Explauto: an open-source Python library to study autonomous exploration in developmental robotics

Clément Moulin-Frier, Pierre Rouanet, Pierre-Yves Oudeyer

▶ **To cite this version:**

HAL Id: hal-01061708

https://inria.hal.science/hal-01061708

Submitted on 8 Sep 2014

Explauto: an open-source Python library to study autonomous exploration in developmental robotics

Clément Moulin-Frier, Pierre Rouanet, Pierre-Yves Oudeyer
Flowers team, Inria/ENSTA-Paristech
Bordeaux, France
Email: {firstname}.{lastname}@inria.fr

I. INTRODUCTION

Last year at the ICDL-Epirob conference [1], we presented a probabilistic framework unifying two important families of exploration mechanisms recently shown to be efficient to learn complex non-linear redundant sensorimotor mappings, i.e. the mappings between the motor actions made by a robot and the sensory effects they produce. This can be for example a robot learning how arm movements make physical objects move, or how movements of a virtual vocal tract modulates vocalization sounds. These two exploration mechanisms are: 1) goal babbling [2], [3] and 2) active learning driven by the maximization of empirically measured learning progress [4], [3]. Both principles can be combined in various ways to achieve efficient exploration strategies –as opposed to random motor babbling, where collected sensorimotor data are generally too sparse to efficiently feed machine learning algorithms. This is especially true when the motor and sensory spaces are highly dimensional, the mapping between them is non-linear and redundant, and there is limited time allowed for learning –the general case of robotic systems.

We now provide an open-source Python library, called Explauto , providing a unified API to design and compare various exploration strategies driving various sensorimotor learning algorithms in various simulated or robotics systems. Explauto aims at being collaborative and pedagogic, providing a platform to developmental roboticists where they can publish and compare their algorithmic contributions related to autonomous exploration and learning, as well as a platform for teaching and scientific diffusion. The library is available at this address: https://github.com/flowersteam/explauto.
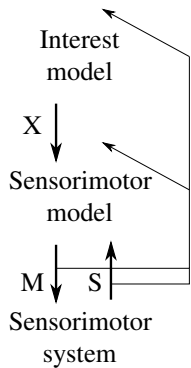
II. THE EXPLAUTO LIBRARY

A. Architecture

The Explauto architecture is illustrated in Figure 1a. We describe it below from bottom to top. We call *sensorimotor system* the physical properties of the robot and the environment in which it evolves. Programmatically, if we call M the space of motor commands allowed for a robot –e.g. target positions of the joints of a robotic arm, and S the space of sensory effects it perceives –e.g. the 3D position of its end effector, the sensory motor system must implement a way to compute the sensory effect $s \in S$ corresponding to a given motor action $m \in M$. In the current version of the library, we provide two simulated sensorimotor systems (a multi-joint arm acting on a plan and an under-actuated torque-controlled inverse pendulum) as well as a system allowing the control of real robots based on Dynamixel actuators by providing bindings to the Pypot library[1]. We also provide bindings to an existing articulatory synthesizer[2] which can be installed as an optional package.

The cognitive architecture of an agent is then composed by a *sensorimotor model*, which iteratively learns internal forward and inverse models from experience, and an *interest model* which actively chooses where to explore in the sensorimotor space. A *sensorimotor model* implements both the iterative learning process from sensorimotor experience, i.e. from the iterative collection of $(m, s) \in M \times S$ pairs by interaction with the environment, and the use of the resulting internal model to perform forward and inverse predictions (or any kind of general prediction between sensorimotor subspaces). Explauto provides several sensorimotor models: simple nearest-neighbor look-up, non-parametric models combining classical regressions and optimization algorithms, online local mixtures of Gaussians, and discrete Lindstone distributions. An *interest model* implements the active exploration process, where sensorimotor experiments are chosen to improve the forward or inverse prediction of the sensorimotor model. It explores in a given *interest space* $X \subset M \times S$, resulting e.g. in motor babbling strategies when $X = M$ or in goal babbling strategies when $X = S$. *Interest models* also have to implement a sampling procedure in X. Explauto provides several sampling procedures: random sampling as well as sampling maximizing the learning progress in forward or inverse predictions.

Explauto comes with a high level programming interface to compare the built-in, optional or user-defined implementations of sensorimotor systems, sensorimotor models and interest models (Figure 1b). It automatically takes advantage of the multi-core processors commonly found in modern personal computers, allowing the parallel execution of multiple experiments (parallelism

---

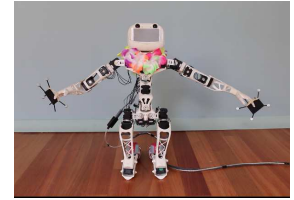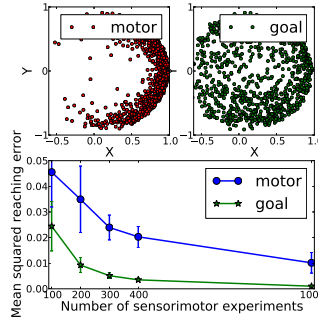[1] https://github.com/poppy-project/pypot
[2] http://www.bu.edu/speechlab/software/diva-source-code/

(a) The Explauto architecture. Light arrows represent training feedbacks

(b) High-level API to compare various exploration strategies

(c) Plotting facilities. Comparison of hand positions reached by a simulated arm (top) using motor vs. goal babbling and associated learning curves (bottom)

(d) Autonomous exploration on a humanoid Poppy robot

Fig. 1. Explauto is based on a unified architecture, provides a high level API allowing to compare various exploration strategies in a few lines of code and can be executed on physical robots such as the open-source humanoid Poppy [6] (see a demonstration video at https://vimeo.com/95405850)

on multiple machines has not been tested yet). It also contains plotting facilities such as the visualisation of the motor commands and reached effects over time, as well as the learning curves (Figure 1c).

B. Extending the library

Explauto is distributed under an open-source license (GNU GPL V3) and can be easily extended by subclassing three abstract classes providing a common interface and the base processing procedures for sensorimotor systems (*Environment* class), sensorimotor models (*SensorimotorModel* class) and interest models (*InterestModel* class). We encourage anyone to contribute by sending us pull-requests from our Github public repository. For example, we are currently working on an interface with a Baxter robot, on the integration of a state-of-the-art sensorimotor learning algorithm (Infinite Mixture of Linear Experts, [5]), as well as new implementations of the learning progress measure [1].

III. CONCLUSION

The Explauto library provides a common platform to the Developmental Robotics community allowing the integration and comparison of various exploration strategies driving various sensorimotor learning algorithms in various simulated or robotics systems. Many other exploration strategies could be easily integrated into the library, as for example Direction Sampling [7], compression progress [4], empowerment [8] and thus be compared in a proper way on various sensorimotor systems and using various sensorimotor internal models.

Explauto also aims to be a tool for scientific diffusion, in particular teaching. The library comes with a series of short Python tutorials explaining by practice the concepts of sensorimotor learning by experience, motor and goal babblings and intrinsically-motivated exploration.

ACKNOWLEDGMENT

REFERENCES

[1] C. Moulin-Frier and P.-Y. Oudeyer, "Exploration strategies in developmental robotics: A unified probabilistic framework," in *International Conference on Development and Learning, ICDL/Epirob, Osaka, Japan*, 2013, pp. 1–6.

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

[3] A. Baranes and P.-Y. Oudeyer, "Active learning of inverse models with intrinsically motivated goal exploration in robots," *Robotics and Autonomous Systems*, 2012.

[4] J. Schmidhuber, "A possibility for implementing curiosity and boredom in model-building neural controllers," in *Proc. SAB'91*, J. A. Meyer and S. W. Wilson, Eds., 1991, pp. 222–227.

[5] B. Damas and J. Santos-Victor, "Online learning of single-and multivalued functions with an infinite mixture of linear experts," *Neural computation*, vol. 25, no. 11, pp. 3044–3091, 2013.

[6] M. Lapeyre, P. Rouanet, J. Grizou, S. Nguyen, A. L. Falher, F. Depraetre, and P.-Y. Oudeyer, "Poppy: Open source 3d printed robot for experiments in developmental robotics." in *International Conference on Development and Learning, ICDL/Epirob, Genova, Italy*, 2014.

[7] M. Rolf, "Goal babbling with unknown ranges: A direction-sampling approach," in *IEEE Int. Conf. Development and Learning and on Epigenetic Robotics, Osaka, Japan*, 2013.

[8] A. S. Klyubin, D. Polani, and C. L. Nehaniv, "All else being equal be empowered," in *Advances in Artificial Life, European Conference on Artificial Life (ECAL 2005)*, ser. LNAI, Springer, Ed., vol. 3630, 2005, pp. 393–402.