

Flexible and Adaptive Manufacturing by Complementing Knowledge Representation, Reasoning and Planning with Reinforcement Learning

1st Matthias Mayr
 Department of Computer Science
 Lund University
 matthias.mayr@cs.lth.se

2nd Faseeh Ahmad
 Department of Computer Science
 Lund University
 faseeh.ahmad@cs.lth.se

3rd Volker Krueger
 Department of Computer Science
 Lund University
 volker.krueger@cs.lth.se

I. INTRODUCTION

Shifting from mass manufacturing towards greater customizations and smaller batch sizes requires specialized architectures and solutions capable of effectively managing this change. A system designed for this transition must not only exhibit a high degree of flexibility but it must also be able to cope with the added complexity introduced by the integration of robotic systems such as manipulators and their different hardware. Finally, it is desirable that even for contact-rich manufacturing tasks, the system should be able to learn from data and improve the execution. In the light of these objectives, we present our work that leverages task-level planning and reasoning and combines it with reinforcement learning. The former provides a structure for a task, while the latter allows to learn and improve the execution from interaction data. This synergy creates a comprehensive pipeline that translates high-level planning goals, often originating from a manufacturing execution system, into tangible improvements in execution within a multi-objective setting on the robot system.

In the following sections, we provide an overview of the skill-based system *SkiROS2* and its planning and reasoning capabilities. We introduce the integration of multi-objective reinforcement learning and our approach to address reinforcement learning for variations of tasks.

II. SKIROS2 - A SKILL-BASED ROBOT SYSTEM

SkiROS2 [1] is a skill-based robot control platform for the Robot Operating System (ROS). Its primary application lies within semi-structured workspaces where an initial understanding of the world exists, but requires ongoing correction and adaptation throughout the execution process. An overview of the architecture is shown in Figure 1.

A core component of *SkiROS2* is its world model, which employs a Resource Description Framework (RDF) graph as the foundation for knowledge representation. Ontologies define the known concepts, properties and relations. Additionally, a scene holds the concrete instances (vocabulary) of these concepts. This knowledge can additionally be used to catch bugs in skills [2]. One or more skill managers are responsible for not only retrieving skills from skill libraries but also executing them on a robot system. Lastly, the task manager

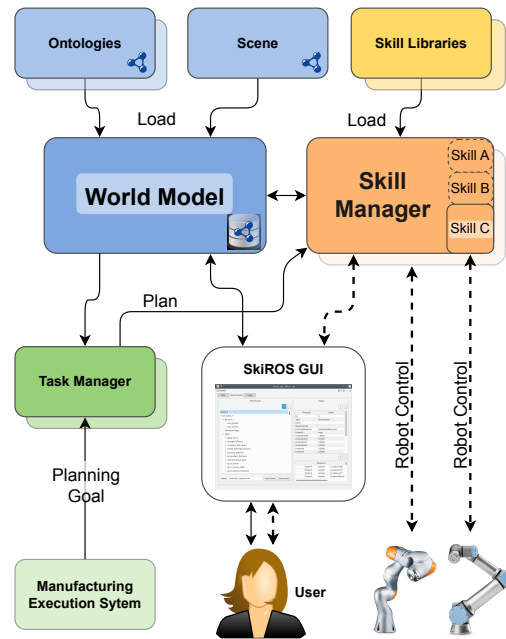


Fig. 1. An outline of the *SkiROS2* architecture. The world model stores the knowledge about the relations, environment and the skills. The skill manager loads and executes the skills. Dashed lines show control flows and solid lines information flows. Shaded blocks indicate possible multiple instances.

plays a pivotal role in receiving planning goals and generating plans, taking into account the current state of the world model.

SkiROS2 has a broad definition of a *skill*. This allows to integrate a diverse range of existing solutions, ranging from motion planning to deep-learning-based vision. A *skill description* describes the skill on a semantic level, specifying its parameters and the pre-, hold-, and post-conditions. Following the extended behavior tree (BT) formalism [3], these conditions can be directly employed to construct a planning domain. *Skill implementations* can implement a skill description. This implementation can take the form of either a semantically atomic *primitive skill* or a *compound skill*. The latter allows the combination of multiple other skills within a BT. A comparison with other skill-based systems can be found in [1].

III. PLANNING, KNOWLEDGE REPRESENTATION AND REASONING

SkiROS2 has a well developed toolset to support flexible manufacturing. The task manager can receive planning goals in the widely supported Planning Domain Definition Language (PDDL) and automatically generate a domain and problem description based on the loaded skills and the current state of the world model.

The knowledge representation within the world model, utilizing an RDF graph, enables an explicit representation of available knowledge that can be easily and automatically generated and modified. This architecture explicitly separates the task knowledge from the skill implementation, promoting adaptability across various robot hardware and tasks [4]. In this light, we use behavior trees and motion generators (BTMG) policy representation for skill implementation to solve tasks [5]. Finally, the relations defined by the ontologies facilitate reasoning capabilities, such as spatial reasoning [5], in the parameterization and execution of tasks.

IV. REINFORCEMENT LEARNING

A recent line of research has explored to learn parameters of skills [6], [7], [8], [9]. Specifically, in [7], [8] the integration of reinforcement learning (RL) with robot manipulators into the planning and reasoning pipeline is introduced. As shown in Figure 2, the operator is actively involved in the creation of the RL scenario and the final selection of the policy. Formulating RL problems is challenging and is specifically difficult in the industrial domain that has many requirements, such as safety constraints. With this learning formulation allowing for multi-objective RL, it actively acknowledges that there is not only a single performance indicator. This eases the learning scenario design and gives the operator a selection of policies along the Pareto frontier to choose from. In [9], an integration of parameter priors for the optimum is introduced. This allows an operator or another source of experience to specify supposedly optimal regions in the parameter space to speed up the search. The methods have been shown to allow for learning contact-rich tasks with compliant control [10] either directly on the real system [6], [9] or in simulation and transferring to the real system [6], [8], [9].

V. LEARNING FOR TASK VARIATIONS

As stated in the introduction, adaptation to different task configurations is important. While *SkiROS2* as a skill-based system fully supports to adapt to task variations, depending on the influence of the learned skill parameters, the policies learned in [6], [7], [8], [9] can only transfer for some changes in the tasks. This has been identified in [11] and addressed in [12] as a supervised learning problem. In short, in [12] a set of BTMG parameters are learned for random task variations using the previously introduced learning pipeline. The results of all learning episodes in all variations are used to train a model that combines Gaussian Processes and a weighted support vector machine classifier. This model learns both the reward and feasibility functions based on skill and task

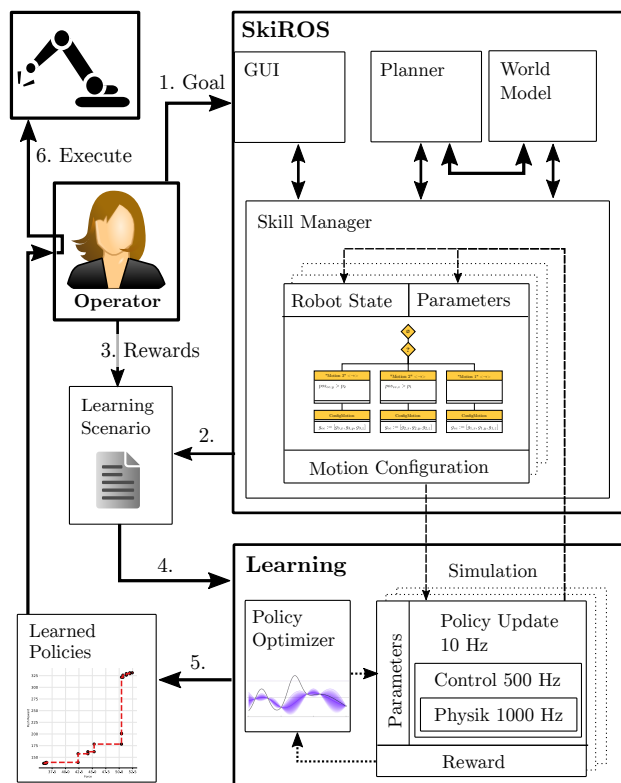


Fig. 2. The architecture of the system that depicts the pipeline: (1) The operator enters the goal state; (2) a learning scenario for the plan is created; (3) rewards and hyperparameters are specified; (4) learning is conducted using the skills and the information in the world model; (5) after policy learning, the operator can choose which policies to execute on the real system (6).

variation parameters. Rewards represent the task performance, while feasibility predicts successful task execution with specific parameter combinations. We further optimize the model to derive a feasible policy with maximum reward. When a parameterization for an unseen task variation is requested, a combination of the planning and reasoning capabilities of *SkiROS2* and the inference of the learned model is used to provide a full parameterization. The results and comparisons with baseline methods are presented in [12].

VI. CONCLUSIONS

Modern manufacturing needs solutions that are flexible and adaptive. We have outlined a way to combine task-level planning, knowledge representation and reasoning with reinforcement learning techniques. Not only does this combination allow for easy adaptation of tasks, we have also shown how robot systems can learn and improve the task execution. The formulation as a multi-objective learning problem eases the learning problem definition and allows the operator to choose a suitable policy. The possibility to integrate user priors into the search does not only allow an experienced operator to state promising regions, but has also shown to be able to speed up learning and make it safer.

Finally, the recent introduction of a machine learning model that can appropriately address task variations for the learned skill parameters makes it more fit for the use in future factories.

REFERENCES

- [1] M. Mayr, F. Rovida, and V. Krueger, “Skiros2 - a skill-based robot control platform for ros,” in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2023*.
- [2] M. Rizwan, R. Caldas, C. Reichenbach, and M. Mayr, “Ezskiros: A case study on embedded robotics dsls to catch bugs early,” in *2023 IEEE/ACM 5th International Workshop on Robotics Software Engineering (RoSE)*. IEEE, 2023, pp. 61–68.
- [3] F. Rovida, B. Grossmann, and V. Krüger, “Extended behavior trees for quick definition of flexible robotic tasks,” in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017, pp. 6793–6800.
- [4] M. Mayr, F. Ahmad, A. Duerr, and V. Krueger, “Using knowledge representation and task planning for robot-agnostic skills on the example of contact-rich wiping tasks,” in *2023 IEEE 18th International Conference on Automation Science and Engineering (CASE)*. IEEE, 2023.
- [5] F. Rovida, D. Wuthier, B. Grossmann, M. Fumagalli, and V. Krüger, “Motion Generators Combined with Behavior Trees: A Novel Approach to Skill Modelling,” in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 5964–5971.
- [6] M. Mayr, K. Chatzilygeroudis, F. Ahmad, L. Nardi, and V. Krueger, “Learning of Parameters in Behavior Trees for Movement Skills,” in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2021*.
- [7] M. Mayr, F. Ahmad, K. Chatzilygeroudis, L. Nardi, and V. Krueger, “Combining planning, reasoning and reinforcement learning to solve industrial robot tasks,” *IROS 2022 Workshop on Trends and Advances in Machine Learning and Automated Reasoning for Intelligent Robots and Systems*, 2022.
- [8] —, “Skill-based multi-objective reinforcement learning of industrial robot tasks with planning and knowledge integration,” in *2022 IEEE International Conference on Robotics and Biomimetics (ROBIO)*. IEEE, 2022, pp. 1995–2002.
- [9] M. Mayr, C. Hvarfner, K. Chatzilygeroudis, L. Nardi, and V. Krueger, “Learning skill-based industrial robot tasks with user priors,” in *2022 IEEE 18th International Conference on Automation Science and Engineering (CASE)*. IEEE, 2022, pp. 1485–1492.
- [10] M. Mayr and J. M. Salt-Ducaju, “A c++ implementation of a cartesian impedance controller for robotic manipulators,” *arXiv preprint arXiv:2212.11215*, 2022.
- [11] F. Ahmad, M. Mayr, E. A. Topp, J. Malec, and V. Krueger, “Generalizing behavior trees and motion-generator (btmg) policy representation for robotic tasks over scenario parameters,” in *2022 IJCAI Planning and Reinforcement Learning Workshop*, 2022.
- [12] F. Ahmad, M. Mayr, and V. Krueger, “Learning to adapt the parameters of behavior trees and motion generators to task variations,” in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2023.