Symbol Grounding for symbolic robot commands based on physical properties

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 research is the development of robot systems, which have approximately the same cognitive, communicational, and handling abilities like humans. For instance, a human-like communication requires an intuitive and symbolic user interface. Since actual robot control systems typically consist of subsymbolic interfaces, the robot must be able to extract subsymbolic information from given symbolic instructions. This requires the grounding of information and the utilizing of appropriate sensors and components. In this work, we describe the grounding of symbols based on physical properties, since we describe executable actions in form of physical effects in our system [1].**

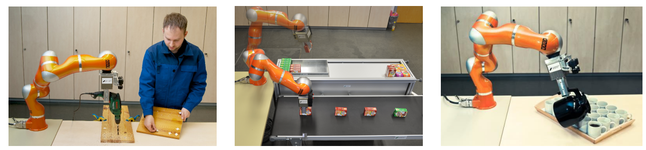

Fig. 1. Typical applications in SMEs or private households which require the execution of sensor based motions. From left to right: Drilling, Paletting, Pouring.

I. Introduction

One long term goal of current robotic research 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 [2] or manipulation primitives [3]. The definition of sensor based actions require expert knowledge in the domain of robotics, since the programmer 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.

For instance, a robot system shall execute a symbolic instruction given according to a domain specific language [4] `shove("the red cube", "towards the green box")`. First, the robot system must be able to analyze the structure of the instruction. In this case, the robot system must be able to detect the instructed action *shove*, and the set of parameters *the red cube* and *towards the green box*. Since the instruction only specifies the goal of the task and not how the execute the task, this information must be grounded in form of a flexible action representation [1].

In addition, the robot needs geometric and dynamic information about the objects, which may be extracted using appropriate sensors and components. In this case, the robot needs the position of the object cube and box, which can be extracted using a camera-based object recognition. Furthermore, the robot needs information about the mass of the object cube and the friction coefficient between the cube and the surface. This is required for the calculation of the applied force for shoving the cube towards the box. This subsymbolic information can be extracted utilizing available components and sensors, for instance an object recognition using an optical sensor, or a mass recognition using a force sensor.

In this work, we analyze and structure the subsymbolic information contained in specific symbols. More specific, we analyze the physical properties manipulated by a symbol, since we ground executable actions in form of verbalized physical effects in our previous work [1]. We describe the relations between symbols, subsymbolic physical parameters which are manipulated by specific symbols, and components for extracting subsymbolic information from the environment. This information is stored in form of a physical dictionary to the robot system.

The remainder of this 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 III, 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. In Section IV, we show the extraction of subsymbolic information from given instructions and highlight the influence of the used symbols on the execution of an instruction. At last, we describe our future work in Section V.

II. Related Work

The problem of assigning semantics to symbolic tokens like words is known as the symbol grounding problem and was described by Harnad [5] 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 [6]. The grounding of symbols can be organized into two subtopics, physical symbol grounding [7], and social symbol grounding [8]. 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.

We highlight systems, which can be operated utilizing symbolic commands. In general, such robot systems are either used within navigational or handling tasks. Lauria et al. [9] describe the navigation of a miniature robot using a predefined functional vocabulary, which includes known actions and parameters. For the navigation of (virtual) robots, Kemke [10] proposes the usage of an ontology based knowledge representation, Matuszek et al. [11] propose the usage of a semantically-labeled map, and Kollar et al. [12] propose the usage of spatial description clauses.

In the area of handling tasks, Laengle et al. [13] use commands with a predefined syntax and known parameters and execute them via predefined plans. Knoll et al. [14] describe the assembly of wooden toys using a skill library with flexible object positions. Pires [15] describes the control of a industrial robot using predefined commands. Tenorth et al. [16] utilize an ontology based knowledge representation and learned action models for the execution of natural language instructions from the world wide web. Stenmark and Nugues [17] describe the programming of industrial robot systems using semantically annotated state machines. Misra et al. [18] define manipulation actions based on a set of low-level instructions, which are parameterized by an symbolic object identifier.

We conclude that diverse robot systems can execute symbolic commands and extract various subsymbolic information. We categorize these systems according to the extractable subsymbolic information in three categories.

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 a action representation utilizing verbalized physical effects and manipulation primitive nets [1], which is parameterized by geometric, kinematic, and dynamic parameters, therefore we need to specify how to extract these quantities from a symbolic representation.

Therefore, the contribution of this paper consists of a component we call physical dictionary, which grounds symbols to a robot system according to manipulated subsymbolic physical properties. Our dictionary consists of two parts. The first part of the dictionary grounds relations between symbols and the manipulated physical information. In contrast to other approaches, our dictionary considers additional physical properties like kinematic and dynamic properties. The second part of the dictionary grounds information about external components and sensors, which can be utilized for the automatic extraction of the needed subsymbolic information from the environment.

III. Approach

The main idea of our 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. Therefore, we introduced the concept of *principal physical effects* $\mathcal{PPE}$s in [20]. We extend the concept of $\mathcal{PPE}$s to *verbalized physical effects* $\mathcal{VPE}$s, a flexible action representation based on $\mathcal{PPE}$s and manipulation primitive nets $\mathcal{MPN}$s [1].

In the following subsections, we first give an overview of our system architecture. Next, we outline the concept of $\mathcal{VPE}$s which is used as action representation and which shall be parameterized using symbolic parameters. Next, we describe the relations between symbols and the contained subsymbolic physical information, which are stored in form of a physical dictionary in our system. At last, we describe the parameter extraction based on the physical dictionary using appropriate sensors and software components.

A. System Overview

An overview of our system architecture is shown in Figure 2. Our system is build according to the 3T architecture [21], a common architecture for systems which have to transform between different types of representations. The system

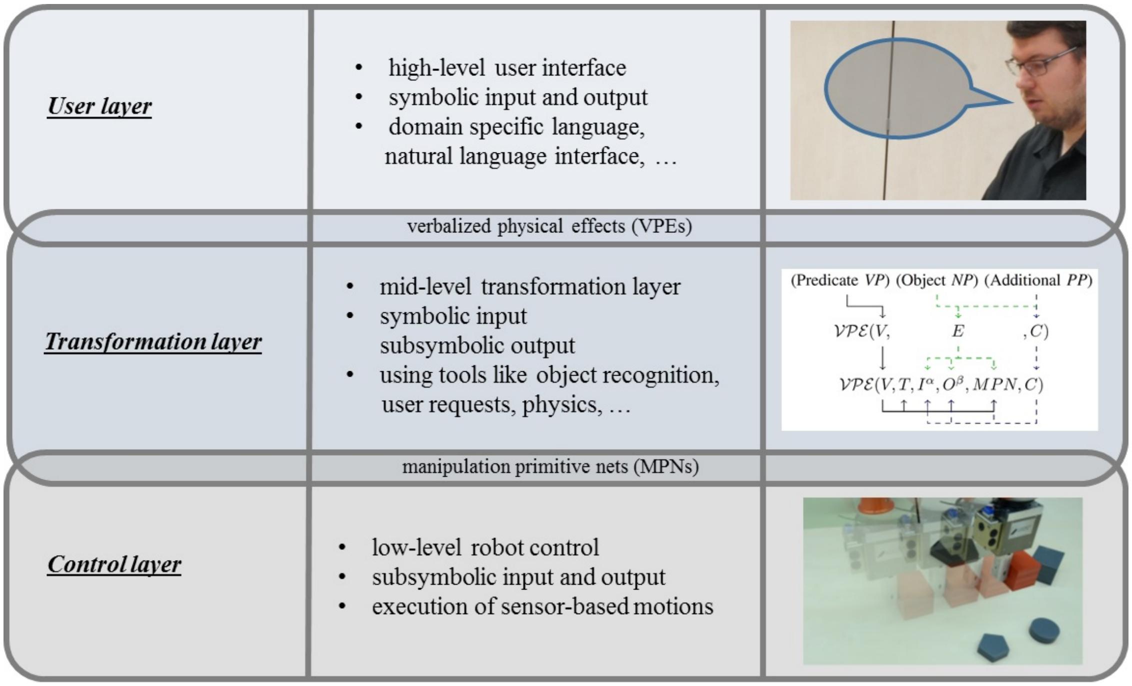

Fig. 2. Overview of the system architecture.

consists of a user layer, which provides intuitive high-level user interfaces. At the current stage, our system provides a domain specific language interface [4] and a simple natural language interface. This high-level user input is mapped to a specific $\mathcal{VPE}$, which is transformed into a representation of a manipulation primitive net using a *toolbox* consisting of the physical dictionary, sensors, and software components. The parameterized manipulation primitive net is then transfered to the control layer, which executes the sensor-based motions defined by the manipulation primitive net.

B. Verbalized Physical Effects

The concept of verbalized physical effects $\mathcal{VPE}$s is used 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 [22]. 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 *geometric*, *kinematic*, and *dynamic* units [23]. 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 [20]. 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 [1].

C. Symbol Analysis and Grounding

The first objective of the physical dictionary is the representation of the manipulated subsymbolic information by a symbol. Since we are interested in an intuitive user interface, we want to use natural language symbols as parameters. Therefore, the first step is to categorize natural language symbols, extract which symbols are relevant for a robot system according to the application of object manipulation tasks, and analyze which subsymbolic information is manipulated by the symbols.

We categorize symbols according to the *syntactical function* and the *part-of-speech*. For the categorization in syntactical functions, we use the tagset described in [24], which is also used for the well-known PENN treebank [25]. The main syntactical functions are a *sentence* (S), an *adjective phrase* $(ADJP)$, an *adverb phrase* $(ADVP)$, a *noun phrase* (NP), a *prepositional phrase* (PP), and a *verb phrase* (VP). The information of the syntactical function is used to determine coherence between different symbols. For instance, given an instruction like "*Shove the red cube towards the green box!*", it is allowed to extract that the adjective *red* refers to the noun *cube*, and the adjective *green* refers to the noun *box*.

Besides the syntactical function, we utilize the part-of-speech of the symbols. For the categorization we use a universal tagset described in [26]. The set of tags consists of *nouns* $(NOUN)$, *verbs* $(VERB)$, *adjectives* (ADJ), *adverbs* (ADV), *pronouns* $(PRON)$, *determiners* and *articles* (DET), *prepositions* and *postpositions* (ADP), *numerals* (NUM), and *conjunctions* $(CONJ)$. In the next paragraphs, we describe the principal function of each word class based on [27]. In addition, we describe the relevant information which must be utilized and grounded to a robot system.

The group of nouns $NOUN$ is typically organized in *common* and *proper* nouns, while common nouns are further categorized in *concrete* and *abstract* nouns. Concrete nouns describe real physical objects (cup, table) of the environment, while abstract nouns describe properties (kindness, fortitude), processes (payout, aging), or states (war, peace). Furthermore, a noun contains information about the affected amount of objects, which is either a single object (cube, box) or multiple objects (cubes, boxes). In the context of robot systems, a noun describes an object of the real world, which has physical properties like a position, mass, or color. In addition, the noun determines if an action shall be executed one or multiple times.

The group of verbs $VERB$ contain information about actions, which shall be executed by the robot system. Since the execution of a verbal expression results typically in more than one elemental robot motion, the robot needs information about subtasks of the specified instruction. Therefore, a flexible representation of executable actions is required. This was analyzed in a previous work [1], where we introduced the concept of a flexible action representation based on verbalized physical effects.

The group of adjectives ADJ describe properties of objects. Each physical quantity can be modified by a specific adjective, for instance geometric (big, right), kinematic (slow, fast), dynamic (light, heavy), or visual (green, blue) properties. Similar to singular and plural of nouns, adjectives can provide information about the frequency of a executable action. In contrast to nouns, which can only specify exactly one or a undefined multiple amount, adjectives can additionally describe boundaries (minimum, maximum). Furthermore, adjectives typically describe properties in form of fuzzy sets. For instance, the adjective *green* describes a visual property of an object and subsumes a range of potential real color values. Since some adjectives are comparable (great, greater, greatest), adjectives can implicit demand a comparison between objects.

The group of adverbs ADV describe properties of actions. Similar to adjectives, adverbs can provide information about the frequency of a task. In addition to nouns and adjectives, adverbs can specify a property exactly (e.g. twice) without comparing properties.

The group of pronouns $PRON$ typically replace words from the group of $NOUN$. The extraction of the substituted noun is one of the classical types of information extraction tasks in natural language processing, the co-reference resolution. Since a pronoun can be replaced by the originally noun, pronouns and nouns describe the same subsymbolic information.

The group of articles and determiners DET describe information about the certainty of an object. This group is categorized in *definite* determiners (the, this) and *indefinite* determiners (a, any). Similar to nouns, DET can describe either a specific object or multiple objects.

The group of prepositions and postpositions ADP describe relations between objects. This relations are typically *spatial* (on, in) or *temporal* (before, after). In context of a robot system, we describe a relation between object by a set of boolean constraints on object properties, which can be checked by the robot system.

The group of numerals NUM describe symbols for exact subsymbolic information. The subsymbolic information contained in a numeral can be extracted directly by a robot system. Since a numeral has no unit, the combination between numeral and unit specifies the manipulated physical quantity.

The group of conjunctions $CONJ$ describe relations between sentences, phrases, or clauses. Similar to prepositions and postpositions, conjunctions are defined by a set of boolean constraints.

Based on the analysis, a schematic representation of the physical dictionary is illustrated in Figure 3 (upper dashed frame).

D. Subsymbolic Parameter Extraction

The second objective of the physical dictionary is to provide information about components and sensors, which can be utilized by the robot system in order to extract physical parameters from the environment. Therefore, we extend the physical dictionary by a component and sensor submodule. The component submodule contains a list of components, which are available in the overall system, and maps extractable quantities to specific components. The sensor submodule describes the required/utilized sensors of a component (see Figure 3, lower dashed frame). This allows the robot system to identify the quantities, which can be extracted utilizing available components and sensors. If no component or sensor for the extraction of a quantity is available, the robot system can identify a missing component and can retrieve the information using user requests or default values. Furthermore, the development of extraction components is decoupled from the overall functionality and new components or sensors can easily be integrated in the physical dictionary. This allows us to integrate existing approaches in the overall system. Since there are typically more components and sensors available for the extraction of a physical quantity (for instance a 2D and a 3D object recognition utilizing the same sensor like a Microsoft

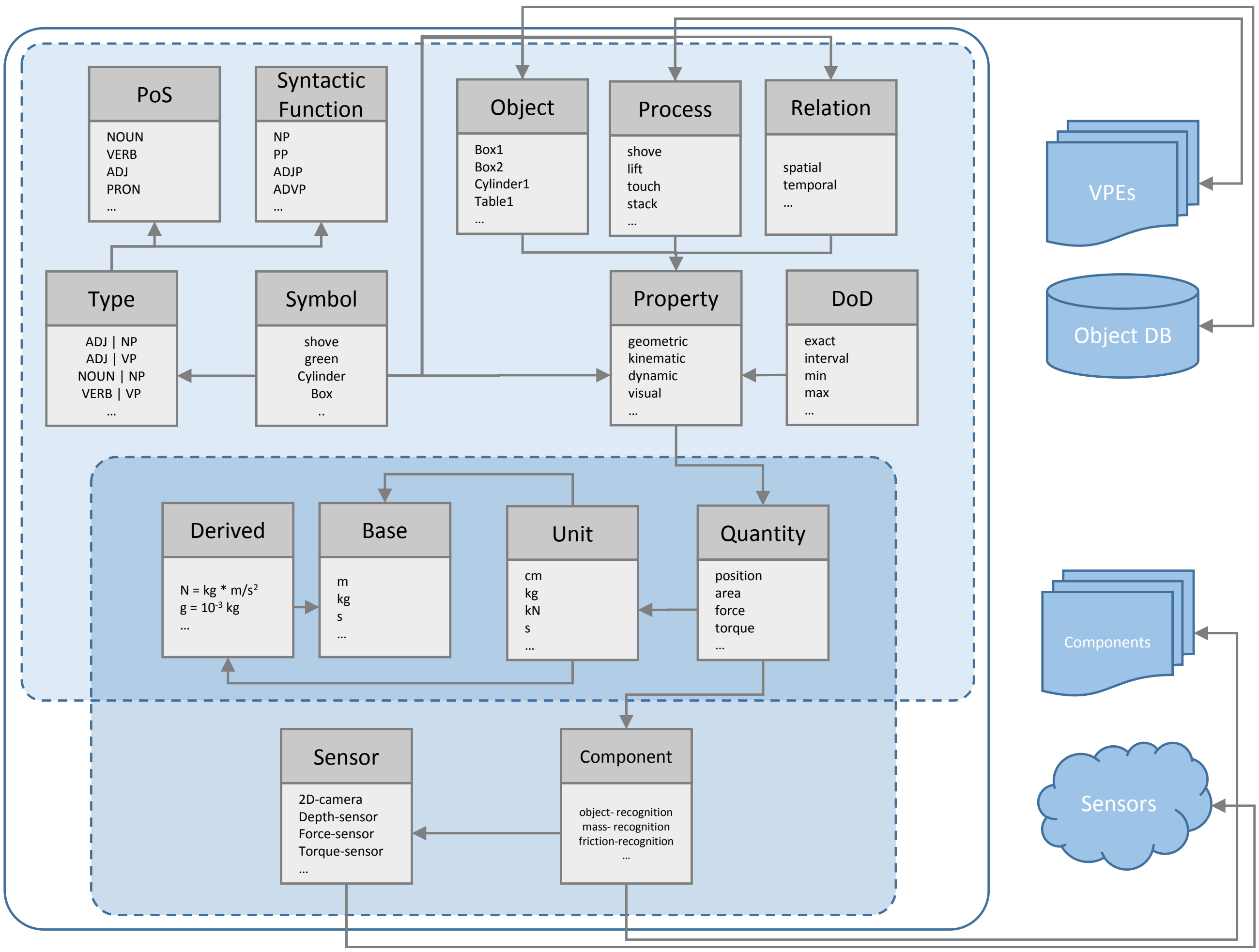

Fig. 3. Schematic representation of the grounded information in the physical dictionary (white frame). The physical dictionary stores the subsymbolic information of a specific symbol (upper dashed frame) and the components and sensors needed for extraction (lower dashed frame). Since the physical dictionary is one component of the overall system, it utilizes other components like the action representation based on verbalized physical effects, an object database, components like an object recognition, and sensors like force-torque sensors for extracting subsymbolic parameters from the real world.

Kinect), it is conceivable to define a criteria function, which describes the most suitable component for a specific task, respectively a specific context. In the actual prototype, we use a boolean flag which sets the active extraction component of a physical quantity.

IV. Evaluation

In the last section, we described our approach for the definition of a physical dictionary. Now, we illustrate a prototype for the execution of symbolic instructions and the extraction of subsymbolic information on a robot system consisting of a Kuka LWR 4, a two-finger-parallel gripper Schunk PG70 and a Microsoft Kinect. The robot system is equipped with internal force-/torque sensors, a depth sensor, and a color sensor. In addition, the system can utilize different components for the extraction of physical information like positions, masses, forces, friction coefficients, or energies. We integrate the physical dictionary in form of a relational database and ground diverse symbols for each class of words. Since we cannot present all executable instructions, we select a subset of executable instructions and focus on two behaviors for the evaluation. On the one hand, the evaluation shall demonstrate the influence of specific symbols to the execution of a specified action. On the other hand, the evaluation shall demonstrate the extraction of subsymbolic information from a given symbolic instruction utilizing the physical dictionary. The executable instructions are defined in our domain specific language [4].

The execution of selected symbolic instructions is shown in Figure 4. We execute three verbalized physical effects and variate the symbolic and subsymbolic parameters. In the next paragraphs, we describe the results in detail. For each of the three scenarios, we highlight the differences due to the variation of the used symbolic parameters, and the subsymbolic information extracted by the robot system.

The first row of Figure 4 (a)-(d) shows the execution of the verbalized physical effect mapped to the verb *to touch*. The action takes one parameter in form of a noun phrase, which specifies the object that shall be touched. We use

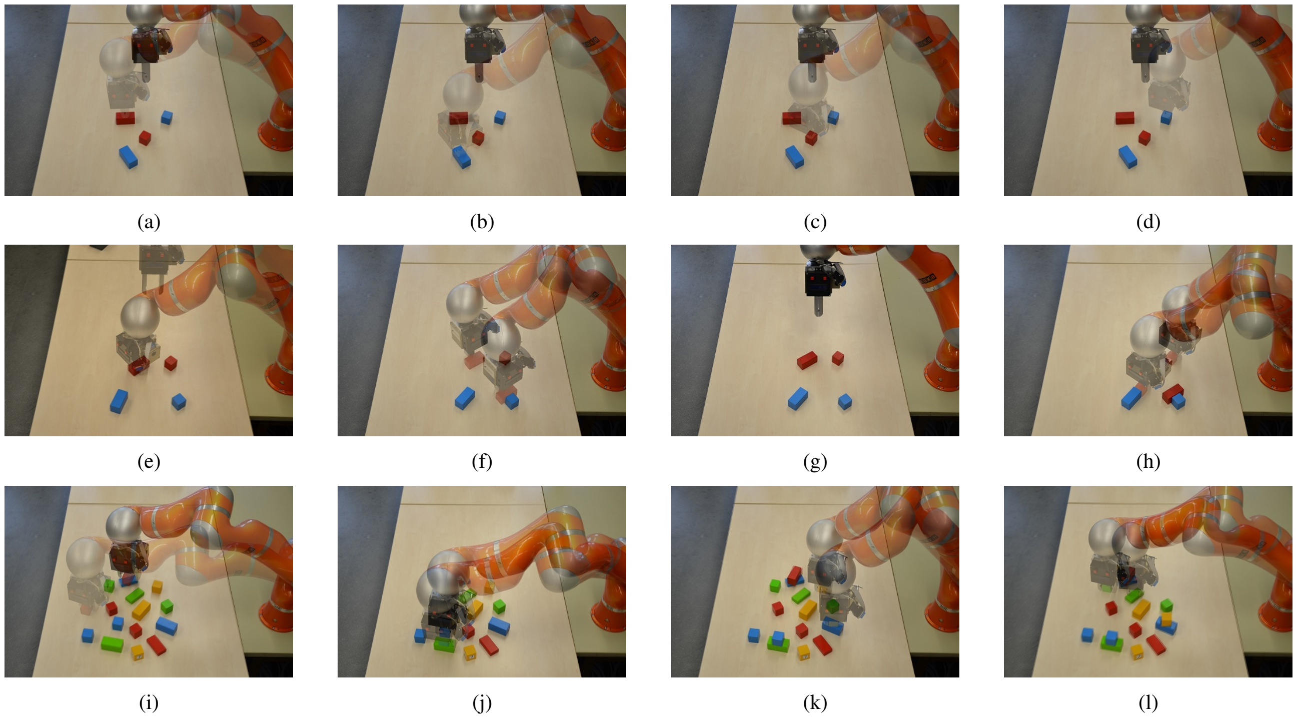

Fig. 4. Schematic demonstration of different symbolic instructions executed by the robot system. Each row describes the execution of a specific verbalized physical effect. The columns illustrate the execution due to the varied symbolic parameters.

this effect to demonstrate the influence of a noun and an adjective on a symbolic instruction. We use the nouns *box* and *cube*. These are known objects which are grounded in an object database. In addition, we use the adjectives *red* and *blue*, which restrict a property of the noun within the noun phrase. The adjectives describe the visual property *color* based on a *fuzzy set*. In the actual system, we describe a fuzzy set in form of a specified interval. The extracted subsymbolic information are the geometric property *position*, and the visual property *color*. Figure 4 (a) shows the execution of the instruction `touch("the red box")`, Figure 4 (b) shows the execution of the instruction `touch("the blue box")`, Figure 4 (c) shows the execution of the instruction `touch("the red cube")`, and Figure 4 (d) shows the execution of the instruction `touch("the blue cube")`.

The second row of Figure 4 (e)-(h) shows the execution of the verbalized physical effect mapped to the verb *to shove*. The principal physical effect mapped to the verbal expression is *transform a force into a displacement*, which has two parameter. The first parameter describes the object, on which a force has to be applied in order to achieve a displacement of the object. In addition, this parameter estimates the value of the force. The second parameter is used to calculate the direction of the force. We use this effect to demonstrate the extraction of diverse subsymbolic parameter by the robot system. The extracted subsymbolic information are the geometric property *position* of the objects, the visual property *color* of the objects, the dynamic property *mass* of the object which shall be shoved, and the geometrical property *distance* between the two objects. The mass and distance parameter is utilized for the calculation of the value and direction of the force, which has to be applied on the manipulated object. For the extraction of the dynamic property *mass*, we utilize either the force sensors (see Figure 4 (e)) or a geometric based component using an approximation of the density (see Figure 4 (g)). This shows the flexible usage of components for the extraction of the same physical property. Figure 4 (e) shows the extraction of the dynamic parameter *mass* using the force sensor component, Figure 4 (f) shows the execution of the instruction `shove("the red box", "the blue cube")`, Figure 4 (g) shows the extraction of the dynamic parameter *mass* using a geometric based component, and Figure 4 (h) shows the execution of the instruction `shove("the red cube", "the blue box")`.

The third row of Figure 4 (i)-(l) shows the execution of the verbalized physical effect mapped to the verb *to place*. The action takes two parameter in form of a noun phrase, which specifies the object that shall be placed. The second parameter describes the destination of the object, typically in form a prepositional phrase. We use this effect to describe the influence of determiners on a symbolic instruction. We use the determiners *the* and *a*. The determiners specify either only one object, or allow a selection of objects. The determiners describe the property *amount* either *exact*, or *free*. In addition, we describe the preposition *on* in this scenario. The preposition on is specified by boolean constraints according to the geometric property *area* and *height* of the objects. The extracted subsymbolic information are the geometric property *position* and *height*, and the visual property *color*. Figure 4 (i) shows the execution of the instruction `place("the red box", "on the blue cuboid")`, Figure 4 (j) shows the execution of the instruction `place("a blue cube", "on a box")`, Figure 4 (k) shows the execution of the instruction `place("a cube", "on the blue box")`, and Figure 4 (l) shows the execution of the instruction `place("a cube", "on a box")`.

V. Conclusion and Future Work

In this work, we introduced a system which utilizes a physical dictionary for the extraction of subsymbolic information from symbolic commands. The physical dictionary provides information about the physical properties, which are manipulated by specific symbols. Furthermore, it provides information about components and sensors, which can be utilized to extract subsymbolic information from the environment. This information is required for the parameterization of verbalized physical effects, a flexible action representation based on the description of executable actions in form of principal physical effects [1].

Based on the analysis of word classes and syntactic functions, we showed that symbols contain and specify physical subsymbolic information, which can be systematically described and utilized by a robot system. Furthermore, we showed that symbols can specify subsymbolic information in different degrees of determination like exact, fuzzy, or by defining intervals with or without specified endpoints.

We evaluated the physical dictionary using a real robot system, which uses an action representation based on verbalized physical effects for the execution of sensor based robot motions. The physical dictionary act as connector between the symbolic representation suitable for the users of the robot system, the components used for information extraction, and the subsymbolic representation, needed by the robot control. Since the actual extraction of subsymbolic information is decoupled, existing approaches for extracting subsymbolic information can be used in a new context.

In future work, we will extend the supported vocabulary. Furthermore, we will address the definition of a criteria function, which describes the most suitable extraction method for a specific physical quantity. In addition, we will extend the toolbox of components, which are available for the extraction of subsymbolic information from the environment.

References

[1] M. Spangenberg and D. Henrich, "Grounding of actions based on verbalized physical effects and manipulation primitives," in *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*. IEEE, 2015, pp. 844–851.

[2] 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.

[3] 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.

[4] M. Spangenberg and D. Henrich, "Towards a domain specific language for sensor-based actions," *Applied Mechanics and Materials Vol. 840 - Robotics and Automated Production Lines*, pp. 42–49, 2016.

[5] S. Harnad, "The symbol grounding problem," *Physica D: Nonlinear Phenomena*, vol. 42, no. 1, pp. 335–346, 1990.

[6] 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.

[7] P. Vogt, "The physical symbol grounding problem," *Cognitive Systems Research*, vol. 3, no. 3, pp. 429–457, 2002.

[8] 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.

[9] 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.

[10] C. Kemke, *" From Saying to Doing"-Natural Language Interaction with Artificial Agents and Robots*. INTECH Open Access Publisher, 2007.

[11] 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.

[12] 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.

[13] 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.

[14] 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.

[15] 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.

[16] 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.

[17] M. Stenmark and P. Nugues, "Natural language programming of industrial robots," in *2013 44th International Symposium on Robotics (ISR)*. IEEE, 2013, pp. 1–5.

[18] 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*, p. 0278364915602060, 2015.

[19] J. Piskorski and R. Yangarber, "Information extraction: Past, present and future," in *Multi-source, multilingual information extraction and summarization*. Springer, 2013, pp. 23–49.

[20] 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.

[21] 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.

[22] International Organization of Standardization, *ISO Standards Handbook: Quantifies and Units*, 1993.

[23] J. Awrejcewicz, *Classical Mechanics: Kinematics and Statics*, ser. Advances in Mechanics and Mathematics. Springer, 2012.

[24] A. Taylor, M. Marcus, and B. Santorini, "The penn treebank: an overview," in *Treebanks*. Springer, 2003, pp. 5–22.

[25] M. P. Marcus, M. A. Marcinkiewicz, and B. Santorini, "Building a large annotated corpus of english: The penn treebank," *Computational linguistics*, vol. 19, no. 2, pp. 313–330, 1993.

[26] S. Petrov, D. Das, and R. McDonald, "A universal part-of-speech tagset," *arXiv preprint arXiv:1104.2086*, 2011.

[27] K. Kessel and S. Reimann, *Basiswissen Deutsche Gegenwartssprache, 4. durchges. Aufl.*, ser. UTB / UTB. UTB GmbH, 2012.