Symbolic Robot Commanding utilizing Physical Properties - System Overview

Michael Spangenberg and Dominik Henrich

Chair for Applied Computer Science III
Robotics and Embedded Systems
University of Bayreuth, D-95440 Bayreuth, Germany
E-Mail: {michael.spangenberg | dominik.henrich}@uni-bayreuth.de

Abstract. One long term goal of artificial intelligence and robotics research is the development of robot systems, which have approximately the same cognitive, communicational, and handling abilities like humans. This yields several challenges for future robot systems. For instance in the field of communicational abilities, future robot systems have to bridge between natural communication methods of the human, primarily utilizing symbols like words or gestures, and the natural communication methods of artificial systems, primarily utilizing low-level subsymbolic control interfaces. In this work, we outline a system which utilizes physical properties, respectively physical effects for the mapping between a high-level symbolic user interface and a low-level subsymbolic robot control interface.

1 Introduction

There are several long term goals in current robotic research. One is the development of robot systems, which have approximately the same cognitive, communicational, and handling abilities like humans. As part of this ongoing development, application domains for robot systems shall be expanded, from industrial settings with separated working cells, fixed object positions, and preprogrammed motions towards a flexible usage in small or medium-sized enterprises (SMEs) or private households. This sets additional requirements to the abilities of future robot systems. In the field of cognitive abilities, future robot systems must utilize appropriate sensors to extract information from the environment. In the field of handling abilities, future robot systems need action representations, which allow a flexible parameterization and execution of a specific task. In the field of communicational abilities, future robot systems must provide an intuitive and symbolic user interface.

The interaction between cognitive, communicational, and handling abilities is crucial for future robot systems. In Figure 1, potential tasks in SMEs or private households are visualized. Such tasks typically require the definition of sensor-based actions, which are defined utilizing a subsymbolic robot control interface like iTaSC [1] or manipulation primitives [2]. The definition of sensor based actions require expert knowledge in the domain of robotics, since the programmer

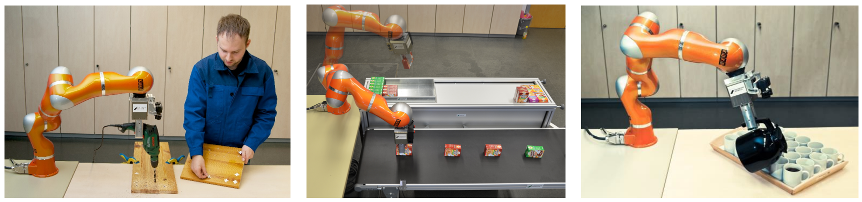

Fig. 1. Typical applications in SMEs or private households which require the execution of sensor based motions. From left to right: Drilling, Paletting, Pouring.

must define subsymbolic parameters like positions, forces, setpoints, or control strategies. In SMEs or private households, it cannot be assumed that this expert knowledge in robotics is available. Therefore, future robot systems must provide an intuitive user interface, which allows a symbolic communication. Such a robot system needs information about the semantics of the used symbols, for example executable actions or manipulable objects. Furthermore, the robot system must be able to extract the needed subsymbolic information from the environment, utilizing appropriate software components and sensors.

In the following sections, we give an overview of our system, which utilizes physical effects, respectively physical properties for the grounding of symbols and parameterization of subsymbolic sensor-based motions.

The remainder of the work is organized as follows: The related work is described in the next section. Here, an overview of robot systems utilizing a symbolic user interface is given. In Section 3, we give an overview of our system, outline the action representation based on verbalized physical effects, and describe the relations between the used symbols, physical parameters, and components for the extraction of the needed subsymbolic parameters from given symbolic instructions. At last, we describe our future work in Section 4.

2 Related Work

The problem of assigning semantics to symbolic tokens like words is known as the symbol grounding problem and was described by Harnad [3] with aspects from psychology and artificial intelligence. Since practical applications of artificial intelligence, for example in form of robots and intelligent systems, become more complex, also researchers from these domains have to consider about the problem of symbol grounding [4]. The grounding of symbols can be organized into two subtopics, physical symbol grounding [5], and social symbol grounding [6]. While social symbol grounding focuses on sharing symbols in populations of agents, physical symbol grounding focuses on building relations between sensor values and symbols. Since we want to extract subsymbolic physical parameters, we focus on physical symbol grounding in more detail. There are already systems, which can be operated utilizing symbolic commands. In general, such robot systems are either used within navigational [7],[8], [9], [10] or handling tasks [11], [12], [13], [14], [15], [16]. These systems can be categorized according to the extractable subsymbolic information. The first category of systems allows *no extraction* of subsymbolic information, i.e. they can only execute predefined instructions. The second category of systems is able to extract *geometric* information from known object identifiers utilizing an object database and an object recognition system. Systems of the third category can additionally extract *spatial relations* from symbolic instructions.

Because all of the described systems are based on action representations, which utilize geometric information, they do not need to extract *kinematic* and *dynamic* parameters like forces, torques, or energies. Our system is based on an action representation utilizing verbalized physical effects and manipulation primitive nets [17], which is parameterized by geometric, kinematic, and dynamic parameters, therefore we need to specify how to extract these quantities from a symbolic representation.

3 System Overview

An overview of our system architecture is shown in Figure 2. The system is build according to the 3T architecture [18], a common architecture for systems which have to transform between different types of representations. In case of our system, we need to transform a high-level symbolic user representation into a low-level subsymbolic robot control representation. Typically, these high-level representations cannot be mapped directly to a low-level robot control representation. Therefore, such systems consists of an additional transformation layer, which describes the mapping between the high-level user interface and the low-level robot control interface. In the following subsections, we outline the realization of the three tiers of our system.

3.1 User Layer

The main function of this layer is to provide high-level user interfaces, which allow the usage of the robot system by non-experts. Therefore, we focus on intuitive symbolic representations like a domain specific language (DSL) or a natural language interface. We introduced a domain-specific language for sensor-based actions in [19]. In the DSL, executable actions are described by verbal expressions, and parameterized by phrases. For instance, the DSL provides a sensor-based action `shove`, which takes a *noun phrase* and a *prepositional phrase* as parameter. This allows the user to specify an instruction like `shove("the red cube", "towards the gray box")`. Users are able to instruct executable actions to the robot system, without specifying low-level control parameters. Since these parameters are required for the execution, the robot system must be able to extract low-level parameters utilizing additional components like a knowledge base, action skeletons, or environment information gathered by sensors.

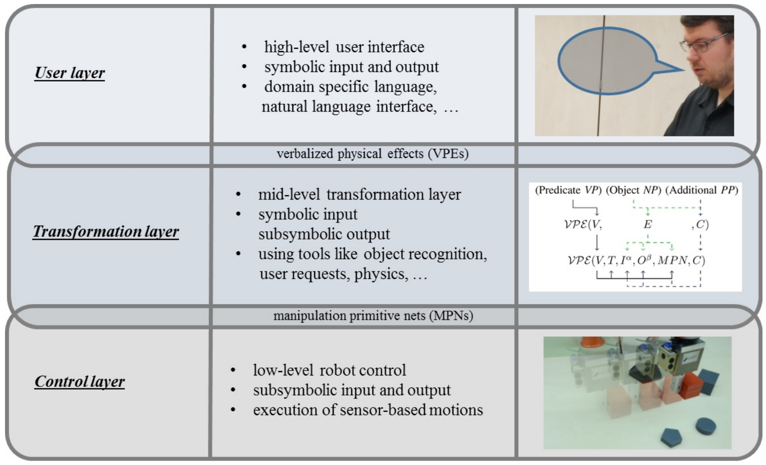

Fig. 2. Overview of the system architecture.

3.2 Transformation Layer

The next step is to transform the high-level user input into a suitable representation for the low-level robot control. The robot control typically consists of subsymbolical interfaces, which require the definition of parameters like set points, control strategies, or task frames. These parameter are not specified explicitly by the user, therefore this information must be specified implicitly based on the context, respectively based on the semantics of the used symbols. This information must be grounded to the robot system.

The main idea of our symbol grounding approach is based on the working hypothesis that object manipulation tasks consist of mechanical operations and can be described using the laws of physics, especially from the field of mechanics. If we analyze the function of a specific symbol, it represents either an executable action or a parameter for an action. Therefore, we describe the grounding of actions and parameters in the following paragraphs.

Action Grounding. The concept of verbalized physical effects $\mathcal{VPE}$s is used to describe executable actions in terms of physical effects. This representation is utilized for the linkage of symbolic instructions and sensor based motions, and the calculation of subsymbolic parameter from a given symbolic instruction. Furthermore, this concept is used for the identification of needed information and the automatic generation of temporal states, since instructions typically specify only the goal state of a task. In this subsection, we give an overview of the used physical quantities, principal physical effects $\mathcal{PPE}$s, and the mapping of a verbal expression to an specific $\mathcal{PPE}$.

Generally, seven base units are defined in ISO 30-0 [20]. Within an object manipulation task, mechanical base units *length* L, *mass* M and *time* T are manipulated. In addition to these base units, also derived units can be measured and manipulated, which can be categorized in *geometric*, *kinematic*, and *dynamic* units [21]. We use these physical quantities as parameter for a set of principal physical effects and define the five principal effects *absorb*, *change*, *transform*, *merge*, *split* on physical quantities ($\mathcal{PPE}$s).

The next step is to find a suitable verb for a principal physical effect, for example for the physical effects *transform a force into a length (displacement)*, *transform a momentum into a displacement*, or *absorb a force*. These terms are not intuitive to verbalize for a user. The most proper verb for each $\mathcal{PPE}$ can only be evaluated by collecting and analyzing empirical data, which is described in our previous work [22]. There, we collected the data in German, and use here an appropriate translation. For instance, the $\mathcal{PPE}$ *transform a force into a length (displacement)* is mapped to the $\mathcal{VPE}$ consisting of the verbal expression *to shove (schieben)*, the $\mathcal{PPE}$ *transform a momentum into a displacement* to the $\mathcal{VPE}$ consisting of the verbal expression *to push (stoßen)*, and the $\mathcal{PPE}$ *absorb a force* to the $\mathcal{VPE}$ consisting of the verbal expression *to touch (berühren)*. More details about the concept of verbalized physical effects are presented in [17].

Parameter Grounding. Besides executable actions, the semantics of the symbolic parameters have to be grounded to the robot system. These parameters are applied to the defined verbalized physical effects, therefore it is necessary to describe the semantics of the parameters in terms of physical properties. Based on an analysis of symbols, we introduced a physical dictionary for the grounding of symbols based on physical properties in [23].

The first task of the physical dictionary is to ground information about the symbol class and syntactic function of a specific symbol. This information is used to determine coherence between different symbols. Let a user instruct the natural language instruction *Stack the red cylinder on the blue cube!* With the grounded information, we can determine that the determiner *the* and the adjective *red* relates to the noun *cylinder*.

The second task of the dictionary is to ground information about the manipulated properties of a specific symbol. We analyzed that symbols can affect various properties, which can be specified in different degrees of determination. In general, a symbol describes either an object, a process, a relation, or a property. For instance, the class of adjectives describe properties of objects or the class of prepositions describe relations between objects. The degree of determination can be for example exact or within an interval. For instance, a symbol of type numeral describes a property exact, while an adjective describes a property typically by an interval.

Parameter Extraction. The next step towards an robotic execution of the instructed symbolic command is the subsymbolic parameter extraction, dependent on the actual context, respectively environment of the robot system. The extraction of the subsymbolical information is done by specific software components, which utilize the sensors of the actual robot system. Therefore, we expand our knowledge base with a component and a sensor submodule. The component submodule stores information about available extraction methods for physical quantities, and the sensor submodule stores information about the sensors utilized by the specific extraction method. All components share the same interface, which on the one hand allows us to integrate existing approaches in the overall system. On the other hand, the extraction of subsymbolical information is decoupled from the overall functionality and new components or sensors can easily be integrated in the knowledge base. Since there are typically more components and sensors for the extraction of a physical quantity available, we can define a criteria which describes the most suitable component for the actual situation of the environment. An evaluation of the symbol grounding and subsymbolical parameter extraction is described in [23].

3.3 Control Layer

As low-level robot control interface, we use manipulation primitives [2], respectively manipulation primitive nets [24]. In general, a manipulation primitive net is a graph representation of an sensor-based task, consisting of manipulation primitives as nodes and stopping criteria as edges. The definition of a manipulation primitive consists of a hybrid motion, a set of tool commands, and a set of termination criteria.

The hybrid motion describes the executable sensor-based motion based on a local coordinate system, called the task frame. For this task frame, a control strategy for each degree of freedom must be specified. Valid control strategies are for instance position or force control. The set of tool commands holds information about the used tool and the state of the tool. For instance, a gripper shall be opened or closed during the execution of the sensor-based motion. The set of termination criteria is a set of Boolean conditions, which are checked during the execution of a manipulation primitive. The execution of the actual manipulation primitive is stopped, when at least one termination criteria is fulfilled.

The relations between verbalized physical effects, effect parameters, and the mapping and parameterization of the appropriate manipulation primitive net is describe in [17] in more detail.

4 Conclusion and Future Work

In this work, we gave a conceptual overview of our robot system. We outlined the different approaches from a symbolic user interface towards a subsymbolical robot control interface. The transformation between the symbolical and subsymbolical representation is done utilizing physical effects, respectively physical properties.

Our future work mainly focuses on extending the supported vocabulary, which includes on the one hand more complex executable actions, and on the other hand a more flexible parameterization of the actions. Furthermore, we will extend the toolbox of components, which are available for the extraction of subsymbolic information from the environment.

References

1. J. De Schutter, T. De Laet, J. Rutgeerts, W. Decré, R. Smits, E. Aertbeliën, K. Claes, and H. Bruyninckx, "Constraint-based task specification and estimation for sensor-based robot systems in the presence of geometric uncertainty," *The International Journal of Robotics Research*, vol. 26, no. 5, pp. 433–455, 2007.
2. B. Finkemeyer, T. Kroger, and F. M. Wahl, "The adaptive selection matrix a key component for sensor-based control of robotic manipulators," in *IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2010, pp. 3855–3862.
3. S. Harnad, "The symbol grounding problem," *Physica D: Nonlinear Phenomena*, vol. 42, no. 1, pp. 335–346, 1990.
4. S. Coradeschi, A. Loutfi, and B. Wrede, "A short review of symbol grounding in robotic and intelligent systems," *KI-Künstliche Intelligenz*, vol. 27, no. 2, pp. 129–136, 2013.
5. P. Vogt, "The physical symbol grounding problem," *Cognitive Systems Research*, vol. 3, no. 3, pp. 429–457, 2002.
6. A. Cangelosi, "Grounding language in action and perception: From cognitive agents to humanoid robots," *Physics of life reviews*, vol. 7, no. 2, pp. 139–151, 2010.
7. S. Lauria, G. Bugmann, T. Kyriacou, and E. Klein, "Mobile robot programming using natural language," *Robotics and Autonomous Systems*, vol. 38, no. 3, pp. 171–181, 2002.
8. C. Kemke, *" From Saying to Doing"-Natural Language Interaction with Artificial Agents and Robots*. INTECH Open Access Publisher, 2007.
9. C. Matuszek, E. Herbst, L. Zettlemoyer, and D. Fox, "Learning to parse natural language commands to a robot control system," in *Experimental Robotics*. Springer, 2013, pp. 403–415.
10. T. Kollar, S. Tellex, D. Roy, and N. Roy, "Grounding verbs of motion in natural language commands to robots," in *Experimental Robotics*. Springer, 2014, pp. 31–47.
11. T. Laengle, T. C. Lueth, E. Stopp, G. Herzog, and G. Kamstrup, "Kantra-a natural language interface for intelligent robots," in *Intelligent Autonomous Systems (IAS 4)*, 1995, pp. 357–364.
12. A. Knoll, B. Hildenbrandt, and J. Zhang, "Instructing cooperating assembly robots through situated dialogues in natural language," in *IEEE International Conference on Robotics and Automation*, vol. 1. IEEE, 1997, pp. 888–894.
13. N. Pires, "Robot-by-voice: experiments on commanding an industrial robot using the human voice," *Industrial Robot: An International Journal*, vol. 32, no. 6, pp. 505–511, 2005.
14. M. Tenorth, D. Nyga, and M. Beetz, "Understanding and executing instructions for everyday manipulation tasks from the world wide web," in *IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2010, pp. 1486–1491.
15. M. Stenmark and P. Nugues, "Natural language programming of industrial robots," in *2013 44th International Symposium on Robotics (ISR)*. IEEE, 2013, pp. 1–5.
16. D. K. Misra, J. Sung, K. Lee, and A. Saxena, "Tell me dave: Context-sensitive grounding of natural language to manipulation instructions," *The International Journal of Robotics Research*, 2015.
17. M. Spangenberg and D. Henrich, "Grounding of actions based on verbalized physical effects and manipulation primitives," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2015, pp. 844–851.

18. R. Peter Bonasso, R. James Firby, E. Gat, D. Kortenkamp, D. P. Miller, and M. G. Slack, "Experiences with an architecture for intelligent, reactive agents," *Journal of Experimental & Theoretical Artificial Intelligence*, vol. 9, no. 2-3, pp. 237–256, 1997.
19. M. Spangenberg and D. Henrich, "Towards a domain specific language for sensor-based actions," *Applied Mechanics & Materials*, vol. 840, pp. 42–49, 2016.
20. International Organization of Standardization, *ISO Standards Handbook: Quantifies and Units*, 1993.
21. J. Awrejcewicz, *Classical Mechanics: Kinematics and Statics*, ser. Advances in Mechanics and Mathematics. Springer, 2012.
22. M. Spangenberg and D. Henrich, "Towards an intuitive interface for instructing robots handling tasks based on verbalized physical effects," in *RO-MAN: The 23rd IEEE International Symposium on Robot and Human Interactive Communication*. IEEE, 2014, pp. 79–84.
23. ——, "Symbol grounding for symbolic robot commands based on physical properties," in *IEEE International Conference on Information and Automation (accepted, to appear)*. IEEE, 2016.
24. U. Thomas and F. M. Wahl, "Planning sensor feedback for assembly skills by using sensor state space graphs," in *Intelligent Robotics and Applications*. Springer, 2012, pp. 696–707.