An Intrinsically Motivated Planning Architecture for Curiosity-driven Robots*

Angelo Oddi, Riccardo Rasconi, Vieri Giuliano Santucci, Gabriele Sartor, Emilio Cartoni, Francesco Mannella, and Gianluca Baldassarre

ISTC-CNR, Via San Martino della Battaglia, 44 - 00185 Rome, Italy
{angelo.oddi,riccardo.rasconi,vieri.santucci,gabriele.sartor, emilio.cartoni,francesco.mannella,gianluca.baldassarre}@istc.cnr.it

1 Background and Objectives

This paper presents a summary of the IMPACT (*Intrinsically Motivated Planning Architecture for Curiosity-driven roboTs*) project funded by the European Space Agency (ESA). The project aimed at investigating the possibility of employing Artificial Intelligence (AI) techniques to increase both the cognitive and operational *autonomy* of artificial agents in general, and of robotic platforms targeted at the space domain in particular. The idea is based on the creation of a virtuous loop in which the agent increases its learned capabilities through the direct interaction with the real environment, and then exploits the autonomously acquired knowledge to execute activities of increasing complexity: this process is cumulative and virtually *open-ended* [3] as the information and abilities acquired up to a certain time are employed to further increase the agent's knowledge of the application domain, as well as the skills to adequately operate in it. This self-induced tendency towards autonomously learning new skills, based on *intrinsic motivations* (IM) [16, 5], will enable the artificial agent to face situations and solve problems not foreseeable when the agent is designed and implemented, especially because of the limited knowledge on the environment the agent will operate in.

The IMPACT software framework [11] aimed at extending the well-known *three-layered* robot control architecture [1, 4, 6] commonly accepted in general robotics to support the *Sense-Plan-Act* (SPA) autonomous deliberation and execution paradigm. Indeed, the IMPACT system implements a *Discover-Plan-Act* (DPA) cycle, which directly extends the SPA cycle with a more general open-ended learning step (*Discover*) acquiring new knowledge from the external environment. In particular, within the three-layer architecture we integrated the following new functionalities: (1) autonomous learning of new *skills* based on self-generated goals driven by intrinsic motivations (intrinsic goals) [15]; (2) automatic translation of the newly acquired skills [12], from a low-level sub-symbolic representation to a high-level symbolic representation (e.g., expressed in Planning Domain Definition Language - PDDL [10]); (3) autonomous enrichment of the planning domain by adding knowledge on new states and operators expressed in

* This research has been supported by the European Space Agency (ESA) under contract No. 4000124068/18/NL/CRS, project IMPACT - Intrinsically Motivated Planning Architecture for Curiosity-driven roboTs. The view expressed in this paper can in no way be taken to reflect the official opinion of the European Space Agency.

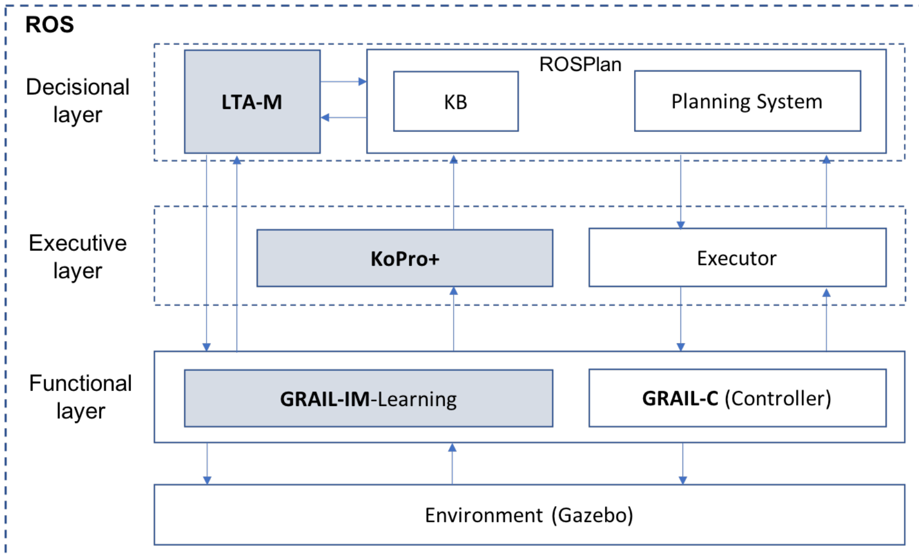

Fig. 1. IMPACT high-level software architecture

high-level symbols. Next section describes the IMPACT software architecture, whereas Section 3 introduces two test scenarios using the Gazebo Robot Simulator [7] [1]: a planetary rover and a robotic arm.

2 IMPACT Architecture

Figure 1 represents the IMPACT high-level software architecture. As introduced above, it is based on the three-layer robot control architecture [1, 4, 6] commonly accepted in general robotics to support the *Sense-Plan-Act* (SPA) autonomous deliberation and execution paradigm. In this architecture, the *decisional layer* implements the high-level planning, plan execution, and re-planning capabilities; the *executive layer* controls and coordinates the execution of the functions distributed in the software, according to the requirements of the high-level tasks that have to be executed; lastly, the *functional layer* implements all the basic, built-in robot sensing, perception, and actuation capabilities. In Figure 1 the new blocks are highlighted (in light grey) within the three layers to represent the proposed extensions for including learning capabilities. In the following subsections we describe in details the single blocks.

The Decisional Layer. Within the *classical* framework of the three-layer architecture the decisional layer traditionally contains a task *planner* to generate a sequence of operations to reach a certain *goal* provided by the users. The IMPACT project has the main challenge of integrating this planning capabilities with autonomous intrinsically motivated learning (IM-learning) features for increasing the number of known skills and

[1] http://gazebosim.org/

extending the related high-level model in the robot's knowledge base. We think that the given extended vision of the three-layer architecture can be seen as one of the possible ways for addressing the *long-term autonomy* (LTA) problem for robotic systems [9]. In general, LTA can be seen as: (i) the ability of a robotic system to perform reliable operations for long periods of time under changing and unpredictable environmental conditions; (ii) the capability of *increasing its knowledge* about the working environment as time passes. Within this framework of LTA, the integration of IM-learning capabilities is one of the possible "enablers" for long-term robot autonomy[9]. For this reason, within the decisional layer of the IMPACT architecture has a long-term autonomy manager (LTA-M), see Figure 1, which provides a set of strategies to coordinate the achievements of both *extrinsic* and *intrinsic goals*, i.e., the mission goals, coming from outside Vs. the internal goals generated by curiosity driven behaviours [16, 5].

Starting from the system proposed in [2], we extended the design of the ROSPlan system with the integration of IM-learning capabilities as it follows. ROSPlan includes two main ROS[2] [14] nodes: (1) the Knowledge Base (KB) and the (2) Planning System (PS). KB is a collection of interfaces, and is intended to collate the up-to-date model of the environment. The Planning System acts as a wrapper for the internal planner such that: (1) builds the initial state automatically - as a PDDL problem instance - from the knowledge stored in the Knowledge Base; (2) passes the PDDL problem instance to the planner; (3) dispatches each action, deciding when to reformulate and *re-plan*. ROSPlan substantially implements the above introduced *Sense-Plan-Act* (SPA) autonomous deliberation and execution paradigm, whereas we implement a *Discover-Plan-Act* (DPA) cycle, which represents a significant extension of the SPA cycle by means of a more general open-ended learning step (*Discover*) such that: (i) it refers (as in ROSPlan) to the process of populating the Knowledge Base from sensor data; (ii) it represents also the process of *autonomous enrichment of the planning domain* by adding knowledge on new states and operators expressed in high-level symbols.

The Functional Layer. A functional layer (also called skill layer in [1]), which has access to the system's sensors and actuators and provides reactive behaviour which is robust even under environmental disturbances, for example with the help of closed-loop control. In IMPACT the GRAIL [15] system works as functional layer. In particular, as shown in Figure 1 GRAIL has a twofold role. On the one hand, within the functional layer, the block "GRAIL-C" will provide the set of *stable* controllers (i.e., the *experts*, see [15]) available in IMPACT for action execution. On the other hand, the block "GRAIL-IM-Learning" provides the autonomous learning component of new skills based on self-generated goals driven by intrinsic motivations (intrinsic goals).

The Executive Layer. The executive layer mediates between the decisional and functional layer, i.e. it activates or deactivates the reactive operations according to the deliberator's specification. In general, it represents the interface between the decisional and the functional levels. It controls and coordinates the execution of the functions distributed over the various functional level modules (i.e., the *experts*, see [15]) according to the task requirements. The above role, that is, the process of mapping symbolic and

[2] Robot Operating System (ROS) - https://www.ros.org/

abstract plans to continuous actions into the real working domain, is well recognised in the literature [1, 4, 6]. However, we have also inserted at the same layer the subsystem (called "KoPro+", see [8]) for the automatic translation of the newly acquired skills, from a low-level sub-symbolic representation (generated by "GRAIL-IM-Learning") to a high-level symbolic representation. In some sense, the "KoPro+" module has a *dual role* in comparison to the Executor; in fact, the learning process can be seen as opposite to the execution process, as it maps low-level sub-symbolic representation into abstract symbolic representation.

3 Operational Evaluation

The validation and verification of the IMPACT framework was carried out in the form of two demonstration test cases.

The *Rover* scenario, which demonstrates how the IMPACT system can discover new ways to reach an already known effect by applying the procedure (called KoPro+, see [8]) for the automatic translation of the newly acquired skills from the sub-symbolic level to the PDDL (symbolic) level. This scenario proposes a situation where the orientation mechanism of a planetary rover antenna has been damaged and the rover can no longer use it to point the antenna and establish a stable communication. In this case, our technology can be used to demonstrate how the rover is capable of enriching its planning domain with the necessary knowledge to orient the antenna merely using the locomotion capabilities, for example moving around the entire body in order to reach the correct attitude to gain and maintain communication, possibly exploiting terrain slopes and/or small rocks.

The *Robot Arm* scenario, which demonstrates the ability of the IMPACT system to acquire new ways to interact with the environment and integrate them in its planning domain. In this scenario, a robot equipped with a gripper actuator attached to a manoeuvrable arm tries to grasp a "vase shaped" rock whose size exceeds the max opening span of the gripper. The robot is thus not able to pick-up the rock with its basic grasping skill - however, upon failure, the IMPACT system will automatically trigger the learning of a new skill and the robot will at the end be able to pick-up the "vase shaped" rock by grasping it from its edge.

4 Conclusion and Future Work

We propose an extended version of the well-known three-layer architecture used in system robotics, extending the SPA cycle with a more general open-ended learning step (*Discover*) acquiring new knowledge from the external environment. Two different functionalities has been added to the classical three-layers architecture: (i) we have connected a goal-discovering and skill-learning robotic architecture (GRAIL) see [15] to the symbolic abstraction procedure proposed in [8], creating a processing pipeline from the low-level direct interaction of the agent with the environment, to the corresponding symbolic representation of the same environment; (ii) we have integrated a long-term autonomy manager (LTA-M) in the IMPACT architecture, which provides a set of strategies to coordinate the achievement of both *extrinsic* and *intrinsic goals*.

Possible directions of future work are: (i) the integration of symbolic planning and open-ended learning to increase the ability on one agent to autonomously acquire new skills (*bootstrap learning* [3]) ; (ii) to use a different *simulation engine*, for example the 3DROV environment [13] developed by ESA.

References

1. Bonasso, R.P., Firby, R.J., Gat, E., Kortenkamp, D., Miller, D.P., Slack, M.G.: Experiences with an architecture for intelligent, reactive agents. Journal of Experimental & Theoretical Artificial Intelligence **9**(2-3), 237–256 (1997). https://doi.org/10.1080/095281397147103, https://doi.org/10.1080/095281397147103
2. Cashmore, M., Fox, M., Long, D., Magazzeni, D., Ridder, B., Carrera, A., Palomeras, N., Hurtos, N., Carreras, M.: Rosplan: Planning in the robot operating system. In: ICAPS 2015, the 25th International Conference on Automated Planning and Scheduling (2015), https://www.aaai.org/ocs/index.php/ICAPS/ICAPS15/paper/view/10619
3. Doncieux, S., Filliat, D., Diaz-Rodriguez, N., Hospedales, T., Duro, R., Coninx, A., Roijers, D.M., Girard, B., Perrin, N., Sigaud, O.: Open-ended learning: A conceptual framework based on representational redescription. Frontiers in Neurorobotics **12**, 59 (2018). https://doi.org/10.3389/fnbot.2018.00059, https://www.frontiersin.org/article/10.3389/fnbot.2018.00059
4. Gat, E.: Three-layer architectures. In: Kortenkamp, D., Bonasso, R.P., Murphy, R. (eds.) Artificial Intelligence and Mobile Robots, pp. 195–210. MIT Press, Cambridge, MA, USA (1998), http://dl.acm.org/citation.cfm?id=292092.292130
5. Hester, T., Stone, P.: Intrinsically motivated model learning for developing curious robots. Artificial Intelligence **247**, 170 – 186 (2017). https://doi.org/https://doi.org/10.1016/j.artint.2015.05.002, http://www.sciencedirect.com/science/article/pii/S0004370215000764, special Issue on AI and Robotics
6. Ingrand, F., Lacroix, S., Lemai-Chenevier, S., Py, F.: Decisional autonomy of planetary rovers. Journal of Field Robotics **24**(7), 559–580 (2007). https://doi.org/10.1002/rob.20206, https://onlinelibrary.wiley.com/doi/abs/10.1002/rob.20206
7. Koenig, N., Howard, A.: Design and use paradigms for gazebo, an open-source multi-robot simulator. In: 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566). vol. 3, pp. 2149–2154 vol.3 (Sept 2004). https://doi.org/10.1109/IROS.2004.1389727
8. Konidaris, G., Kaelbling, L.P., Lozano-Perez, T.: From skills to symbols: Learning symbolic representations for abstract high-level planning. Journal of Artificial Intelligence Research **61**, 215–289 (2018), http://lis.csail.mit.edu/pubs/konidaris-jair18.pdf
9. Kunze, L., Hawes, N., Duckett, T., Hanheide, M., Krajník, T.: Artificial intelligence for long-term robot autonomy: A survey. IEEE Robotics and Automation Letters **3**(4), 4023–4030 (Oct 2018). https://doi.org/10.1109/LRA.2018.2860628
10. Mcdermott, D., Ghallab, M., Howe, A., Knoblock, C., Ram, A., Veloso, M., Weld, D., Wilkins, D.: PDDL - The Planning Domain Definition Language. Tech. rep., CVC TR-98-003/DCS TR-1165, Yale Center for Computational Vision and Control (1998), http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.37.212
11. Oddi, A., Baldassarre, G., Rasconi, R., Santucci, V.G., Sartor, G., Cartoni, E.: Sviluppo di architetture software innovative per l'autonomia in ambito spaziale. In: Workshop AI for Space, Ital-IA 2019, Convegno Nazionale CINI sull'Intelligenza Artificiale, Roma, 18 Marzo 2019 (03 2019)

12. Oddi, A., Rasconi, R., Cartoni, E., Sartor, G., Baldassarre, G., Santucci, V.G.: Learning high-level planning symbols from intrinsically motivated experience (2019), arXiv:1907.08313
13. Poulakis, P., Joudrier, L., Wailliez, S., Kapellos, K.: 3drov: A planetary rover system design, simulation and verification tool. In: International Symposium on Artificial Intelligence, Robotics and Automation in Space i?SAIRAS 2008, Hollywood, USA, February 26-29, 2008 (02 2008)
14. Quigley, M., Conley, K., Gerkey, B.P., Faust, J., Foote, T., Leibs, J., Wheeler, R., Ng, A.Y.: Ros: an open-source robot operating system. In: ICRA Workshop on Open Source Software (2009)
15. Santucci, V.G., Baldassarre, G., Mirolli, M.: GRAIL: A Goal-Discovering Robotic Architecture for Intrinsically-Motivated Learning. IEEE Transactions on Cognitive and Developmental Systems **8**(3), 214–231 (Sept 2016). https://doi.org/10.1109/TCDS.2016.2538961
16. White, R.W.: Motivation reconsidered: The concept of competence. Psychological Review **66**(5), 297–333 (1959). https://doi.org/http://dx.doi.org/10.1037/h0040934