

Contents lists available at [ScienceDirect](https://www.sciencedirect.com)

Software Impacts

journal homepage: www.journals.elsevier.com/software-impacts

Original software publication

MERLIN2: MachinEd Ros 2 pLanINg

Miguel Á. González-Santmartá*, Francisco J. Rodríguez-Lera, Camino Fernández-Llamas,
Vicente Matellán-Olivera

Grupo de Robótica, Universidad de León, Campus Vegazana, s/n, 24007, León, Spain



ARTICLE INFO

Keywords:

Cognitive robotics
Hybrid architecture
Symbolic planning
Finite state-machines
Reactivity
Knowledge managing

ABSTRACT

Any service robot should be able to make decisions and schedule tasks to reach predefined goals such as opening a door or assisting users at home. However, these processes are not single short-term tasks anymore and it is required to set long-term skills for establishing a control architecture that allows robots to perform daily tasks. This paper presents MERLIN2, a hybrid cognitive architecture based on symbolic planning and state machine decision-making systems that allows performing robot behaviors. The architecture can run in any robot running ROS 2, the latest version of the Robot Operative System. MERLIN2 is available at <https://github.com/MERLIN2-ARCH/merlin2>.

Code metadata

Current code version	v1.0
Permanent link to code/repository used for this code version	https://github.com/SoftwareImpacts/SIMPAC-2023-26
Permanent link to Reproducible Capsule	https://codeocean.com/capsule/9822103/tree/v1
Legal Code License	GPL-3.0 License
Code versioning system used	Git
Software code languages, tools, and services used	ROS 2, Python3
Compilation requirements, operating environments & dependencies	Docker OR ROS 2, Python3
If available Link to developer documentation/manual	https://merlin2.readthedocs.io/en/latest/
Support email for questions	mgons@unileon.es

1. Introduction

A social robot requires a considerable number of components working together for assisting a person in daily tasks. There are a number of challenges to overcome in a real scenario. The dynamics of environment such as humans walking randomly, long-term operations periods or managing multiple goals complicate decision making on the robot.

Under these circumstances, behavior generation in autonomous robots is an open problem associated with artificial intelligence systems. Several paradigms gave rise to what is known as control architectures [1], particularly the authors' approach focus on those cognitively inspired. For instance, we can find deliberative architecture [2], focused on planning approaches or subsumption architecture [3] and reactive architectures [4], which face quick behavior generation. These

architectures and their combination have had more impact on the community: the three-layer architecture of Gat [5], which organizes the decision-making component of the architecture in three different layers; and the hybrid architectures [6], which merges deliberative and reactive concepts in order to fulfill short and long-term goals. As a result, it is possible to find in the research community several implementations of similar approaches, each one developed for its own purpose with issues for repeatability, making the picture of cognitive architectures [7] hard to follow and to deploy in real robots.

These days we can find specific approaches for previous paradigms developed in ROS [8]. From a deliberative perspective, planning approaches such as ROSPlan [9] or PlanSys2 [10] can be used to produce plans to achieve the high-level goals of the robot. These applications

The code (and data) in this article has been certified as Reproducible by Code Ocean: (<https://codeocean.com/>). More information on the Reproducibility Badge Initiative is available at <https://www.elsevier.com/physical-sciences-and-engineering/computer-science/journals>.

* Corresponding author.

E-mail addresses: mgons@unileon.es (M.Á. González-Santmartá), fjrodl@unileon.es (F.J. Rodríguez-Lera), cferll@unileon.es (C. Fernández-Llamas), vmato@unileon.es (V. Matellán-Olivera).

<https://doi.org/10.1016/j.simpa.2023.100477>

Received 24 January 2023; Received in revised form 3 February 2023; Accepted 4 February 2023

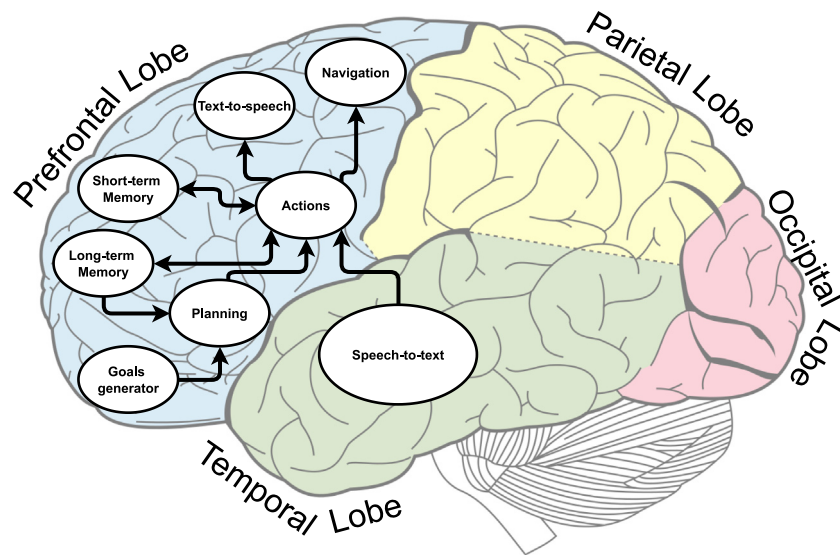


Fig. 1. Bioinspiration of MERLIN2 components.

are based on the Planning Domain Definition Language (PDDL) [11] that has been used to represent the environment of the robot using symbolic knowledge. However, these approaches do not give an easy way to modify the knowledge and do not provide persistence.

From a reactive perspective, there are several solutions, for instance, SMACH the hierarchical state machine generator [12] or behavior trees [13]. The former has alternatives such as FlexBee and the later has become the most extended paradigm in behavior generation in 2022. However, to integrate all these solutions in a single architecture can be difficult and for its own, some of them lack of functionalities such as visualization or symbolic knowledge manipulation.

This work posed the software implementation of a cognitive architecture called MERLIN2 (evolution of MERLIN [14]) fully compatible with ROS 2, the standard *de facto* in software development. It provides a cognitive architecture framework that suits the hybrid architecture paradigm but it also includes generic architectural tools for managing symbolic knowledge and scheduling robot behaviors.

2. Architecture

MERLIN2, which comes from MachinEd Ros 2 pLannINg, is a software architecture that represents the paradigm of a hybrid cognitive architecture for robots composed of four layers [14]. It comprises the main software functionalities for generating robot behaviors: knowledge manipulation, for world model representation; deliberative decision making system, for accomplish missions and behavior generation for answering dynamic environments. This organization of MERLIN2 resemble human cognitive capacities.

In nature, cognitive capacities such as short memory, perception or sensory-motor; are performed by some cerebrum lobes of the human brain [15,16]. Organizing MERLIN2 components as cerebrum cognitive functions gave us the sketch presented in Fig. 1. The human cerebrum is composed of four structures called prefrontal, temporal, occipital and parietal [17]. The prefrontal lobe has most of the modules from MERLIN2. It is in charge of memory, which is divided into long-term and short-term. The long-term memory, which represents the state of the environment, corresponds with the knowledge base managed by KANT. The short-term memory corresponds with the blackboards of the state machines of YASMIN. Besides, long-term memory is used in the planning process to choose the actions that the robot has to use to achieve its goals. These actions are implemented with YASMIN state machines and can also manage the knowledge of long-term memory. Finally, actions can use the navigation, text-to-speech and speech-to-text modules to interact with the environment and humans. The

prefrontal lobe is also in charge of motor control, navigation, and speech production, text-to-speech; while the temporal lobe is in charge of the audio processing, speech-to-text.

These cognitive functions are translated to the classic cognitive architecture depicted in Fig. 2. It presents in detail the MERLIN2 architecture technically divided into four layers classically associated with hybrid architectures and multi-layer cognitive proposals.

2.1. Mission layer

This layer is in charge of generating the high-level goals of the robots. These goals are related to robot duty, particular works that are internally or externally motivated, which means that are bred from the proprioceptive or exteroceptive stimulus (low battery status or human request).

The software component at this layer sets the goals of the robot in a given instant. That goal starts from an initial state made of current environment facts and sets the facts that the robots want to be true.

2.2. Planning layer

This layer is in charge of generating plans to achieve high-level goals previously created. It is associated with the classic Deliberative subsystem and requires formal representation of the environment, plan solvers, task dispatcher, and execution [2].

MERLIN2 proposes a set of five components for this layer:

- Knowledge Base: this component is the knowledge base of KANT, which corresponds with long-term memory. It stores the knowledge of the environment. There are two types of knowledge base depending on the storage type: a ROS 2 node that stores in process memory the knowledge and a MongoDB [18] database, which is more persistent.
- PDDL Generator: this component is in charge of generating the PDDL text from the knowledge base.
- Planner: this component uses the PDDL text to create plans. Several PDDL planners can be used. The available planners are POPF [19], SMTP [20], unified-planning [21] and VHPOP [22].
- Plan Dispatcher: this component is in charge of executing the actions of the generated plan. It also has to cancel the current action if needed.
- Executor: this component is a facade between this layer and the Mission Layer. It executed its component of this layer and cancel the Plan Dispatcher.

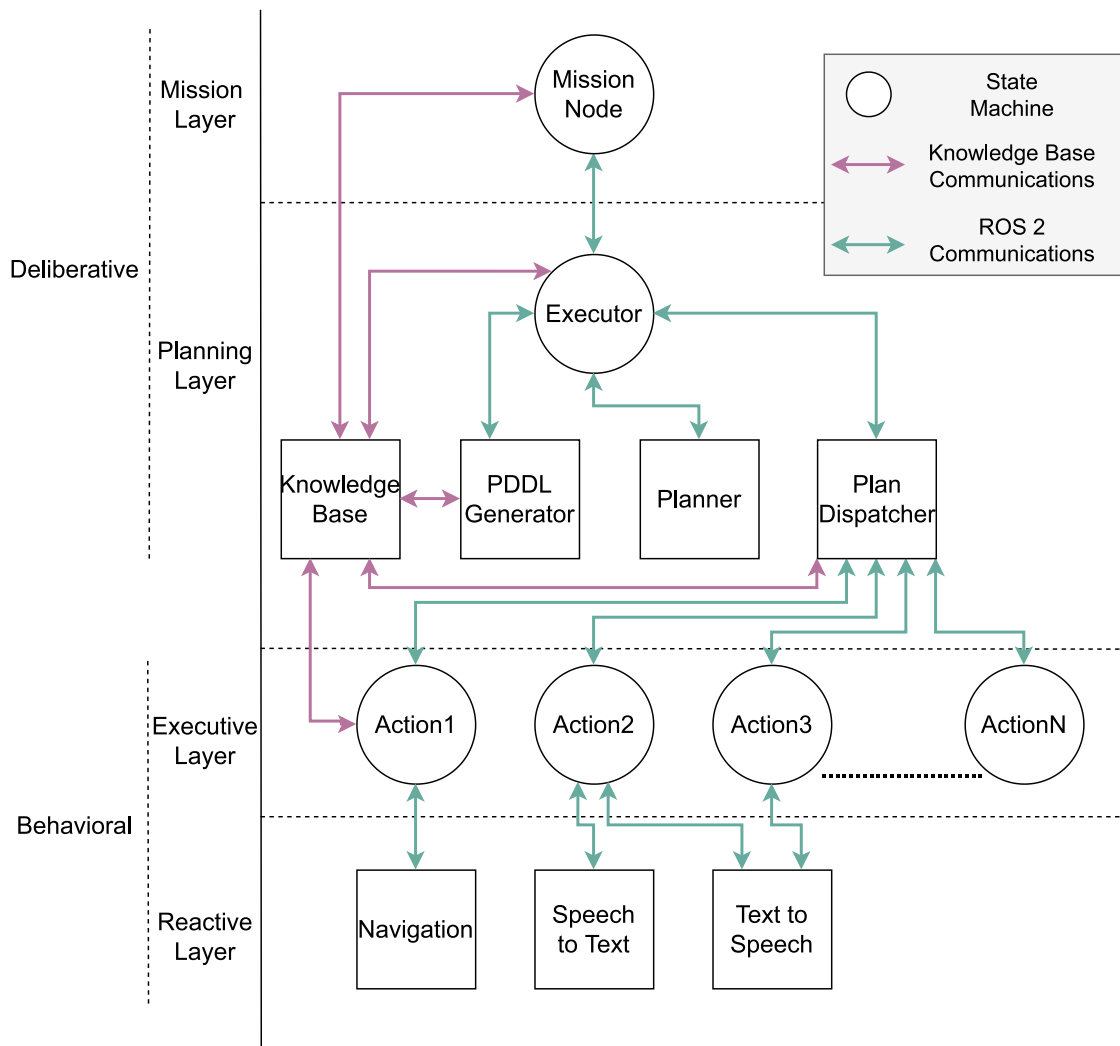


Fig. 2. MERLIN2 architecture.

2.3. Executive layer

This layer is composed of the actions that the robot can carry out. When an action is instantiated, its associated knowledge is saved in the knowledge base. Besides, these actions are implemented as YASMIN state machines. As has been mentioned before, the blackboard of the state machines is the short-term memory, that stores the initial objects needed by the PDDL action and the partial solution of the state machine. Besides, YASMIN has a viewer that can be used to monitor the actions at a high level. Finally, new actions can be developed so that the robot can perform other tasks. As a result, more complex behaviors can be implemented.

2.4. Reactive layer

This layer provides the set of software components that bring bounded and specific applications that would be considered robot skills such as navigation (world exploration), text-to-speech (talking) or sound recognition (listening). This version of MERLIN2 includes three specific components that are suitable for validating the use of the cognitive architecture:

- Navigation: this component corresponds with Nav2 [23], which is the navigation system of ROS 2.
- Speech-to-text: this component is in charge of listening for speech and converting it to text.

- Text-to-speech: this component is in charge of producing speech from a sentence.

3. Software impact

Cognitive architectures in robotics face the development and organization of software components to imitate the human mind. It is an active field of research where different software technologies and mechanisms can be found together. Some of them have been presented before, such as state machines and planning systems.

MERLIN2 cognitive architecture contains the elements that can be beneficial to research conducted in robotics competitions, in real¹ and simulated² environments, including but not limited to RoboCup or SciRoc. It allows the management of symbolic and sub-symbolic information through the use of KANT, generating Finite State Machine with YASMIN, and a mechanism for integrating robot tasks.

1. MERLIN2 aims to speed up the process of producing robot behaviors to face the daily short and long-term tasks of a service robot using ROS 2.
2. MERLIN2 combines four high-value tools for robotics developers: YASMIN, KANT, a Generic Planner System and a visual monitoring engine for mission and action tracking.

¹ <https://youtu.be/BfbsLQNG0gg>

² <https://youtu.be/ZKOqBHytXAI>

3. The four layers of MERLIN2 allow the generation of behaviors for performing complex robot tasks. The Mission Layer is supported by a mission engine which allows to the creation of an initial problem and the capability to create and execute goals as a consequence of developed actions. The Planning Layer manages symbolic task-solving. The Executive Layer has the actions, which are the result of associating and grouping skills based on two concepts: YASMIN Finite State Machine and (2) skill reuse. The Reactive Layer deploys the robot's skills, adding a new one is a straightforward process. The current repository has examples with three basic robot skills: Navigation, STT and TTS.
4. MERLIN2 adds the feature for symbolic knowledge manipulation through KANT supported on DTO and DAO patterns. KANT provides two approaches: one for bringing knowledge persistence, using MongoDB; while the other uses in-memory storage. KANT engine makes it accessible to the developer to update/change the technology used.
5. MERLIN2 includes a generic planning system based on PDDL (Planning Layer). This planning system automatically extracts PDDL from the knowledge base, generates plans and executes the actions in a transparent manner for the developer. The solver motor includes simplified access to a set of PDDL solvers attending the version required.

The results enabled by the software individually and the whole architecture have been reported in the following academic publications:

- Miguel Á. González-Santamarta, Francisco J. Rodríguez-Lera, Francisco Martín, Camino Fernández, and Vicente Matellán. Kant: A tool for grounding and knowledge management. In José Manuel Ferrández Vicente, José Ramón Álvarez-Sánchez, Félix de la Paz López, and Hojjat Adeli, editors, *Bio-inspired Systems and Applications: from Robotics to Ambient Intelligence*, pages 452–461, Cham, 2022. Springer International Publishing. [24]
- Miguel Á. González-Santamarta, Francisco J. Rodríguez-Lera, Vicente Matellán-Olivera, and Camino Fernández-Llamas. Yasmin: Yet another state machine. In Danilo Tardioli, Vicente Matellán, Guillermo Heredia, Manuel F. Silva, and Lino Marques, editors, *ROBOT2022: Fifth Iberian Robotics Conference*, pages 528–539, Cham, 2023. Springer International Publishing. [25]
- Miguel Á. González-Santamarta, Francisco J. Rodríguez-Lera, Claudia Álvarez-Aparicio, Ángel M. Guerrero-Higuera, and Camino Fernández-Llamas. MERLIN a cognitive architecture for service robots. *Applied Sciences*, 10(17):5989, aug 2020. [14]

4. Limitations and further improvements

Finally, several future works are taken into account. MERLIN2 needs a perception module that enables more complex behaviors. Besides, it can be used to perform grounding and anchoring [26] improving the long-term and short-term memory of MERLIN2.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

This work has been partially funded by an FPU fellowship provided by the Spanish Ministry of Universities (FPU21/01438) and the Grant

PID2021-126592OB-C21 funded by MCIN/AEI/10.13039/501100011033 and by ERDF A way of making Europe.

References

- [1] Danial Nakhaeina, S. Hong Tang, S.B. Mohd Noor, O. Motlagh, A review of control architectures for autonomous navigation of mobile robots, *Int. J. Phys. Sci.* 6 (2) (2011) 169–174.
- [2] Félix Ingrand, Malik Ghallab, *Deliberation for autonomous robots: A survey*, *Artificial Intelligence* 247 (2017) 10–44.
- [3] Rodney A. Brooks, *New approaches to robotics*, *Science* 253 (5025) (1991) 1227–1232.
- [4] R. Peter Bonasso, R. James Firby, Erann Gat, David Kortenkamp, David P. Miller, Mark G. Slack, *Experiences with an architecture for intelligent, reactive agents*, *J. Exp. Theoret. Artif. Intell.* 9 (2–3) (1997) 237–256.
- [5] Erann Gat, R. Peter Bonasso, Robin Murphy, et al., *On three-layer architectures*, *Artif. Intell. Mob. Robots* 195 (1998) 210.
- [6] Ronald Arkin, Tucker Balch, AuRA: Principles and practice in review, *J. Exp. Theoret. Artif. Intell.* 9 (1970).
- [7] Peijun Ye, Tao Wang, Fei-Yue Wang, *A survey of cognitive architectures in the past 20 years*, *IEEE Trans. Cybern.* 48 (12) (2018) 3280–3290.
- [8] Morgan Quigley, Ken Conley, Brian Gerkey, Josh Faust, Tully Foote, Jeremy Leibs, Rob Wheeler, Andrew Y Ng, et al., *ROS: an open-source robot operating system*, in: *ICRA Workshop on Open Source Software*, vol. 3, no. 3.2, Kobe, Japan, 2009, p. 5.
- [9] Michael Cashmore, Maria Fox, Derek Long, Daniele Magazzeni, Bram Ridder, Arnau Carrera, N. Palomeras, N. Hurtós, Marc Carreras, *ROSplan: Planning in the robot operating system*, in: *Proceedings International Conference on Automated Planning and Scheduling, ICAPS*, vol. 2015, 2015, pp. 333–341.
- [10] Francisco Martín, Jonatan Ginés, Francisco J. Rodríguez, Vicente Matellán, *PlanSys2: A planning system framework for ROS2*, in: *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2021*, Prague, Czech Republic, September 27 - October 1, 2021, IEEE, 2021.
- [11] Maria Fox, Derek Long, *PDDL2.1: An extension to PDDL for expressing temporal planning domains*, *J. Artif. Intell. Res. (JAIR)* 20 (2003) 61–124.
- [12] Jonathan Bohren, Steve Cousins, *The SMACH high-level executive*, *IEEE Robot. Autom. Mag.* 17 (4) (2011) 18–20.
- [13] Michele Colledanchise, Lorenzo Natale, *On the implementation of behavior trees in robotics*, *IEEE Robot. Autom. Lett.* 6 (3) (2021) 5929–5936.
- [14] Miguel Á. González-Santamarta, Francisco J. Rodríguez-Lera, Claudia Álvarez-Aparicio, Ángel M. Guerrero-Higuera, Camino Fernández-Llamas, *MERLIN a cognitive architecture for service robots*, *Appl. Sci.* 10 (17) (2020) 5989.
- [15] Richard S.J. Frackowiak, *Human Brain Function*, Elsevier, 2004.
- [16] Bryan Kolb, Ian Q. Whishaw, G. Campbell Teskey, *An Introduction to Brain and Behavior*, Worth New York, 2001.
- [17] Geraldine Ed Dawson, Kurt W. Fischer, *Human Behavior and the Developing Brain*, The Guilford Press, 1994.
- [18] S. Bradshaw, K. Chodorow, E. Brazil, *MongoDB: The Definitive Guide: Powerful and Scalable Data Storage*, in: *The Expert's Voice in Open Source*, O'Reilly Media, Incorporated, 2019.
- [19] Amanda Coles, Andrew Coles, Maria Fox, Derek Long, *Forward-chaining partial-order planning*, in: *ICAPS 2010 - Proceedings of the 20th International Conference on Automated Planning and Scheduling*, 2010, pp. 42–49.
- [20] Michael Cashmore, Maria Fox, Derek Long, Daniele Magazzeni, *A compilation of the full PDDL+ language into SMT*, in: *Workshops at the Thirtieth AAAI Conference on Artificial Intelligence*, 2016.
- [21] aiplan4eu, *AIPLAN4EU/unified-planning: The AIPLAN4EU Unified Planning Library*, GitHub.
- [22] Håkan L.S. Younes, Reid G. Simmons, *VHPOP: Versatile heuristic partial order planner*, *J. Artificial Intelligence Res.* 20 (2003) 405–430.
- [23] Steve Macenski, Francisco Martín, Ruffin White, Jonatan Ginés Clavero, *The marathon 2: A navigation system*, in: *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS*, 2020.
- [24] Miguel Á. González-Santamarta, Francisco J. Rodríguez-Lera, Francisco Martín, Camino Fernández, Vicente Matellán, *KANT: A tool for grounding and knowledge management*, in: José Manuel Ferrández Vicente, José Ramón Álvarez-Sánchez, Félix de la Paz López, Hojjat Adeli (Eds.), *Bio-Inspired Systems and Applications: From Robotics to Ambient Intelligence*, Springer International Publishing, Cham, 2022, pp. 452–461.
- [25] Miguel Á. González-Santamarta, Francisco J. Rodríguez-Lera, Vicente Matellán-Olivera, Camino Fernández-Llamas, *YASMIN: Yet another state machine*, in: Danilo Tardioli, Vicente Matellán, Guillermo Heredia, Manuel F. Silva, Lino Marques (Eds.), *ROBOT2022: Fifth Iberian Robotics Conference*, Springer International Publishing, Cham, 2023, pp. 528–539.
- [26] Silvia Coradeschi, Amy Loutfi, Britta Wrede, *A short review of symbol grounding in robotic and intelligent systems*, *KI-Künstliche Intelligenz* 27 (2) (2013) 129–136.