learning, unsupervised learning without prior knowledge, and, last but not least, its incremental capacity to learn from few data.

In this paper, we present preliminary investigations on applying developmental learning for improving human-machine interaction in the field of social robotics. The work reported here was performed within the project BEHAVIORS.AI (an Engine enHancing verbAL and non-Verbal InteractiOns of RobotS, based on Artificial Intelligence). BEHAVIORS.AI is a joint laboratory[1] involving academic researchers and social robotics engineers which aim is to design and develop solutions to make human-robot interaction more instinctive and empathic. For that purpose, we explore the opportunities offered by developmental learning, we propose implementations on social robots, and we use the social robots field to implement and improve this framework.

Our goal is to implement a form of "interactional intelligence" in social robots. For that purpose, we focus on several aspects of social interactions: emotional intelligence, timing of the interaction, adaptability to the changing context, preference learning, etc. In the context of this project, we focus on how developmental learning could enable continuous learning of interaction skills and, ultimately, lead to the emergence of appropriate (and unique) behaviors in social robots. In this work, our goal was to set up a common framework for implementing, experimenting and evaluating developmental learning algorithms with various social robots. For bootstrapping our research, we decided to start with a very simple developmental learning algorithm that we chose because it is easy to implement and to analyze, and it offers interesting challenges (hierarchical learning, implementation of motivational systems, etc.). In the future, we will focus on more advanced algorithms.

This paper is organized as follows. In the next section, we briefly introduce developmental learning and we describe some of the theoretical hypotheses we rely on. In section III, we describe our work in progress and report on two distinct contributions. First, we present a first implementation. This implementation validates the potential of developmental learning algorithms to enhance the intelligence of interactions in social robots and highlights the next challenges to face. Next, we present a visualization tool that we have designed in order to help researchers better design developmental algorithms for social robots. The paper ends with a discussion on the challenges and perspectives we foresee for the next steps of the project.

II. Developmental Learning

Developmental learning is a constructivist approach, inspired by theories on the cognitive development of human being (learn like a baby) [1], [2], [9], that aims to address the "Symbol Grounding problem" [10] by making an agent interacting with its environment "constructs sensorimotor transformation knowledge rather than an internal mirror of

*This work was supported by the French *Agence Nationale de la Recherche* (ANR) contract ANR-16-LCV2-0003-01.

[1, 2, 3]Alexandre Galdeano, Alix Gonnot, and Clément Cottet are with Hoomano, 4 rue du Professeur Charles Appleton, 69007, LYON, France and with Univ Lyon, Université Lyon 1, CNRS, LIRIS, F-69621, VILLEURBANNE, France {agaldean, agonnot}@liris.cnrs.fr, clement.cottet@etu.univ-lyon1.fr

[3, 4,5]Salima Hassas and Mathieu Lefort are with Univ Lyon, Université Lyon 1, CNRS, LIRIS, F-69621, VILLEURBANNE, France {shassas, mlefort}@liris.cnrs.fr

[6]Amélie Cordier is with Hoomano, 4 rue du Professeur Charles Appleton, 69007, LYON, France amelie.cordier@hoomano.com

[1]BEHAVIORS.AI involves people from the company HOOMANO and LIRIS's SMA research group. See behaviors.ai for more details.

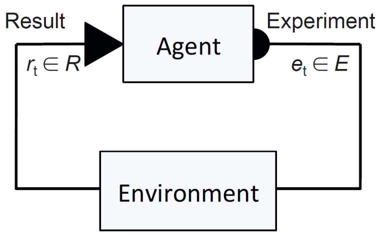

Fig. 1. Traditionally, the agent receives an input (observation) and computes the output (action). However, in our approach, the agent initiates the cycle by acting (experiment) then perceives the consequences (result). This is a way to implement an intrinsically motivated learning algorithm. Reprinted from [19, p. 246].

some external reality" [11, p. 164]. As such, the learning process occurs through the experience developed by the agent along its interaction with its environment [12], and, according to some authors, learning these sensorimotor contingencies may even lead to some kind of consciousness [13]. Moreover, the agent's behavior is driven by an intrinsic motivation and its value function depends on the agent's behavior rather than of the environment states. We claim that such an approach has the necessary ingredients to develop autonomous robots that are able to adapt to highly dynamic environments that include humans. Using active perception may help the agent to deal with complex environments [13] and autonomously adapt to new situations thanks to its incremental learning and its intrinsic motivation. Learning from its experience of interaction with humans, including imitating them or drawing inspiration from their behaviors, allows for more natural interactions.

This learning paradigm may be implemented in several ways. For instance, in [14], the authors use an implementation based on neural networks to detect contingencies in sensorimotor networks. In [15] developmental learning is based on a set of "tripartite structure[s] comprising a context, action, and result" called schema, this structure is then completed in [16] by adding a target value, and in [17] the original schemas have been extended to "improve the original learning criteria to handle POMDP domains".

In this preliminary work, we chose to rely on a hierarchical implementation of schemas, drawing inspiration from the intrinsically motivated schema mechanism proposed in [18]. We made this choice because it is a simple and intuitive way to envision developmental learning. Our goal is to demonstrate that rudimentary implementations of developmental learning algorithms could be performed on social robots and could lead, in the long run, to new ways of implementing unsupervised learning abilities on these robots. More complex approaches and implementations are kept for future work.

This model is grounded on several key concepts:

- An **interaction** is an atomic element defined as a couple composed of the experiment performed by the agent and the result obtained for this experiment (see figure 1). Focusing on the interaction implements a form of active perception. Moreover, the agent does not require any direct access to the state of the environment.
- An **intended interaction** is an interaction that the agent tries to perform (i.e., it is an action and an expected result for this action).
- An **enacted interaction** is an interaction that the agent has actually performed (i.e., an action and its result, that may be different from the expected result).
- The **valence** of an interaction is the internal value associated by the agent to it. It can be seen as the cost of doing the interaction from the agent's perspective.
- The **proclivity** is a weighted value computed by the agent, depending on its internal emotional state. At each step, the agent computes the proclivity of all the interaction that it can enact and chooses the best one given its current decision-making strategy (depending on its emotional state).

In our approach, we observe the following properties.

- The agent reacts to its **active perception** of the environment, after experimenting an interaction with it, by attempting a new experiment based on its internal evaluation of its perception (difference between its prediction and the actual result of the experiment).
- The agent creates **hierarchical aggregated schemas (a kind of abstraction)**. It memorizes the enacted interactions and aggregates them hierarchically, thus creating composite interactions. A composite interaction is an interaction composed of several primitive or composite interactions that can be performed in a row (see figure 2 for an example). Composite interactions can be seen as higher level representations of the abilities of the agent.
- The agent implements a strategy to **balance exploration and exploitation**. The agent chooses what to do next based on the set of enacted interactions using different strategies depending on the agent's internal state, the interactions' proclivity, and the previous interactions. Typically, the agent chooses the interaction with the highest proclivity but the agent's internal state may make it choose another interaction. This mechanism improves the diversity of enacted interactions and allows a better exploration of the interaction space.

This approach has been successfully applied on several toy problems. Albeit it seems a promising way to address the problem of unsupervised and intrinsically motivated learning without any prior knowledge on the environment,

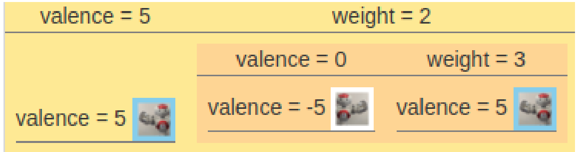

Fig. 2. Example of a composite interaction with its weight and valence, and its sub-interactions. Here, a complex move performed by the robot is in fact a sequence of primitive moves. One can observe that the background color of the symbols changes: it reflects the result perceived by the agent, i.e., blue for "my head has been touched" and white for "my head has not been touched".

it also raises numerous concerns. Just to list a few, we could mention the memory consumption (the current naive implementation is very costly as it records all interactions from the start), the sensitivity to noise, the required regularity of the environment, and the convergence of the learning process.

Our goal in the preliminary work reported here was to implement this approach on social robots, first to validate that they could at least provide the same results as in simulation, and second, to better identify the problems related to the reality gap. These contributions are described in the next section.

III. Contributions

We started with toy implementations and we plan to gradually increase the complexity of our experiments.

A. Simple Experiment on a Social Robot

In this experiment we wanted to demonstrate that developmental learning could be used to make a social robot able to favor specific actions according to the reactions of the human. In the scope of social interaction, the robot's user act as the environment.

Using this approach, we were able to make a Nao robot learn sequences of two actions according to the preferences of the human interacting with it. In this experiment, Nao can do four primitive actions: 1) turn its head to the left; 2) turn its head to the right; 3) rise its left arm; 4) rise its right arm. After each movement, the robot waits a few seconds for the environment's feedback—here the user touching, or not, the robot's head. Each pair (movement, feedback) is an interaction. All the interactions which feedback is “The head sensor has been triggered” have a positive valence and the other interactions have a null valence. Starting from the second interaction enacted, the robot memorizes every distinct pair of interactions it enacts—i.e., if the robot enacts ABC, it memorizes AB and BC, and then associates a weight to it. The weight goes up every time the pair of interactions is enacted and starts at 1 when the pair is enacted for the first time. Each pair of interaction, called *composite interactions*, is considered as an interaction so that it can be combined with others. The valence of the composite interaction is equal to the valence of its last interaction.

The robot alternates regularly between exploration and exploitation. **Exploitation.** The robot chooses which simple interaction should enable it to enact the best *composite interaction*. The best *composite interaction* is the one with the highest proclivity (in this implementation, $proclivity = valence \times weight$). **Exploration.** The robot chooses a random action in order to learn a new *composite interaction*.

With this experiment, we showed that we were able to perform a robot implementation of a previous algorithm performing well in simulation. Besides, when using this algorithm, the robot is able to learn a simple behavior adapted to the user's preferences.

We learned a lot about the problems we will face when increasing the complexity of our algorithms. We identified two obvious obstacles. First, the robot's movement and the user's reactions are a few seconds long which—compared to the near-instantaneous reaction times in simulation—is very long and that makes the execution of the algorithm with the robot a lot slower than in a simulation. Our research question here is to manage timing, i.e., to find a strategy to decide when and how to gather feedback from the environment and to match it with the active perception strategy of the agent. Second, the robot and the environment are less reliable than the computer and some errors may occur during the execution of the algorithm, causing the robot to learn wrong behaviors and to have to deal with noisy information. To go further and to facilitate the research work, we implemented a tool for the researchers for accelerating the design and the tunning of developmental learning algorithms.

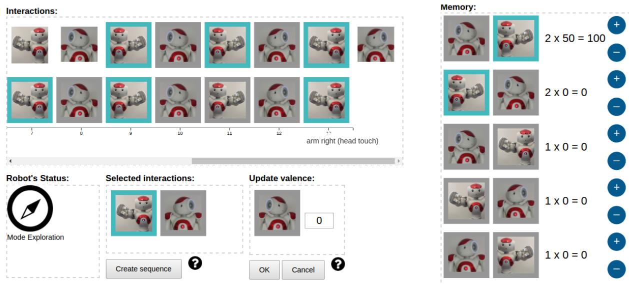

Fig. 3. Visualization tool is used for the developmental algorithm presented in section III-A. The robot's memory and history of actions, the current type of behavior it has, and what it is about to do are displayed in real-time. The valence of each sequences may be altered for experimental purposes.

B. Web Application for Visualization Experiment Control

The interface enables us to visualize the execution trace of the algorithms and to modify it on the fly (see figure 3). The interface displays the mental state of the robot in real-time by showing every simple interaction the robot ever enacted, the content of its memory (i.e., all the enacted composite interactions with weights and valences) and the type of behavior it is currently in (i.e., exploration or exploitation).

Elements currently active in the decision and memory processes are highlighted on the screen in real-time. This interface also allows the control of the agent's memory by the designer: increasing or decreasing the weight of a composite interaction, modifying the valence of a composite interaction, or add a new composite interaction like it was enacted. Modifying these elements through the interface enables us to observe how the learning process is impacted without re-enacting the experiment from scratch.

Next, we created a second version of this interface with additional features (see figure 4). This new version is capable of displaying complex composite interactions and have some control on the algorithms, particularly with a virtual agent: (i) start, pause, and stop the agent's execution, (ii) define the agent's execution rate, i.e., how many milliseconds are between each steps, (iii) define the policy that will have the best valence and (iv) define some of the agent's preferences.

These two interfaces help us to have a better understanding

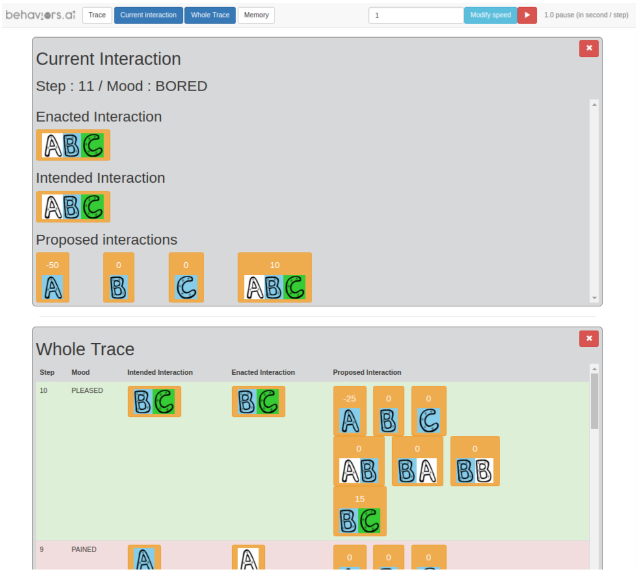

Fig. 4. This version is an improvement to the first one. The robot's memory and history of actions, its mood are displayed. The algorithm can be controlled and configured—to some extent—through this interface.

of how our algorithms behave under different and changing configurations. The control features can be used to speed up the robot's learning and therefore constitute useful tools for researchers and interaction designers in an experimentation context. They can quickly test and validate hypotheses through this tool, in order to produce configuration ready to be tested in real-world experiments.

C. *Multi-robots implementations and experimentations*

So far, we have built an architecture enabling us to experiment different developmental learning algorithms on various robots, including virtual robots (agents in a simulation environment). We have conducted experiments with Nao and Pepper, and we have an ongoing implementation for Cozmo. A virtual agent—combined with a virtual environment—has also been implemented in order to enable us to debug and test the behavior of an algorithm under various—and possibly varying—conditions quickly. We have proposed three different implementations of developmental learning algorithms, with increasing complexity, in different contexts, but their description is out of the scope of this paper. Demonstrating that the same algorithms behave well with different robots and for different application purposes is important as it shows that active perception and intrinsic motivation are key factors to enable learning without prior knowledge on the environment.

IV. Discussion and Future Work

We defended the idea that developmental artificial intelligence is a relevant approach for developing social robots that can interact in a more natural way with humans. Indeed, the developmental paradigm is intrinsically defined to provide autonomy, lifelong learning and adaptability to an agent, which are required properties for a social robot. Moreover, taking inspiration from human may provide more natural behaviors to robots. Preliminary experiments show that a developmental architecture, based on interaction, can lead a robot to learn a behavior desired by the human, providing indirect feedback through its actions. However, developmental AI is a recent research field and some major questions still have to be answered. To go beyond, we developed a web application that allows to monitor (and eventually to modify) the internal representations learned by the agent.

Our preliminary work is a starting point, raising several issues that we have to address to reach the objectives of our project. For instance, to go further, we will need to address scalability issues and robustness to noisy environments. Some other issues are related to improving the implementation and optimizing memory usage, choosing more complex—but more efficient—approaches, dealing with forgetting issues, etc. But more broadly, in the context of the BEHAVIORS.AI project, we want to study:

a) How to choose the next best action to perform (depending on several parameters such as the intrinsic motivation, the trade-off between exploration and exploitation, the context), and how to take into account the timing in the process, which is of particular importance in social interactions?
b) Using high level actions and perceptions may produce more complex behaviors without significantly increasing the search space's size, but how does it affect the algorithm's adaptability and relevance? In addition, the environment is, by definition, noisy. How to deal with noise and the potential lack of consistency of the environment?
c) How to include a form of emotional intelligence, through empathy, in the process. Could we envision a way to share representations of the interactional context between humans and robots to globally improve the interactions?
d) Could we include multimodality in the process in order to combine verbal and non-verbal interactions in the context understanding process?
e) Could we estimate how developmental learning mechanisms implemented in social robot's applications affect the user's experience?

Last, we are deeply interested in finding ways to evaluate the performances of developmental learning. Evaluating performances of such algorithms, when we cannot identify an a priori goal, is never easy. How can we assess that a solution performs well, or better than another, when the goal is not defined? How can we measure the impact on the user experience? Proposing tools for evaluating both the technical performances of our contributions and the improvements they bring on the user experience is a major objective for our future work.

References

[1] J. Piaget, *La construction du réel chez l'enfant*. Delachaux and Niestlé, 1937.

[2] A. Cangelosi, M. Schlesinger, and L. B. Smith, *Developmental robotics: From babies to robots*. 2015.

[3] P. Savastano and S. Nolfi, "A robotic model of reaching and grasping development," *IEEE transactions on autonomous mental development*, vol. 5, no. 4, pp. 326–336, 2013.

[4] P.-Y. Oudeyer and F. Kaplan, "What is intrinsic motivation? a typology of computational approaches," *Front. Neurorobot.*, vol. 2009, 2009. DOI: `10.3389/neuro.12.006.2007`.

[5] R. C. Arkin, "Motor schema—based mobile robot navigation," *The International journal of robotics research*, vol. 8, no. 4, pp. 92–112, 1989.

[6] C. Moulin-Frier and P.-Y. Oudeyer, "The role of intrinsic motivations in learning sensorimotor vocal mappings: A developmental robotics study," in *INTERSPEECH 2013, 14th Annual Conference of the International Speech Communication Association, Lyon, France, August 25-29, 2013*, 2013, pp. 1268–1272.

[7] 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," *IEEE Transactions on Autonomous Mental Development*, vol. 2, no. 3, pp. 167–195, 2010.

[8] A. Cully, J. Clune, D. Tarapore, and J.-B. Mouret, "Robots that can adapt like animals," *Nature*, vol. 521, no. 7553, pp. 503–507, 2015.

[9] F. Guerin, "Learning like a baby: A survey of artificial intelligence approaches," *The Knowledge Engineering Review*, vol. 26, no. 2, pp. 209–236, 2011.

[10] S. Harnad, "The symbol grounding problem," *Physica D: Nonlinear Phenomena*, vol. 42, no. 1, pp. 335–346, 1990.

[11] T. Ziemke, "The construction of 'reality' in the robot: Constructivist perspectives on situated artificial intelligence and adaptive robotics," *Foundations of science*, vol. 6, no. 1, pp. 163–233, 2001.

[12] S. Mazac, F. Armetta, and S. Hassas, "On bootstrapping sensori-motor patterns for a constructivist learning system in continuous environments," in *Alife 14: Fourteenth International Conference on the Synthesis and Simulation of Living Systems*, 2014.

[13] J. K. O'Regan and A. Noë, "A sensorimotor account of vision and visual consciousness," *Behavioral and brain sciences*, vol. 24, no. 5, pp. 939–973, 2001. (visited on 03/31/2017).

[14] A. Pitti, H. Mori, S. Kouzuma, and Y. Kuniyoshi, "Contingency perception and agency measure in visuo-motor spiking neural networks," *IEEE Transactions on Autonomous Mental Development*, vol. 1, no. 1, pp. 86–97, 2009.

[15] G. L. Drescher, *Made-up minds: A constructivist approach to artificial intelligence*. MIT press, 1991.

[16] F. Guerin and D. McKenzie, "A piagetian model of early sensorimotor development," *Proceedings from Epigenetic Robotics*, vol. 2008, 2008.

[17] M. P. Holmes *et al.*, "Schema learning: Experience-based construction of predictive action models," in *Advances in Neural Information Processing Systems*, 2005, pp. 585–592.

[18] O. L. Georgeon, A. Mille, A. Cordier, A. Kong Win Chang, J. Garnier, S. L. Gay, A. Corny, C. Barbier-Gondras, and M. Jonathan. (Jul. 12, 2017). Implementation of DEvelopmentAl learning (IDEAL) course, Implementation of DEvelopmentAl Learning (IDEAL) MOOC, [Online]. Available: `http://liris.cnrs.fr/ideal/mooc/`.

[19] O. L. Georgeon and A. Cordier, "Inverting the interaction cycle to model embodied agents," *Procedia Computer Science*, vol. 41, pp. 243–248, 2014.