Honda Research Institute EU
Honda Research Institute Europe GmbH
https://www.honda-ri.de/

Improving HRI through robot architecture transparency

Lukas Hindemith, Christiane Wiebel, Heiko Wersing, Britta Wrede, Anna-Lisa Vollmer

2025

Preprint:

This is a post-peer-review, pre-copyedit version of an article published in International Journal of Social Robotics . The final authenticated version is available online at: https://doi.org/10.1007/s12369-025-01237-2

International Journal of Social Robotics (2025) 17:2981–3001
https://doi.org/10.1007/s12369-025-01237-2

Improving HRI Through Robot Architecture Transparency

Lukas Hindemith[1,2] · Christiane B. Wiebel-Herboth[1] · Heiko Wersing[1] · Britta Wrede[2] · Anna-Lisa Vollmer[2]

Accepted: 19 February 2025 / Published online: 4 April 2025
© The Author(s) 2025

Abstract
One ongoing challenge in human-robot interaction design is minimizing user misunderstandings and confusion. While engineers constantly improve the reliability of robots, the user's mental model about robots and their limitations have to be addressed as well. In this work, we investigate ways to improve the human understanding about robots. For this, we propose *FAMILIAR – FunctionAl user Mental model by Increased LegIbility ARchitecture*, a transparent robot architecture with regard to the robot behavior and decision-making process. We conducted an exploratory online simulation user study (N=81) to evaluate two complementary approaches to convey and increase the knowledge about this architecture to non-expert users: a dynamic visualization of the system's processes as well as an interface for defining the sequence of user and robot actions for teaching the robot, the interaction protocol. The experimental scenario consisted of teaching a robot about a simulated indoor environment. The results of this study reveal that the definition of an interaction protocol improves knowledge about the architecture measured via a questionnaire on knowledge of the different conceptual elements of the system (Sensors, Interaction Protocol, Behaviors, Preconditions, Actions, and how these interact: the Process). Furthermore, we show that with increased knowledge about the control architecture of the robot, users were significantly better in reaching the interaction goal. Moreover, we interestingly found that anthropomorphism may actually reduce interaction success. Our results support the crucial role of considering user mental models in robot architecture design.

Keywords Explainable robots · Human-robot interaction · Mental models · Visualization · Control architecture · User study

1 Introduction

Advancements in robot capabilities have facilitated the transition of application fields from mainly static environments to more dynamic ones with an increased involvement of humans [1, 2]. Thereby, scenarios emerged where the robot cannot pre-plan all actions in advance to accomplish a certain goal; instead, only close interactions with a human partner allow the robot to determine the subsequent action [1, 2]. Moreover, not only experts are interacting with a robot, but also *lay users*. The term *lay user* in the following denotes a human user who neither has a computer science or similar background nor is familiar in interaction with a robotic system. However, we assume that users live in a society where almost everyone is exposed to technology to some degree, such as computers or smartphones.

Although to some degree robots are capable of interacting with lay users, issues such as attention to false targets by robot or human, bad timing in utterances or perception failures can occur regularly [3, 4]. These issues can generally be attributed to both the robots themselves and the users [5]. As robots become advanced enough to operate in a variety of complex real-world settings, the potential for errors increases if technology readiness has not reached maturity: to perceive the world, various hardware sensors need to work reliably, several modules need to communicate with each other and data needs to be transferred [6]. The behavior of a robot also

✉ Anna-Lisa Vollmer
anna-lisa.vollmer@uni-bielefeld.de

Lukas Hindemith
lukas.hindemith@web.de

Christiane B. Wiebel-Herboth
christiane.wiebel@honda-ri.de

Heiko Wersing
Heiko.Wersing@honda-ri.de

Britta Wrede
bwrede@techfak.uni-bielefeld.de

[1] Honda Research Institute Europe GmbH, Carl-Legien-Straße30, 63073 Offenbach, Germany

[2] Medical School OWL and Center for Cognitive Interaction Technology (CITEC), Bielefeld University, Universitätsstraße 25, 33615 Bielefeld, Germany

depends on successful perceptions of the environment, working actuators, and a correct design by engineers [7]. Besides errors in the robot system, users have a tremendous influence on the performance of a robot. Especially in close human-robot interactions and cooperation, the successful behavior of a robot highly depends on the correct input by the user [5, 8]. A more detailed failure taxonomy can be found in [5].

Lay users are often unaware of the functionality and limitations of a robot. This can be ascribed to the *mental model* a lay user has about the robot [9]. According to [10], a mental model is defined as a cognitive framework of internal representations humans build about things they interact with. When a person interacts with an artifact for the first time, an initial representation is built and continuously updated while interacting with it. This is influenced by prior experiences with other artifacts and expectations towards them. In the case of robots, people relate to experiences with other people [11]. Additionally, expectations about robots, formed by movies and media, may lead a naïve user's mental model further away from reality [12]. Consequently, a mental model forms in the user's mind, which could lead them to provide input that doesn't align with how the robot processes input (e.g., using a modality the robot does not use or giving commands in the wrong order), and which could cause them to misunderstand the robot's behavior [13] often leading to interaction failure [14]. This calls for mechanisms that elicit the formation of - or the correction of these models towards - *functional mental models* in the user that are either congruent with the functionality and limitations of a robot or may be incongruent but nevertheless result in correct user input.

Our proposed approach for supporting the formation of a functional mental model in the user is based on a) recent work on *technical transparency mechanisms* (see Sect. 2.1) and b) the teacher's role in the co-construction of interaction patterns in adult-child teaching/learning interactions in contrast to fixed interaction protocols (i.e., the structure of the human-robot teaching interaction, see Sect. 2.2 for an example) in current human-robot teaching/learning interaction (see Sect. 2.2). We, thus, aim to convey the robot's internal architecture to the user via visualizations and user-defined interaction protocols.

1.1 Contribution

This work contributes to research in the field of explainable robotics as it investigates how to achieve better understanding of the robot's internal functioning in human users and whether a better understanding improves interaction with the robot. For this research, in a first effort, we employed a simple robot control architecture and two types of user tools, a) visualizations of the inner processes of the robot and b) a user interface for defining the interaction protocol, to increase understanding of the robot in the user. We evaluated these in an interactive online user study in a simulation environment. Additionally, we looked into the relationship between understanding of the robot's architecture and the success of the interaction, exploring the main architecture concepts responsible for reaching the goal of the interaction successfully.

This work demonstrates that user involvement in protocol definition can significantly enhance their mental model of the robot's architecture. We concentrate on communicating the specific concepts of the complex control architecture of the robot (sensors, interaction protocol, predecessor, behavior, precondition, action, and how they interact, the process, see Sect. 5.6.1) using our specialized transparency tools (a dynamic visualization of the system's processes as well as an interaction protocol interface, see Sects. 4.2 and 4.3), aiming to enhance user comprehension of the robot's inner workings. Our approach is distinctive in that it empowers users by involving them in defining the interaction protocol prior to interaction and through visualizing the robot's current perceptions and active behaviors during.

2 Related Work

In the context of artificial intelligence and robotics, the terms "transparency", "explainability", "understandability", and "legibility" might appear synonymous ([15], cf.). However, each has a distinct meaning within the literature: *Transparency* generally refers to the clarity and openness with which the internal mechanisms of a system are disclosed to users [16]. This can include the processes through which decisions are made (such as path planning in robots) or the components involved in these processes (such as sensor values). Transparency is often approached through visualizations or other forms of output that aim to make the workings of the robot or system clear to those interacting with it [16]. The goal is to allow users to understand how decisions are being made or actions are being performed. In the field of human autonomy teaming, transparency has been found to affect operator performance, situation awareness, perceived usability, and trust ([17], cf.). Through the use of mental models, fluency and adaptability of behavior, communication, and explainability have been elicited in human-agent teams ([18], cf.). Findings are, however, inconsistent with respect to the amount of transparency information conveyed as too much information can be overwhelming for humans ([17], cf.). *Explainability* focuses more on the logic or reasoning behind decisions [15]. It involves the ability to describe in understandable terms why a system is behaving in a certain way or why it has made a particular decision. An explainable AI system should be able to justify its actions or outputs in a way that is comprehensible to humans, potentially using natural language or through detailed feedback mechanisms [19].

Understandability or comprehensibility refers to how easily the information provided can be followed by users. Understandable systems present information and explanations in a way that is accessible and clear to its intended audience, which often requires the information to be tailored to the user's context and level of expertise [20]. In robotics, *legibility* specifically refers to how easily a human observer can predict a robot's goals based on its actions or behavior [21].

2.1 Improving Mental Models Towards Functionality

Various transparency mechanisms were investigated in the field of human-robot interaction to convey aspects of the robot to users. These techniques aim to promote functional mental models in users by making robots' behavior, decisions, and goals more understandable. Most approaches utilize human-like characteristics and social cues. Verbal explanations have been utilized to clarify the planning of actions, the reasoning process and decisions [22–24] and have been generated autonomously [25–27]. Extensive explanations seem to improve the mental model more effectively than concise ones [28]. Another strand of research focuses on non-verbal communication, such as eye gaze [29–31], where timing is crucial for the comprehension of such social cues [32]. In addition to social cues, the robot's behavior itself can communicate information about intentions and goals by being made more legible, instead of pursuing efficiency [33, 34]. Such behavior has been developed for example for motion planning [35], steering behavior in autonomous driving [36], and robot arm movements [21, 37]. [38] have focused on grasping techniques that convey incapability to grasp an object. Other mechanisms communicate confidence in manipulation or perception [39, 40]. These social transparency mechanisms are geared at being intuitively understandable, they, however, favor anthropomorphism of the robot [41] and, thus, might lead users to overestimate the robot's abilities and limitations [9, 41].

In contrast to social mechanisms, we term mechanisms that focus on technical means for communication *technical transparency mechanisms*. These hold the promise to be able to convey information about the robot without inducing false assumptions [42]. Various work developed visualizations on displays to express the reasoning process of robots [43–45] and the behavior decision process [43]. [45] developed a multimodal interface employing visualizations on the reasoning process of an autonomous system and natural language descriptions. Also simple text displays can be utilized to convey intents [46]. Besides displays, a projector was utilized to show environment perception and path planning on the floor [47]. Similarly, [48] utilized projection of information to inform users about its intentions in a human-robot teamwork scenario. Additionally, head-mounted displays were used for the visualization of robot intentions [49–52] and robot perception for understanding error situations in navigation tasks [53]. [54] have used augmented reality in human-robot interaction to convey information on the perceptual beliefs of the robot as well as on the robot's intentions to enhance the collaboration. [55, 56] developed different light patterns to communicate robot states. However, as some of these technical mechanisms are unknown to naive users, training might be required to understand the transparency mechanisms correctly.

While we also employ visualizations, most of the above techniques concentrate on the communication of general aspects like intentions, goals or uncertainty of the robot, neglecting the underlying concepts of the robot's control architecture. While various studies were conducted to measure the influence of communication strategies on users' attitudes towards robots or team performance measures ([17, 57], cf.), the mental model of how a human user conceptualizes how a robot works seems to have been rarely investigated explicitly. Previous work showed lay users can comprehend the architectural concepts of a robot [42]. Furthermore, the comprehension depends on the familiarity and observability of concepts [42].

2.2 Understanding Through Configurable Interaction Protocols

In human-human interaction and especially in adult-child interaction, recurring interaction patterns can be observed [58, 59] that, unlike interaction protocols in human-robot interaction, are co-constructed by interaction partners [60]. This co-construction involving actions of both partners and mutual shaping of internal constructs allows interaction partners to align and attune to each other based on mutual understanding.

As a first step for achieving a co-construction between a human and a robot we will target the ability of the human to define the interaction protocol based on which the interaction with the robot will be structured. By interaction protocol, we mean the sequence of actions of robot and user which is followed to teach a robot. An example for an interaction protocol in learning locations could be 'the user going to the robot's location, moving in front of the robot, waiting until the robot recognizes them, the robot giving feedback on user recognition, the user telling the robot to follow them, both moving to the middle of the room to be taught and the user uttering a command and the label of the room'.

Our approach is inspired by visual programming interfaces within the realm of End-User Development [61, 62] that allow the user to define the behavior of robots via drag-and-drop of behavioral primitives [63, 64]. To the best of our knowledge, there is no work providing an interface to users with which they can define their own human-robot interaction protocol for robot teaching.

2.3 Transparent Robot Control Architectures

Few other works also target intuitive robot control architectures. In contrast to our work, they mainly focus on being easy to work with for experts (debug and extend [65]) or lay users (intuitive programming [66], learnt intuitive control [67] or interactive task learning [68]) or on producing transparent behavior [69]. In our work, we aim for the control architecture to be transparent with respect to a lay user's understanding of the architecture itself.

For a robot to interact with the world, sensors need to observe the environment. Based on the perception of the robot, the control module defines which actuators will interact with the world to produce a certain behavior. As this robot control module forms the basis for the robot's interaction behavior, the user's perception and comprehension of it have a tremendous influence on the success of the interaction [5, 8]. As previous work showed, the widely used concept of the *state machine* for robot control has been shown to not be intuitively comprehensible for lay users [42]. State machines are composed of states and connections between them. A transition from one state to another depends on the state outcome for a given input [70]. Furthermore, in robotic applications, state machines often allow for concurrency [71]. This highly interlaced structure makes it difficult for lay users to trace the decision-making process, which is especially crucial in case of errors.

Various architectures for robot control use complex control structures to achieve goal-oriented behavior. While these allow for solving challenging tasks, the comprehension by lay users suffers. This is primarily due to the frequently used, hierarchical structure of the planning process [72–75].

In contrast, behavior-based controls, which we will implement here, are more reactive in their traditional form [76]. They define a set of modules (called behaviors), which are composed of expected sensory input as the trigger and behavioral patterns that achieve a certain goal. These behavioral patterns, in turn, produce an output on actuators [77]. Even though the reactivity reduces the applicability for complex tasks, it increases the legibility [78]. This is achieved by the tight coupling of sensors and their impact on executed behaviors. In contrast, state machines mostly run through an extensive process of state transitions to determine the execution of an action.

3 Research Questions

With the goal to improve human-robot interactions through increasing the user's knowledge about the architecture of the robot, we want to tackle the following research questions:

RQ 1 Can we increase the user's architectural knowledge about the robot control system by visualising the inner processes of the robot?

RQ 2 Can we increase the user's *structural* knowledge (i.e., knowledge about the components of the system and how they interact as opposed to *process* knowledge focusing on goal-oriented knowledge on how to interact with the system) about the robot's control system by letting the user define the interaction protocol for the robot themselves?

RQ 3 Does an increased knowledge about the robot's control system help to reduce the number of errors in the human-robot interaction?

4 Robot System

We realized our system with a Gazebo [79] robot simulation of the robot platform Scitos G5 by MetraLabs. The model was realized using the CAD data of the robot.

In the following, we will describe the robot control architecture *FAMILIAR*, which consists of a behavior-based method for controlling primitives (cf. Sect. 4.1). To improve the legibility of the proposed architecture, we developed two complementary ways for the user to engage with the system: a dynamic visualization (DV) that communicates how social primitives are used by the architecture (cf. Sect. 4.2) and a robot interaction protocol interface (IPI) (Sect. 4.3). While the dynamic visualization aims at displaying the inner working (semantic sensors and sensor values such as a recognized speech command as well as the structure of interaction protocols) of the robot as accurately as possible, the interaction protocol interface engages the user to actively define the operative interaction protocol for the interaction task the user should fulfill.

4.1 Control Architecture FAMILIAR

Different factors are conceivable for facilitating understanding of the robot's architecture in the user: The architecture itself could be very simple or intuitive, or the architecture could be conveyed in an intuitive manner through mechanisms of explainability, legibility, transparency etc.

As we want to reduce the occurrence of errors in human-robot interaction, we consider the human-robot interaction failure taxonomy by [5] which describes various sources of errors. According to this taxonomy, faulty interactions can either be based on technical issues of the robot or incorrect input by the user. In both cases, the problem typically affects the behavior of the robot and, with this, becomes observable. Therefore, users need to understand how the behaviors of the robot are triggered. The design of our control system aims for the robot to be capable to act in a dynamic, fast-

changing environment, but also for the control system to be comprehensible for lay users. With the focus of our study of improving HRI through achieving a better understanding in the user, we reduced the complexity of the system by using a behavior-based architecture with only two components: perception and behavior guidance for execution. Additionally, the system incorporates a general visualization to make the architecture observable for users, see Sect. 4.2.1. This visualization forms the basis of our dynamic visualization, see Sect. 4.2.2. We developed our behavior system in python 2.7 [80] using the Robot Operating System (ROS) [81] as middleware. We named the architecture *FAMILIAR – FunctionAl user Mental model by Increased LegIbility ARchitecture*.

4.1.1 General Structure

The general structure of our behavior system is displayed in Fig. 1. The architecture controls the behaviors of the robot by two primary components: the `Perception` and the `Behavior Guidance`. The `Perception` perceives the environment with sensors and extracts higher-level *semantic sensors*. The `Behavior Guidance` contains the selection process of behaviors to execute. Behaviors to execute are determined by the current set of semantic sensor values. The selected behavior for execution is sent to the `Execution` component, which executes low-level actions to generate the desired behavior.

4.1.2 Perception

To perceive the world, robotic systems are usually processing perceptual information within various stages. First, the low-level hardware primitives need to observe the world and represent them in numbers. These observations are then used by algorithm primitives to interpret the observations. While all these operations are important, they are difficult for users to understand. More importantly, if an error occurs in one of these steps, non-expert users would not be able to counteract the problem. Therefore, a more abstract and comprehensible representation of the robot's perception is needed. As [82], we use so-called *semantic sensors* in our architecture. This type of sensor abstracts the low-level perception (i.e., direct sensor values as numbers) and extracts semantically simpler observations of the environment, which are less technical such as the location of the robot or user, or whether an interaction partner is detected. That way, users can more easily understand what the robot observed.

Values of semantic sensors are extracted in three steps. First, the robot uses an arbitrary number of hardware sensors together with software components to perceive the world. This can be, for example, a camera for object recognition or a microphone for speech recognition. The perceptions are sent to a memory module, where they are combined with a set of sensors that represent the world (henceforth referred to as "world state"). This world state is used to extract and update semantic sensors. These semantic sensors are then transmitted to the *Behavior Guidance* module. For example, a sensor module that recognizes people would return an array of bounding boxes, which denotes the locations of people as seen by the camera. A semantic sensor that extracts a user comprehensible meaning might be a sensor that denotes a Boolean value if a person is visible or not.

4.1.3 Behavior Guidance

The `Behavior Guidance` process is at the core of our architecture. Behaviors are not only executed based on the current set of semantic sensors but the interaction context (represented as `Interaction Protocols`) further guides the selection process. The overall guidance component was developed as simple as possible, to be comprehensible for naïve users. When errors in the interaction occur, the user should be able to trace the error back to its origin. With more complex robot control designs, the process of error tracing becomes presumably too complex for non-expert users. For example, [82] integrated a hierarchical task network planner (SHOP2) [83] to plan subsequent actions to achieve more complex behaviors of the robot.

Our system specifies an `Interaction Protocol` as a data structure to group behaviors that need to be executed to fulfill the goal of the interaction. The Interaction Protocol structure allows for more goal-specific interactions which has the potential to support comprehension for users. An Interaction Protocol is a hierarchical structure and consists of one behavior that starts the protocol, an arbitrary number of behaviors that can relate to each other, and one or more exit behaviors to end the Interaction Protocol. The data structure also stores information about the execution status of behaviors, as well as a priority value that determines the rank compared to other protocols. Interaction Protocols with a higher priority value are preferred. Moreover, it is possible to switch from an active Interaction Protocol to a higher prioritized one and later switch back. That way, timing-sensitive goals can be achieved faster.

A *behavior* is described by *preconditions* and *predecessors* that need to be satisfied for execution, and the *action* (with its parameters) that is triggered by the behavior. Preconditions specify desired values of semantic sensors for the behavior to be executable. In addition to preconditions, where semantic sensors are compared to desired values, a behavior can also define a set of predecessors that need to be completed before the behavior can be executed.

When a precondition of a behavior is satisfied and all predecessors are completed, a behavior can be executed. The *Behavior Selection* process determines the behavior to execute (cf. Sect. 4.1.4). Executing it triggers the correspond-

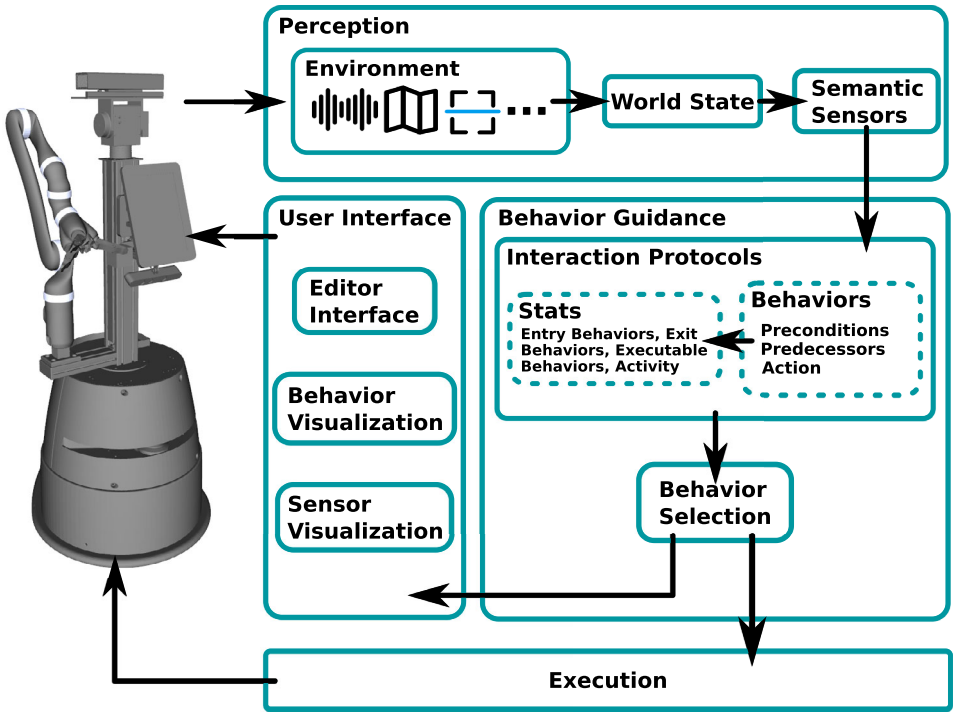


Fig. 1 Overview of the FAMILIAR robot control architecture

ing action. An action is defined by the process triggered in the `Execution` component and its parameters. The corresponding parameters can be either static values (specified in a configuration file) or dynamic values that depend on the current world state. For example, a static parameter value for a navigation action could be the goal (e.g. *kitchen*). In contrast, a dynamic parameter could define the goal as the current position of the interaction partner. The triggered action is transmitted to the `Execution` component. To achieve the goal of the behavior action, the `Execution` component executes multiple low-level actions. After the execution of the behavior action is completed, the corresponding behavior is marked as finished.

4.1.4 Behavior Selection

After semantic sensors are updated, multiple behaviors from different Interaction Protocols might be executable. Our system does not allow concurrent execution of behaviors. That way, users merely need to focus on one behavior. Therefore, a selection process has to determine the subsequent behavior to execute. This task of guiding the behavior execution is carried out by a higher-level process. This process determines which Interaction Protocol is active (and therefore preferred), and which behavior within the active Interaction Protocol should be executed.

The selection process starts with an update of the current world state. Based on the world state, the affected semantic sensors are updated. Preconditions of behaviors are subscribed to their corresponding semantic sensors. Therefore, an update of semantic sensors also leads to an update of preconditions. If a precondition is fulfilled by the update, the status of the corresponding behavior changes to *executable*.

After all semantic sensors are updated and all executable behaviors are determined, the selection process decides what behavior should be executed. In the first step, the process selects an Interaction Protocol from which a behavior should be executed. In general, the Interaction Protocol that is currently active and includes executable behaviors is preferred. The sole exception to this is the priority of Interaction Protocols. A higher prioritized Interaction Protocol is preferred to an active Interaction Protocol. This is important to prefer time-sensitive tasks. When an Interaction Protocol is selected, the behavior within the protocol needs to be determined. If the protocol is inactive yet, only the entry behavior can be executed. Otherwise, multiple behaviors may be executable. In that case, behaviors that have predecessors that were executed last are prioritized. Otherwise, behaviors are selected based on their position within the Interaction Protocol. E.g., behaviors that were defined first are also selected first.

4.2 Visualization

[84] suggest considering robot architectures for social interaction on five abstraction levels. Namely, these levels are the hardware, the algorithm, social and emergent primitives and methods for controlling these primitives. Hardware primitives include the low level hardware, e.g. the laser scanner. The algorithm primitives describe algorithms, such as speech

recognition. The next higher level of abstraction are the social primitives. These primitives are reusable units for social interaction (e.g. speaking). The emergent primitives build on this as functions that emerge through the combination of social primitives. The highest abstraction, the methods of controlling primitives, describes how the lower level abstractions are controlled. The conclusion of their investigations suggests, that the abstraction level of social primitives is most suitable for programming robot social interactions for programmers without experience in programming robots. For conveying the architecture of the robot to the user, we chose to present the architecture of the robot on the abstraction level of social primitives assuming that this level of abstraction is most comprehensible also for lay users. Along with the behavior architecture, we developed a graphical user interface to visually convey each part of the architecture. The goal was to display the inner working of the robot as accurately as possible, while still being abstract enough to be understandable. To address our first research question (cf. Sect. 3), the user interface was developed in two versions. The first version only displayed the static information of the architecture, i.e. the components and structures as described above. Based on this basic visualization, we developed an enhanced version, which also displays the dynamic process information.

4.2.1 Basic Visualization

Our interface was developed as a web application in JavaScript [85] and is divided into two parts (cf. Fig. 2): a visualization of the semantic sensors and their current values, and a visualization of the `Behavior Guidance` component. All visualization parts were integrated in the web interface of the simulation (cf. Sect. 5.4.1).

We assume that in our scenario, if the user understands and is able to track the perception of the robot, this would promote successful human-robot interaction. Our behavior system extracts semantic sensor values that abstract from the low-level perception of the robot (cf. Sect. 4.1.2). In the basic visualization, the *semantic sensor visualization* is displaying representative sensor icons side-by-side at the top of our user interface (cf. Fig. 2). We used icons to represent each sensor to facilitate a fast perception and understanding. For icons to represent the underlying semantic sensor concisely, the selection has to be made with considerable care. Our strategy was to select similar icons to known interfaces. For example, the recognized speech command could be represented by sound waves. This icon is potentially known from speech recognition systems, such as *Google assist*[1]. To further improve the understanding of the meaning of each icon, we added descriptive tooltip boxes. The tooltip of an icon was printed above when the mouse hovered over the sensor icon.

[1] https://assistant.google.com [accessed: 2021-07-05].

Our *Behavior Guidance visualization* was displayed at the center of our visualization. We focused on displaying the important parts of the system as simple as possible without simplifying or abstracting the inner working too much. To represent the Interaction Protocol structure with their behaviors, we used a network representation, using the cytoscape.js library [86]. Each Interaction Protocol is drawn as a rectangular node, including its behaviors as children. Interaction Protocols are vertically aligned, and the names are drawn above the node. The containing behaviors of an Interaction Protocol are aligned horizontally within the node. To display the behaviors in a legible way, we calculated an even distribution of the graphical positions of the behavior nodes within the Interaction Protocol node using the Reingold-Tilford algorithm [87]. The behaviors are drawn as similar rectangular nodes (cf. Fig. 3). Each node includes information about the preconditions that have to be fulfilled, as well as the corresponding action that is triggered by the behavior. To distinguish between the various behaviors immediately, a description of the action that is triggered by the behavior is displayed as the title. The preconditions are visualized with the corresponding semantic sensor icons. To receive more detailed information about a behavior, clicking on the behavior node displays additional information on the right side. In addition, entry behaviors have big double right arrows on the left side, indicating the entry point to the corresponding Interaction Protocol. In contrast, exit behaviors have the arrows on the right side, indicating the exit point of the Interaction Protocol. To express the predecessors of a behavior, arrows are drawn from the behavior node towards the predecessor node.

4.2.2 Dynamic Visualization

The dynamic visualization is an enhanced version of the basic visualization described above. Whereas for the *semantic sensor visualization*, the basic visualization only displays representative icons for each sensor, the dynamic visualization also displays the current value of the sensor. The first icon in Fig. 2 for the location of the robot shows the label of the region the robot is in (e.g., 'kitchen') and 'unknown' when the robot is located in an unknown region. The second 'soundwave' icon would display the recognized sentence. The third icon shows, analogously to the first icon, the location of the avatar. The fourth icon shows 'False' when the avatar is not in sight and 'True' when the avatar is in sight.

For the *Behavior Guidance visualization*, the enhanced dynamic visualization adds coloring and highlighting as mechanisms to indicate the processes of the architecture. For the preconditions, we used colored icons to indicate the satisfaction status. Behaviors of an inactive Interaction Protocol have black colored precondition icons, indicating that the corresponding preconditions are not updated. Otherwise, a

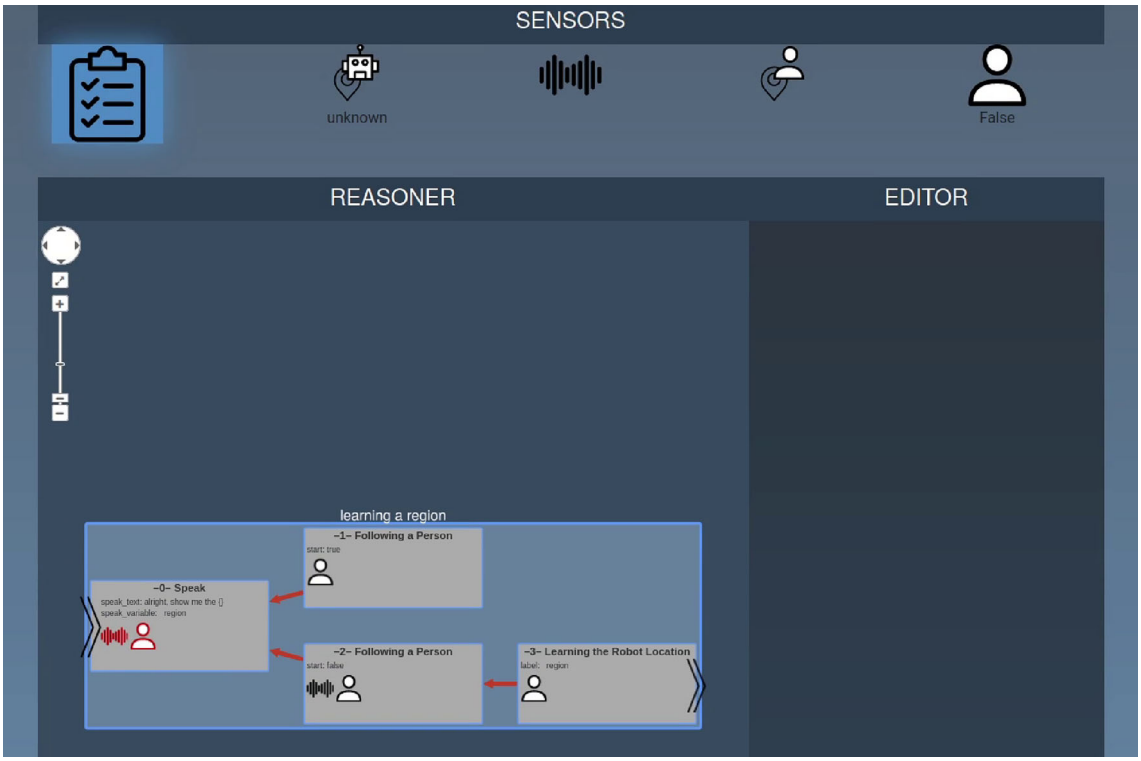


Fig. 2 Architecture Visualization Overview. The sensors are displayed at the top. In the bottom left, the interaction protocols and behaviors are displayed. To the right, details of behaviors are displayed. In addition, new behaviors are defined there. For details on sensor values refer to Sect. 4.2.1

fulfilled condition is displayed in green. In contrast, unfulfilled conditions are displayed in red. Predecessor arrows are colored red while the predecessor is unexecuted, otherwise it is colored green. To highlight the selection process, we used *border* and *background* colors, varying sizes, and zooming in on nodes. Nodes of inactive Interaction Protocols and behaviors have blue border colors and light gray backgrounds. Active Interaction Protocols and executable behaviors have green border colors. When a behavior is executed, the background color of the corresponding node turns green and is zoomed in. After the execution of a behavior is finished, the background and border color turn dark gray to indicate this behavior has already been executed in the current interaction. Only after the respective Interaction Protocol finished, the behavior border-color changes back to blue, and the background color changes to light gray again.

4.3 Interaction Protocol Interface

Our second approach, to improve the users' knowledge about the architecture of the robot, is the active definition of the interaction protocol via an interface inspired by visual programming interfaces. We aim to investigate whether, through the activity, users develop a deeper understanding of the robot's architecture through for example having to process their own role in the interaction, as well as think through the robot's role. Therefore, our interface is designed towards transparency, while we accept a loss of user satisfaction. To enable naive users to engage in the definition of the robot interaction protocol, we developed an editor interface accordingly as well as provided the users with an interactive tutorial.

The editor interface allowed the users to add new Interaction Protocols and define the containing behaviors with their preconditions and actions. This editor interface is structured similarly to filling out a form. The most important part of this is the behavior definition (see Fig. 4). This behavior can be selected to be an *entry*– or *exit* behavior via checkboxes. The predecessor of a behavior can be selected by entering the ID of the preceding behavior. The preconditions are defined by selecting the corresponding sensors and their expected values in drop-down menus. Similarly, the action with its parameters is added.

To guide participants of the study through the process of defining an Interaction Protocol, we developed an interactive tutorial (see Fig. 4). In the tutorial, participants had to read text boxes with information and had to interact with the editor interface. In this way, we were able to introduce the interface, as well as ensure that all participants define the same Interaction Protocol for the following interaction. Each step of the tutorial was only shown once. The information presented in the tutorial was only descriptive of what is shown and which value an input field needs. Therefore,

Table 1 Abbreviations for experimental conditions

		Interaction Protocol	
		On	Off
Dynamic Visualization	On	IPI+DV	DV
	Off	IPI	Baseline

no additional architectural knowledge about the robot was communicated.

The tutorial steps to define the first behavior were designed to be highly detailed, while the other behaviors were only one tutorial step each, describing what the behavior should look like. In that way, we wanted the participants to not only follow each step, but elaborate on the final interaction protocol.

5 Exploratory User Study

5.1 Experimental Design

An exploratory user study was carried out using the online platform prolific[2]. To investigate the impact of our two complementary approaches to increase the legibility of the robot behavior architecture, we employed a 2-by-2 between-subjects design with the two factors: *Interaction Protocol Interface (IPI)* (on vs. off) and *Dynamic Visualisation (DV)* (on vs. off). More precisely, we varied the way users interacted and explored the robot system by either having them engage in defining a robot interaction protocol for realizing the task (IPI on vs. off) or by changing the availability of online information during the interaction task (DV on vs. off). We denote the condition in which neither the dynamic visualization was available, nor did participants engage in the definition of an interaction protocols as *Baseline*. Table 1 summarizes all abbreviations that are used for labeling the different experimental conditions in the following.

5.2 Participants

85 participants took part in the study, of which 4 were excluded due to technical problems. The remaining 81 participants were randomly assigned to the four conditions (`Baseline`: $n = 20$, `DV`: $n = 21$, `IPI`: $n = 20$, `DV+IPI`: $n = 20$). The average age of participants was 35 (SD=9.53) years. Overall, 30 female, 50 male and 1 diverse persons participated. A Kruskal-Wallis test [88] showed no significant difference between the conditions regarding age ($H(4) = 3.9091$, $p = 0.2714$) or gender ($H(4) = 0.9606$, $p = 0.8108$). All participants were born and located in the United States and were native English speakers.

To further characterize the participant sample, we measured the technical affinity of participants using the ATI scale [89]. Moreover, we measured the participants' memory capacity with a word memory test [90]. Again a Kruskal-Wallis test [88] showed no significant difference between the experimental conditions regarding the ATI score ($H(4) = 0.8627$, $p = 0.8344$) or word recall ($H(4) = 0.5732$, $p = 0.9025$), which rules them out as potential confounding variables.

5.3 Interaction Task

The task of the participants was to teach the robot about three regions in a simulated apartment. To that end, both the robot and the user were represented in a simulated apartment. The user was visualized by means of an avatar (see Fig. 5), while the robot was displayed as the Scitos (see Fig. 1 on the left). The apartment consisted of x rooms. Participants had a bird-eye view on the simulated apartment, which enabled them to oversee the whole scene at all times. Participants could move their own avatar by clicking on a desired goal position on the map. To complete the task, participants had to start an interaction with the robot, guide it to a target region in the apartment and teach the respective label for the region to the robot. To start an interaction, participants had to move the avatar in sight of the robot and give the first command. As a first step, the robot had to follow the avatar to the region to be taught. This behavior was triggered by the first command ("I will show you the [*name of the region to be learned*].") which also provided the label to associate with the region to the robot. The user then had to move the avatar around the apartment to guide the robot to the region to learn. When the avatar and the robot arrived at the location to learn, the user had to inform the robot that they arrived at the target location (command: "Here we are.") This triggered the robot to learn the current x-y location of the robot and associate it with the previously provided label.

All commands were given in text form via the keyboard. The region learning was realized with a scikit[3] implementation of the k-nearest neighbor algorithm. Therefore, for learning a region an x-y coordinate had to be stored together with a region name as the label. When a region in the apartment was successfully taught to the robot, it appeared with a colored texture on the floor. An example of how a classified apartment could have looked like is shown in Fig. 5.

To realize this scenario, we developed several sensors and actuators for the robot. To allow the user to interact with the robot, we equipped the robot with the ability to recognize the avatar of the user and to interpret natural language. To simulate how person recognition would be in the real world, the robot was only able to recognize the avatar around a certain

[2] www.prolific.co [accessed: 2021-07-05].

[3] https://scikit-learn.org [accessed: 2021-07-05].

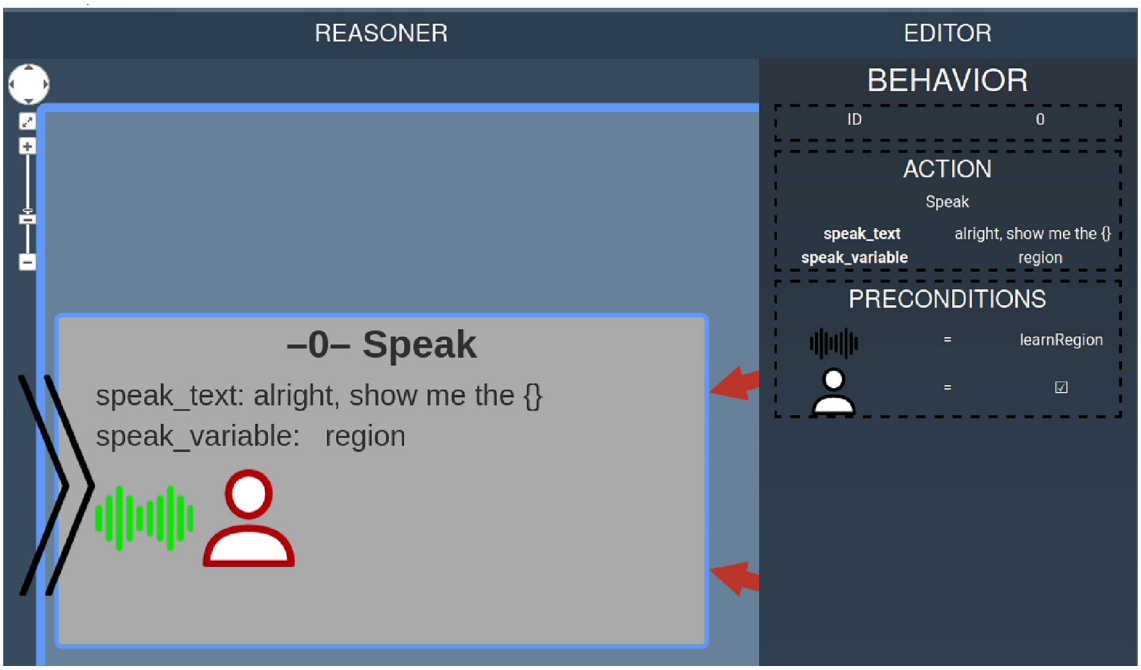


Fig. 3 A close-up view of a behavior. When clicking on a behavior, this behavior is zoomed in and details are displayed on the right side. Here, behavior 0, Speak, is shown which triggers the action of producing the utterance "Alright, show me the {region}," (*region* being a variable) upon the preconditions that the command for learning a region (i.e., learnRegion) has been recognized and the user is visible

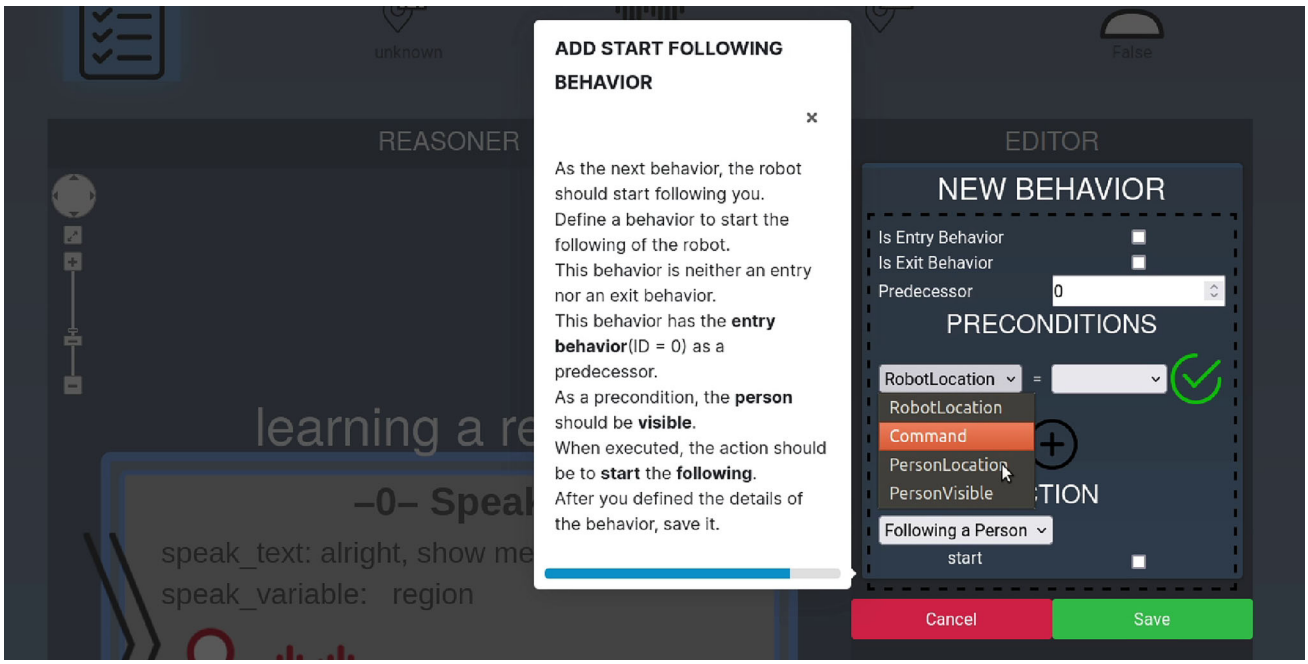


Fig. 4 The behavior editor interface with a tutorial step. The tutorial step describes how the behavior should be defined. In the interface, all parts of a behavior can be defined

radius from itself. For this, we calculated the Euclidean distance between the current position of the robot and the avatar. This distance determined, whether the avatar is in sight of the robot. This represented approximately the real behavior of the person recognition component, which requires users to be well visible and close enough. For the natural language understanding, we used the *Snips NLU* [91] project. We trained the NLU module to understand more commands than those necessary to achieve the interaction goal. The module also supported the recognition of slight variations of the intended commands. This allowed us to differentiate, later, if participants of the study only applied the exclusion procedure or understood how to interact with the robot. To interact with the environment, the robot was able to verbally answer the user, via a chat box. Moreover, the robot was able to navigate through the apartment and follow the avatar.

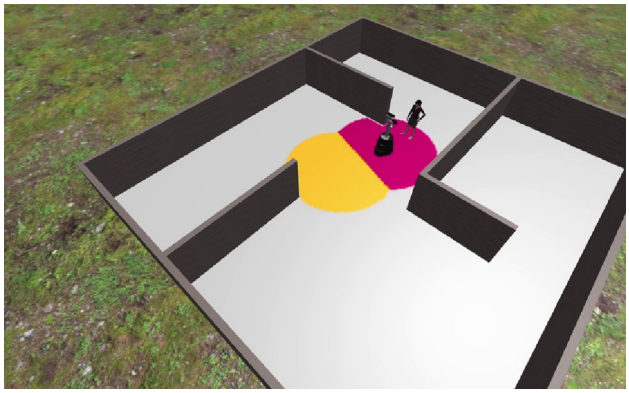


Fig. 5 Example of an apartment with two regions taught to the robot. The classification of the robot is indicated on the floor. The yellow region is the kitchen, the purple region is the entrance and the white areas are unclassified

5.4 Experimental Setup

An online simulation was realized to conduct the interactive user study. The simulation was developed with Gazebo [79] and ROS. For users to interact with the simulation remotely, we implemented a web visualization using gzweb[4]. For easy deployment, the simulation was built as a docker [92] image and ran on university servers.

5.4.1 Web-Interface

The Web-Interface to interact with the simulation consisted of five parts (cf. Fig. 6). In the top left corner was the graphical interface of the gazebo simulation located. On the right side was the Chat-Box interface. The visual components of the control architecture were displayed below the simulation and Chat-Box interface.

5.4.2 Gazebo-Simulation

The visualization of the gazebo simulation was realized with gzweb. The camera view was fixed at one location and was oriented towards the apartment. Therefore, the user was able to see the avatar, the robot, and the whole apartment. The height of the walls of the apartment together with the height of the camera was adjusted to prevent the users' view from being blocked. To reduce the amount of resources that need to be loaded, and to reduce the calculation time of the simulation, we excluded any furnishings.

The user was able to move the avatar, by clicking in the simulation window. The 2D position of the click event was then projected on the floor plane of the simulation. The avatar's position was updated with the new position. Therefore, the avatar did not move continuously towards the new location, but directly appeared there. Because this mechanism sometimes resulted in the avatar falling over, the physics calculations of the avatar were disabled.

To prevent the avatar from appearing outside the apartment, in a wall, or too close to the robot, the position of the click-event was checked to be free of obstacles and within the borders of the apartment. In case the user clicked in an unreachable position, a pop-up window appeared, which informed the user that this location is unreachable. (The starting positions of the robot and the avatar were so far apart, that the avatar was not in sight of the robot.)

5.4.3 Chat-Box

To verbally interact with the robot, we developed a chat-box window. For ease of access, the design was inspired by common messengers, such as WhatsApp[5] or Facebook Messenger[6]. The main window displayed the communication history, while the bottom window consisted of elements to write new messages.

The messages of the user were displayed in speech bubbles on the right side. Because the robot could only understand certain commands, the color and content of these speech bubbles depended on the content of the message. If a message could not be interpreted by the robot, the speech bubble was colored *red* and the message was displayed in it. In the case, that the message could be interpreted by the robot, the speech bubble was colored *green*. In Addition, the content of the message, as well as the classified intent by the robot, was displayed in the speech bubble. The answers of the robot were displayed in a white speech bubble on the left side. In addition to the verbal communication of the user and the

[4] http://gazebosim.org/gzweb [accessed: 2021-07-05].

[5] www.whatsapp.com [accessed: 2021-07-05].

[6] www.messenger.com [accessed: 2021-07-05].

robot, the chat-box window also displayed the (de-)activation events of the robot control architecture.

The bottom window, to write new messages, consisted of three elements. At the center of this window, the message could be entered in an input field. This message could be sent by clicking on the *send*-button on the right side. In addition, an *information*-button was displayed on the left side. Clicking this button displayed a pop-up window, presenting the intents the robot was able to understand, as well as example sentences for each intent.

5.5 Experimental Procedure

Participants were forwarded to our study website, where they were assigned to one of the four conditions. The study procedure can be seen in Fig. 7. First, all participants were shortly briefed about the structure of the study and the estimated time scope. Afterwards, their gender, age, and prolific-id were gathered.

In a next step participants were shown an instruction video, introducing the simulation and the robot control architecture. Depending on the condition, participants were forwarded to one of two stages. The `Baseline` and `DV` conditions were shown another video, explaining at the *relevance* level, how to achieve the interaction goal. In contrast, the `IPI` and `IPI+DV` conditions followed the tutorial to program the behaviors.

In the next phase of the study, all participants interacted with the robot, trying to accomplish the experimental task (cf. Sect. 5.3). To have the same amount of time for each participant, we limited the interaction with the robot to thirty minutes (to keep the overall length of the study reasonable and since we observed that it was enough time for most participants to complete the task successfully). After the interaction, participants were forwarded to the knowledge questionnaires. Participants who did not finish the experimental task within the time limit were asked why they could not achieve it before they were also forwarded to the knowledge questionnaires.

To receive more subjective ratings about the system, after the knowledge questionnaires, we collected parts of the Godspeed questionnaire [93] and the System-Usability-Scale (SUS) [94]. We limited the Godspeed questionnaire to the scales of *Anthropomorphism*, *Likability* and *Perceived Intelligence*. These scales were most relevant to rate the technical system. To end the study, participants were thanked for their participation and forwarded to prolific for their payment.

5.6 Measures

5.6.1 Architectural Knowledge

To measure the participants' architectural knowledge we asked 19 multiple-choice questions about the robot's functionality. These questions are a direct measure of the subject's acquired knowledge of the system. Each question consisted of three statements, of which only one statement was correct. Questions targeted the following categories: *structural* knowledge and *process* knowledge. We define structural knowledge as knowledge about the conceptual elements of the system and how they interact ('knowing that') and process knowledge as knowledge about how to interact with the system to achieve the task at hand ('knowing how') [95]. *Structural* knowledge included questions on `Sensor`, `IntProtocol`, `Behavior`, `Precondition`, `Action`, and `Predecessors`. *Process* knowledge questions only had one category `Process`. All questions could be answered based on the information provided in the introduction video. Therefore, all experimental conditions were given the same foundation to answer these questions. The respective experimental conditions including the dynamic visualization and interaction protocol definition should have only helped to consolidate the knowledge.

5.6.2 Interaction Success

As a measure of overall task success, participants were divided into two groups based on whether they achieved the interaction goal: interactions of participants who taught all three regions to the robot within the time limit, were classified as `successful`, otherwise the interaction was labeled `unsuccessful`.

5.6.3 Interaction Failures

To further evaluate failures in the interaction, we counted the number of wrong commands provided by the user. Thereby, we also classified a command as wrong, if it was unneeded to reach the interaction goal. That is, a command could be wrong even though the robot was able to interpret it. Moreover, we counted the amount of times users moved the avatar out of sight of the robot. Thus, indicating how aware participants were about the perception limitations of the robot. These results were used as implicit measures for the interaction success.

5.6.4 Godspeed, SUS

We collected data of the Godspeed questionnaire (Anthropomorphism, Likability, Perceived Intelligence) and the System-Usability-Scale. In contrast to the objective measures

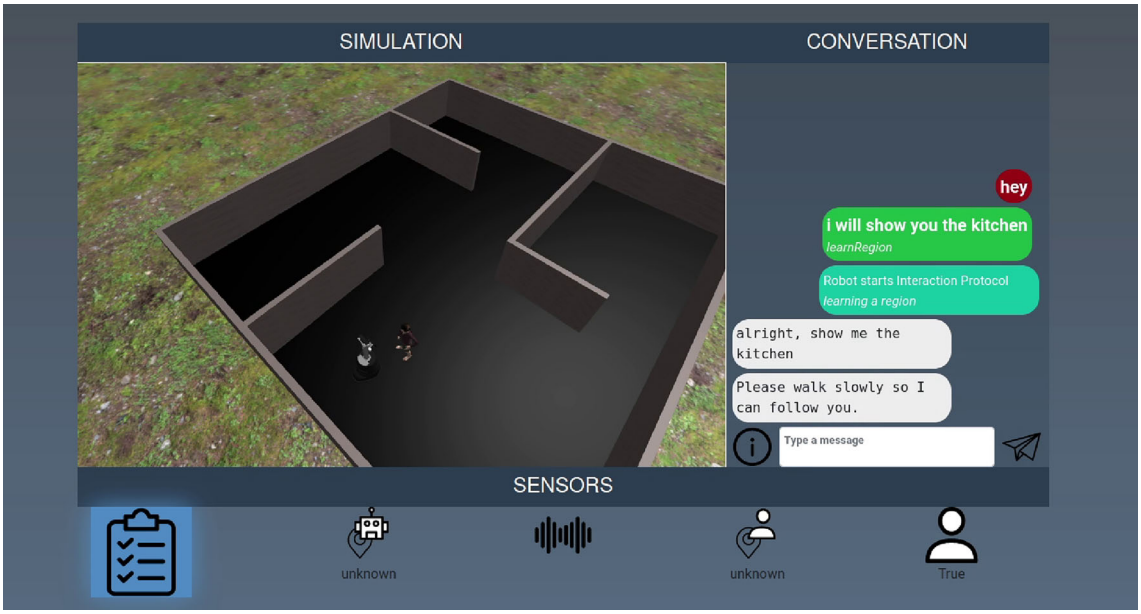


Fig. 6 The Simulation Web-Interface. The simulation visualization is displayed in the top left. On the right side, the chat-box interface to communicate with the robot is displayed. Below, the *FAMILIAR*-architecture visualization is located

described above, these results were used to get insights of the subjective ratings of the interaction.

6 Results

For a first overview, we assessed the overall scores of correctly answered questions for the knowledge questionnaires. For this, the mean correctness of the multiple-choice questions was calculated. While the `Sensor` (77.46%) concept could be answered by many participants, the `IntProtocol` (17.90%) and `Predecessor` (28.40%) concepts had a low correct answer rate. In contrast, the concepts `Behavior` (51.85%), `Precondition` (52.26%), `Action` (58.64%), `Process` (40.12%) and `All` (46.66%) could be answered by approximately half the participants. Detailed results for the individual architecture components can be found in Fig. 8.

In the following, we will discuss the results in detail with respect to our research questions.

6.1 RQ1: Can the proposed process visualization improve process knowledge?

To gain insights in our first research question, we compared questionnaire scores in the process knowledge category between each experimental condition and the `DV` and `DV+IPI` conditions pooled together across all participants with a Kruskal-Wallis H test. No significant differences between the conditions ($\chi^2(3) = 2.23, p = 0.53$) or the pooled conditions ($U = 762.0, p = 0.56$) could be found, indicating that on average this experimental manipulation did not improve process knowledge.

6.2 RQ2: Does defining an interaction protocol improve structural knowledge about the robot?

We investigated whether visually programming the interaction would improve the structural knowledge about the robot which corresponds to the concepts `Sensor`, `IntProtocol`, `Behavior`, `Precondition`, `Action`, and `Predecessor`. To evaluate this, we compared the questionnaire scores between the conditions including the definition of an interaction protocol taken together (`IPI`, `IPI+DV`) and the conditions *without definition of an interaction protocol* (`Baseline` and `DV`) pooled together across all participants. Results can be found in Table 2.

A Shapiro-Wilk normality check with a follow-up t-test revealed that participants that defined an interaction protocol had significantly more knowledge about the overall architecture ($t(81) = -2.83, p = 0.006$).

In addition, pairwise comparisons were done for each subcategory of the structural knowledge. A Mann–Whitney-U test showed that knowledge about the `Behavior` concept was also significantly increased ($U = 520.50, p = 0.00$).

6.3 RQ3: Can architecture knowledge improve human-robot interaction?

To tackle this question, we considered first the number of successful interactions. As it can be seen in Fig. 9, 80.95%

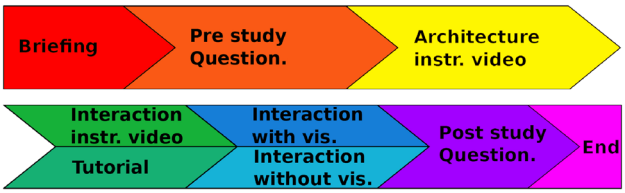


Fig. 7 Study schedule. 1. (red) Participants were briefed and 2. (orange) had to answer preceding questionnaires. Afterwards, 3. (yellow) the architecture instruction video was shown. 4. (green) Depending on the condition, participants completed the tutorial or watched an interaction instruction video. In the main part 5. (blue), participants interacted with the robot to achieve a joint goal (either with or without visualization). After the interaction, 6. (purple) post-study questionnaires were collected. Then 7. (pink) the study ended. (Color figure online)

(17) of the participants in the DV condition achieved the interaction goal, while only 65% (13) of the Baseline and 60% (12) of the IPI and IPI+DV conditions achieved the goal. To further investigate the influence of architectural knowledge on the interaction, we compared the successful (n = 54) and unsuccessful (n = 27) interaction groups. We applied a Shapiro-Wilk normality check and a Mann–Whitney-U test for non-normal distributed data and a t-test otherwise. The results can be seen in Table 2 and Fig. 8. Overall, the results revealed that participants who achieved the interaction goal had more knowledge about the robot's architecture ($t(81) = 2.25, p = 0.03$). A closer look at each category revealed that knowledge about Behavior ($U = 945.0$, $p = 0.02$), Precondition ($U = 935.0$, $p = 0.03$) and Process ($U = 925.0, p = 0.03$) were higher for participants who achieved the interaction goal, but results do not reach statistical significance after correcting for multiple testing (Bonferroni corrected alpha level).

Besides, we investigated the number of wrong commands as another indicator for interaction success that is assumed to be related to accurate architecture knowledge. To compare participants who used a high number of wrong commands with those who used fewer wrong commands, we calculated a median split for the number of wrong commands. Again, we compared the answer score of each knowledge category and the overall score with a Mann–Whitney-U or t-test. Results can be seen in Table 2. Results showed a tendency for differences in the overall knowledge ($T(81) = 1.92$, $p = 0.05$), the IntProtocol($U = 677.0$, $p = 0.1$) and Behavior($U = 988.50$, p=0.08) concepts, as well as the Process knowledge ($U = 1068.0, p = 0.01$)) suggesting that participants with a low number of wrong commands had more architectural knowledge. Yet, results did not reach statistical significance after Bonferroni correction.

Next, we investigated how often the participants moved the avatar out of sight of the robot. For this, the dynamic visualization conditions (DV+(IPI+DV)) were compared with the conditions without the dynamic visualization (Baseline+IPI). We only analyzed the successful interactions, because some participants, who did not achieve the interaction goal, never moved the avatar (due to not finishing the tutorial). Thus, they also never moved the avatar actively out of sight of the robot. A Mann–Whitney-U test showed a tendency of decrease in the number of avatar losses between the *no dynamic visualization* ($\mathcal{M} = 9.16$, $\mathcal{SD} = 4.95$) and the ($\mathcal{M} = 6.72$, *dynamic visualization* $\mathcal{SD} = 4.30$) conditions ($U = 468.0$, $p = 0.07$).

6.4 Subjective Ratings of the Interaction

In addition to the knowledge questionnaires, we also explored the subjective user experience in the interaction with the robot via the data collected with the godspeed questionnaire and the SUS with a Kruskal-Wallis H test. No differences for the *likability* ($\chi^2(3) = 4.16$, $p = 0.24$), the *perceived intelligence* ($\chi^2(3) = 4.19$, $p = 0.24$) or the *anthropomorphism* ($\chi^2(3) = 2.17$, $p = 0.54$) score could be found. In a recent meta-analysis, it was emphasized that in social robotics it is seen as a positive result if users anthropomorphize a robot as this may indicate that the robot is seen as a social entity ([96], cf.). However, the authors also revealed that anthropomorphism did not necessarily have a positive effect on task success ([96], cf.). To explore the influence of the level of anthropomorphization on the interaction success in our experiment, as well as on the SUS and ATI score, we applied a median split on the anthropomorphism score. We first compared the interaction success of participants with lower anthropomorphism score with the interaction success of participants with a higher anthropomorphism score (cf. Fig. 10). A Shapiro-Wilk normality check (*statistic* = 0.9096, p = 0.0000) with a Mann–Whitney-U test ($U = 488.50$, $p = 0.05$) revealed that participants with a lower anthropomorphism score had a higher likelihood to achieve the interaction goal (cf. Table 3). We then compared the SUS and ATI for the anthropomorphism median split. A Mann–Whitney-U test revealed that higher anthropomorphism is related to a higher rating of the system usability ($U = 417.50$, $p = 0.00$), as well as a higher ATI score ($U = 602.50$, $p = 0.04$). Additionally, a Mann–Whitney-U test showed a significantly lower SUS for participants in the IPI conditions ($U = 1089.50$, $p = 0.01$).

7 Discussion

The analysis of our user study revealed some interesting insights into the effects of robot architecture knowledge on the interaction quality. Participants, who achieved the interaction goal, seem to have more knowledge about the robot's architecture, though differences concerning different concepts in closer inspection, were not significant after Bonferroni correction.

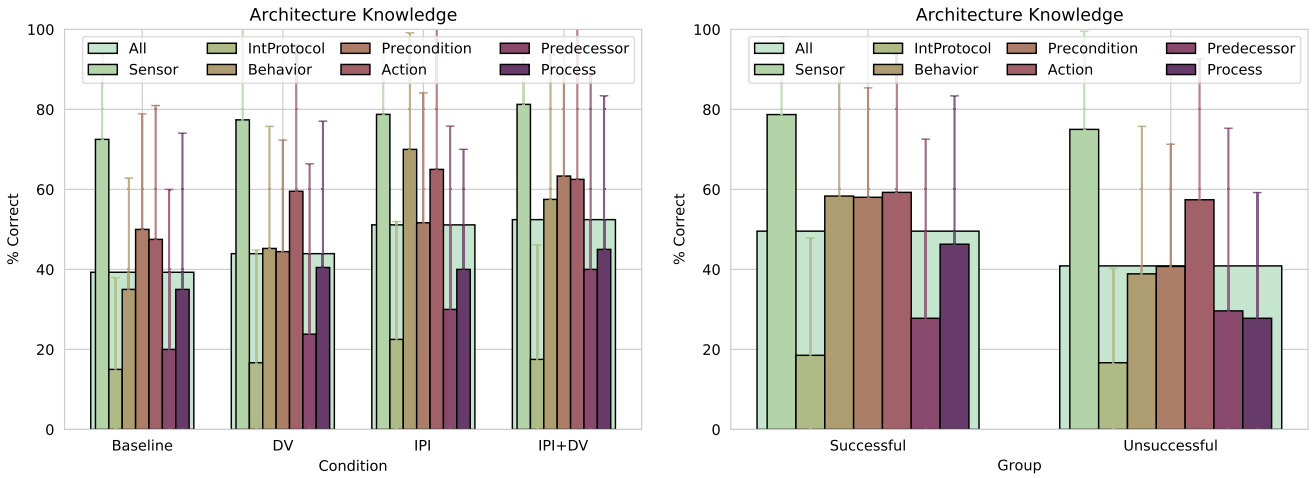


Fig. 8 Architecture Knowledge for each condition (left); and groups `successful` and `unsuccessful` (right)

Table 2 Statistical analyses of the architecture knowledge

Category	Int. Prot.		Success		Wrong commands	
	Statistic	*p*-value	Statistic	*p*-value	Statistic	*p*-value
Sensor (mwu)	765.0	0.58	770.50	0.66	762.50	0.56
`IntProtocol` (mwu)	768.0	0.55	729.0	0.99	677.0	0.09
`Behavior` (mwu)	520.50	0.00	945.0	0.02	988.50	0.08
`Prec.` (mwu)	650.0	0.08	935.0	0.03	983.50	0.10
`Action` (mwu)	693.50	0.2	753.0	0.8	925.50	0.26
`Pred.` (mwu)	713.0	0.2	715.50	0.87	875.0	0.51
`Process` (mwu)	754.0	0.50	925.0	0.03	1068.0	0.01
All (t-test)	−2.83	0.006	2.25	0.03	1.92	0.06

The first analysis ('Int. Prot.') compares conditions where an interaction protocol is defined (IPI, IPI+DV) with conditions where no interaction protocol is defined (Baseline, DV), reported in Sect. 6.2. The second analysis ('Success') compares successful and unsuccessful interaction groups and the third analysis ('Wrong Commands') compares interactions with a low number of wrong commands with those with a high number, both reported in Sect. 6.3. Tests were either Mann–Whitney U-test or t-test depending on the results from the Shapiro-Wilk normality check

Table 3 Statistical tests of anthropomorphism median split for *interaction success*, ATI and SUS

Test	Interaction success		ATI		SUS	
	Statistic	*p*-value	Statistic	*p*-value	Statistic	*p*-value
sw	0.91	**0.00**	0.97	**0.00**	0.96	**0.00**
mwu	488.50	**0.02**	602.50	**0.04**	417.50	**0.00**

p-values ≤ 0.05 are indicated by bold font

Concerning the definition of an interaction protocol as an approach to improve knowledge about the structure of the architecture, the results revealed that participants of the `IPI` conditions had increased knowledge about the architecture. Especially the `Behavior` knowledge was significantly improved. While our interaction protocol interface showed positive effects on the knowledge, not all aspects of our architecture could be conveyed. Additionally, a negative side effect, in the form of a lower SUS rating, could be observed. Therefore, further work needs to investigate how an interface for defining interaction protocols should be designed to successfully communicate all aspects of the architecture. For this, an increased focus should be put on the `Precondition` and `Process` knowledge. We expect that the abstract mechanism of defining the interaction protocol, together with the highly flexible way people act, has a major influence on the difficulty to understand these concepts. Even though `Preconditions` in our architecture are more flexible than in e.g. state machines, they still expect the environment to be in a specific state. In contrast, humans can reason about their environment and other humans on a higher level. To overcome these false assumptions, the

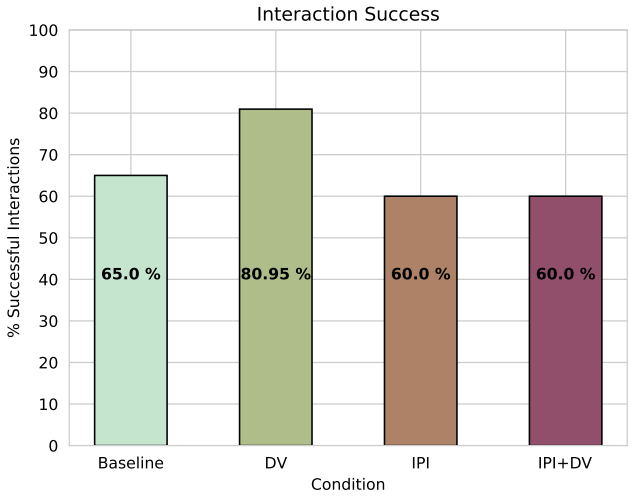


Fig. 9 Amount of successful interactions per condition

interaction protocol interface should be closely related to the interaction itself. In that way, users might better understand the impact of the concepts on the interaction.

The dynamic visualization of our architecture did not lead to a measurable improvement of the knowledge. Also, the effect of an improved interaction could not be observed for the combination of the IPI and the DV in the `DV+IPI` condition. One reason might be a cognitive overload. Presumably, a more accessible interaction protocol interface together with the dynamic visualization could have improved the interaction.

Besides the influence of the communicated knowledge for improved interaction, we observed that participants with lower anthropomorphism scores were more likely to achieve the interaction goal. This is in line with previous literature [42, 96]. For example, in the work of [42], participants with a lower anthropomorphism score were more likely to detect errors in a human-robot interaction.

This suggest that in contrast to many current approaches, inducing an anthropomorphic image on users [97] may have unwanted effects on the interaction success.

7.1 Limitations

Simulated environments may not fully capture the complexities of physical presence and behavior exhibited by robots, potentially affecting the perception and evaluation of the robot's actions and functioning as shown in other teaching scenarios [98]. Virtual or augmented reality (AR) could offer more immersive experiences compared to flat simulations, possibly altering user perceptions further, while studies involving real robots provide the most authentic insights but are often more logistically challenging to conduct. Some works have suggested that AR headsets affect users' interpretations of robots [51, 99]. The exact impact on the mental model of the user in teaching/learning interactions remains an open question.

In addition, future work should address the reasons for the unsuccessful interactions that have been observed here more explicitly. For the 27 participants that did not reach the

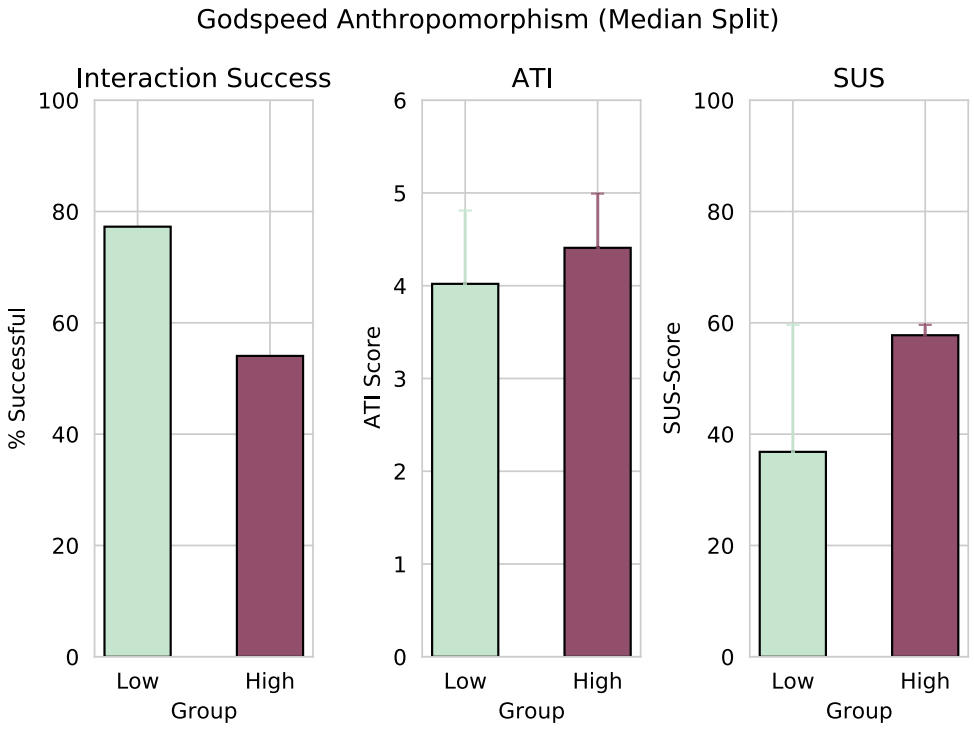


Fig. 10 Comparison of godspeed anthropomorphism median split for *Interaction Success*, *ATI* and *SUS*

interaction goal, 18 participants did not even teach one region of which 14 participants were in one of the IPI conditions. We can not say whether this stems from a lack of understanding of the experimental protocol or a specific misunderstanding or lack of information in the interface or interaction design due to our experimental setup.

Examples from investigating the log data of the visual programming condition showed that some participants may had difficulties specifying the predecessor and add preconditions to the behaviors. In addition, we found that errors in the interaction often arose from using wrong commands that the robot could not interpret. This could indicate that users assumed that the robot was capable to engage in a chat-bot like interaction and was not restricted to the use of specific commands on the interaction. A third example for a typical error in the interaction flow showed in the lack of triggering the *learning* command after having successfully completed the *following* behavior. All of these examples indicate difficulties of some users with understanding the interaction protocol or with using the visual programming interface. Furthermore, some of the users may have not been aware of the limitations of the robot (with respect to e.g. its language processing capabilities) which resulted in unsuccessful interactions.

To gain further insights into the underlying mechanisms that have moderated interaction success/failure, this should be more deeply investigated in the future. The results of our exploratory study may help thereby to develop more fine-grained explanatory hypothesis that could be leveraged and experimentally tested in future work.

8 Conclusion and Future Work

In this work, we investigated the influence of acquired knowledge about the robot control architecture for successful human-robot interactions. For this, we employed a behavior-based architecture and evaluated two complementary approaches (a dynamic visualization of the system's processes and an interaction protocol interface) which aim at increasing the user's knowledge about the system to non-expert users.

In a first exploratory simulation user study, we found that the employed interaction protocol approach improved the overall knowledge of the architecture. However, knowledge about some concepts could not be improved significantly (only the overall architecture and the Behavior concept were improved; not the Sensor, IntProtocol, Precondition, Action, and Predecessor concepts). Future work should disentangle in greater detail which aspects of our proposed approach contribute to an increase of successful human-robot interactions and which aspects should be revisited. We believe that our study provides first insights into this question and a basis for generating new experimental hypotheses that can support the future development of user-friendly human-robot interaction design.

Appendix A: Questionnaire items for Architecture Knowledge

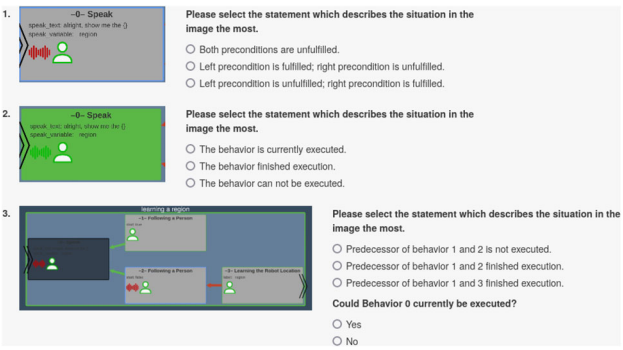
Please select the statements which are most applicable.

1.

- ○ Behaviors define how the robot behaves.
- ○ Behaviors define when the robot behaves.
- ○ Behaviors define when and how the robot behaves.

2.

- ○ Behaviors consists of preconditions, interaction protocols and actions
- ○ Behaviors consists of preconditions and actions
- ○ Behaviors consists of preconditions, predecessors and actions.



Please select the statements which are most applicable.

1.

- ○ Interaction Protocols define how the robot interacts.
- ○ Interaction Protocols define the context of the interaction.
- ○ Interaction Protocols define when the robot interacts.

2.

- ○ Interaction Protocols consists of one or more behaviors.
- ○ Interaction Protocols consists of preconditions and behaviors.
- ○ Interaction Protocols consists of preconditions and actions.

Please select the statements which are most applicable.

1.

- ○ A Precondition defines when a behavior should be executed.
- ○ A Precondition defines how a behavior should be executed.
- ○ A Precondition defines when AND how a behavior should be executed.

2.

- ○ A Precondition consists of a sensor, a desired value and an action.
- ○ A Precondition consists of a sensor.
- ○ A Precondition consists of a sensor, an operator and a desired value.

3.

- ○ Each behavior needs more than one precondition.
- ○ Each behavior needs one or more preconditions.
- ○ Each behavior can only have one precondition.

Please select the statements which are most applicable.

1.

- ○ Sensors are updated based on changes in the environment.
- ○ Sensors are updated based on updates of preconditions.
- ○ Sensors are updated based on the execution of behaviors.

2.

- ○ A precondition is updated, based on the value of the corresponding sensor.
- ○ A precondition is updated, based on the execution of the corresponding action.
- ○ A precondition is updated, based on the status of the corresponding behavior.

Please select the statements which are most applicable.

1.

- ○ An action defines what the robot did.
- ○ An action defines what the person did.
- ○ An action defines what the robot will do.

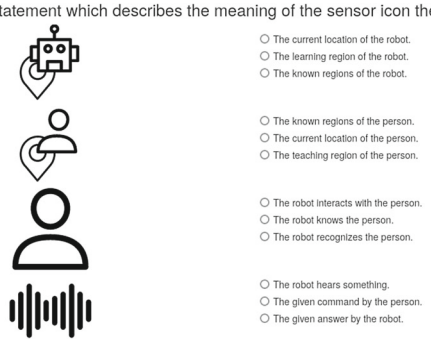
2.

- ○ Each behavior executes multiple actions.
- ○ Each behavior executes one or more actions.
- ○ Each behavior executes only one action.

3.

- ○ A predecessor describes preceding behaviors.
- ○ A predecessor describes preceding interaction protocols.
- ○ A predecessor describes preceding actions.

Please select the statement which describes the meaning of the sensor icon the most.



Funding Open Access funding enabled and organized by Projekt DEAL. This work was supported by the Honda Research Institute Europe GmbH, Carl-Legien-Strasse 30, 63073 Offenbach, Germany. Additionally, the research was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation): TRR 318/1 2021 – 438445824.

Data Availability The datasets generated during and/or analysed during the current study are available from the corresponding author on reasonable request.

Declarations

Conflict of interest Christiane B. Wiebel-Herboth and Heiko Wersing are employed by the Honda Research Institute Europe GmbH. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Ethical approval This study was carried out in accordance with the recommendations of the local Ethics Committee. The protocol was approved by the Ethics Committee.

Consent to participate Informed consent was obtained from all individual participants included in the study.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

References

1. Sendhoff B, Wersing H (2020) Cooperative intelligence-a humane perspective. In: 2020 IEEE international conference on human-machine systems (ICHMS), pp 1–6. IEEE
2. Krüger M, Wiebel CB, Wersing H (2017) From tools towards cooperative assistants. In: Proceedings of the 5th international conference on human agent interaction, pp 287–294

3. Johnson RC, Saboe KN, Prewett MS, Coovert MD, Elliott LR (2009) Autonomy and automation reliability in human-robot interaction: a qualitative review. In: Proceedings of the human factors and ergonomics society annual meeting, vol 53, pp 1398–1402. SAGE Publications Sage CA: Los Angeles, CA
4. Tsarouhas PH, Fourlas GK (2016) Mission reliability estimation of mobile robot system. Int J Syst Assur Eng Manag 7(2):220–228
5. Honig S, Oron-Gilad T (2018) Understanding and resolving failures in human-robot interaction: literature review and model development. Front Psychol 9:861
6. Brooks DJ (2017) A human-centric approach to autonomous robot failures. PhD thesis, University of Massachusetts Lowell
7. Steinbauer G (2012) A survey about faults of robots used in robocup. In: Robot Soccer World Cup, pp 344–355. Springer
8. Fong T, Nourbakhsh I, Dautenhahn K (2003) A survey of socially interactive robots. Robot Auton Syst 42(3–4):143–166
9. Phillips E, Ososky S, Grove J, Jentsch F (2011) From tools to teammates: toward the development of appropriate mental models for intelligent robots. In: Proceedings of the human factors and ergonomics society annual meeting, vol 55, pp 1491–1495 . SAGE Publications Sage CA: Los Angeles, CA
10. Staggers N, Norcio AF (1993) Mental models: concepts for human-computer interaction research. Int J Man Mach Stud 38(4):587–605
11. Nass C, Steuer J, Tauber ER (1994) Computers are social actors. In: Proceedings of the SIGCHI conference on human factors in computing systems, pp 72–78
12. Banks J (2020) Optimus primed: media cultivation of robot mental models and social judgments. Front Robot AI 7:62
13. Kriz S, Ferro TD, Damera P, Porter JR (2010) Fictional robots as a data source in hri research: exploring the link between science fiction and interactional expectations. In: 19th International symposium in robot and human interactive communication, pp 458–463. IEEE
14. Norman DA (1981) Categorization of action slips. Psychol Rev 88(1):1
15. Rosenfeld A, Richardson A (2019) Explainability in human-agent systems. Auton Agent Multi-Agent Syst 33:673–705
16. Clinciu MA, Hastie HF (2019) A survey of explainable ai terminology. In: 1st Workshop on interactive natural language technology for explainable artificial intelligence 2019, pp 8–13. Association for Computational Linguistics
17. Bhaskara A, Skinner M, Loft S (2020) Agent transparency: a review of current theory and evidence. IEEE Trans Hum-Mach Syst 50(3):215–224
18. Tabrez A, Luebbers MB, Hayes B (2020) A survey of mental modeling techniques in human-robot teaming. Curr Robot Rep 1:259–267
19. Miller T (2019) Explanation in artificial intelligence: insights from the social sciences. Artif Intell 267:1–38
20. Hellström T, Bensch S (2018) Understandable robots-what, why, and how. Paladyn, J Behav Robot 9(1):110–123
21. Dragan AD, Lee KC, Srinivasa SS (2013) Legibility and predictability of robot motion. In: 2013 8th ACM/IEEE international conference on human-robot interaction (HRI), pp 301–308 . IEEE
22. Broekens J, Harbers M, Hindriks K, Van Den Bosch K, Jonker C, Meyer J-J (2010) Do you get it? user-evaluated explainable bdi agents. In: Multiagent system technologies: 8th German conference, MATES 2010, Leipzig, Germany, September 27-29, 2010. Proceedings 8, pp 28–39. Springer
23. Kaptein F, Broekens J, Hindriks K, Neerincx M (2017) Personalised self-explanation by robots: the role of goals versus beliefs in robot-action explanation for children and adults. In: 2017 26th IEEE international symposium on robot and human interactive communication (RO-MAN), pp 676–682. IEEE
24. Koo J, Kwac J, Ju W, Steinert M, Leifer L, Nass C (2015) Why did my car just do that? explaining semi-autonomous driving actions to improve driver understanding, trust, and performance. Int J Interact Des Manuf (IJIDeM) 9:269–275
25. St. Clair A, Mataric M (2015) How robot verbal feedback can improve team performance in human-robot task collaborations. In: Proceedings of the tenth annual ACM/IEEE international conference on human-robot interaction, pp 213–220
26. Gong Z, Zhang Y (2018) Behavior explanation as intention signaling in human-robot teaming. In: 2018 27th IEEE international symposium on robot and human interactive communication (RO-MAN), pp 1005–1011. IEEE
27. Hayes B, Shah JA (2017) Improving robot controller transparency through autonomous policy explanation. In: Proceedings of the 2017 ACM/IEEE international conference on human-robot interaction, pp 303–312
28. Kulesza T, Stumpf S, Burnett M, Yang S, Kwan I, Wong W-K (2013) Too much, too little, or just right? ways explanations impact end users' mental models. In: 2013 IEEE symposium on visual languages and human centric computing, pp 3–10. IEEE
29. Breazeal C, Kidd CD, Thomaz AL, Hoffman G, Berlin M (2005) Effects of nonverbal communication on efficiency and robustness in human-robot teamwork. In: 2005 IEEE/RSJ international conference on intelligent robots and systems, pp 708–713. IEEE
30. De Greeff J, Belpaeme T (2015) Why robots should be social: enhancing machine learning through social human-robot interaction. PLoS ONE 10(9):0138061
31. Thomaz AL, Breazeal C (2006) Transparency and socially guided machine learning. In: 5th international conference on development and learning (ICDL)
32. Admoni H, Hayes B, Feil-Seifer D, Ullman D, Scassellati B (2013) Are you looking at me? perception of robot attention is mediated by gaze type and group size. In: 2013 8th ACM/IEEE international conference on human-robot interaction (HRI), pp 389–395 . IEEE
33. Takayama L, Dooley D, Ju W (2011) Expressing thought: improving robot readability with animation principles. In: Proceedings of the 6th international conference on human-robot interaction, pp 69–76
34. Beetz M, Stulp F, Esden-Tempski P, Fedrizzi A, Klank U, Kresse I, Maldonado A, Ruiz F (2010) Generality and legibility in mobile manipulation: learning skills for routine tasks. Auton Robot 28:21–44
35. Jarrassé N, Paik J, Pasqui V, Morel G (2008) How can human motion prediction increase transparency? In: 2008 IEEE international conference on robotics and automation, pp 2134–2139. IEEE
36. Huang SH, Held D, Abbeel P, Dragan AD (2019) Enabling robots to communicate their objectives. Auton Robot 43(2):309–326
37. Dragan AD, Bauman S, Forlizzi J, Srinivasa SS (2015) Effects of robot motion on human-robot collaboration. In: Proceedings of the tenth annual ACM/IEEE international conference on human-robot interaction, pp 51–58
38. Kwon M, Huang SH, Dragan AD (2018) Expressing robot incapability. In: Proceedings of the 2018 ACM/IEEE international conference on human-robot interaction, pp 87–95
39. Zhou A, Hadfield-Menell D, Nagabandi A, Dragan AD (2017) Expressive robot motion timing. In: Proceedings of the 2017 ACM/IEEE international conference on human-robot interaction, pp 22–31
40. Chao C, Cakmak M, Thomaz AL (2010) Transparent active learning for robots. In: 2010 5th ACM/IEEE international conference on human-robot interaction (HRI), pp 317–324 . IEEE
41. Duffy BR (2003) Anthropomorphism and the social robot. Robot Auton Syst 42(3–4):177–190
42. Hindemith L, Göpfert JP, Wiebel-Herboth CB, Wrede B, Vollmer A-L (2021) Why robots should be technical: correcting mental models through technical architecture concepts. Interact Stud 22(2):244–279

43. Wortham RH, Theodorou A, Bryson JJ (2017) Robot transparency: Improving understanding of intelligent behaviour for designers and users. In: towards autonomous robotic systems: 18th annual conference, TAROS 2017, Guildford, UK, July 19–21, 2017, Proceedings 18, pp 274–289 . Springer
44. Chakraborti T, Fadnis KP, Talamadupula K, Dholakia M, Srivastava B, Kephart JO, Bellamy RK (2017) Visualizations for an explainable planning agent. arXiv preprint arXiv:1709.04517
45. Hastie H, Chiyah Garcia FJ, Robb DA, Laskov A, Patron P (2018) Miriam: A multimodal interface for explaining the reasoning behind actions of remote autonomous systems. In: Proceedings of the 20th ACM international conference on multimodal interaction, pp 557–558
46. Matthews M, Chowdhary G, Kieson E (2017) Intent communication between autonomous vehicles and pedestrians. arXiv preprint arXiv:1708.07123
47. Chadalavada RT, Andreasson H, Krug R, Lilienthal AJ (2015) That's on my mind! robot to human intention communication through on-board projection on shared floor space. In: 2015 European conference on mobile robots (ECMR), pp 1–6. IEEE
48. Andersen RS, Madsen O, Moeslund TB, Amor HB (2016) Projecting robot intentions into human environments. In: 2016 25th IEEE international symposium on robot and human interactive communication (RO-MAN), pp 294–301. IEEE
49. Rosen E, Whitney D, Phillips E, Chien G, Tompkin J, Konidaris G, Tellex S (2020) Communicating robot arm motion intent through mixed reality head-mounted displays. In: Robotics research: the 18th international symposium ISRR, pp 301–316. Springer
50. Ruffaldi E, Brizzi F, Tecchia F, Bacinelli S (2016) Third point of view augmented reality for robot intentions visualization. In: Augmented reality, virtual reality, and computer graphics: third international conference, AVR 2016, Lecce, Italy, June 15-18, 2016. Proceedings, Part I 3, pp 471–478. Springer
51. Walker M, Hedayati H, Lee J, Szafir D (2018) Communicating robot motion intent with augmented reality. In: Proceedings of the 2018 ACM/IEEE international conference on human-robot interaction, pp 316–324
52. Rotsidis A, Theodorou A, Bryson JJ, Wortham RH (2019) Improving robot transparency: An investigation with mobile augmented reality. In: 2019 28th IEEE international conference on robot and human interactive communication (RO-MAN), pp 1–8 . IEEE
53. Dyck L, Beierling H, Helmert R, Vollmer A-L (2023) Technical transparency for robot navigation through ar visualizations. In: Companion of the 2023 ACM/IEEE international conference on human-robot interaction, pp 720–724
54. Wang C, Belardinelli A, Hasler S, Stouraitis T, Tanneberg D, Gienger M (2023) Explainable human-robot training and cooperation with augmented reality. In: Extended abstracts of the 2023 CHI conference on human factors in computing systems, pp 1–5
55. Baraka K, Paiva A, Veloso M (2016) Expressive lights for revealing mobile service robot state. In: Robot 2015: second iberian robotics conference: advances in robotics, vol. 1, pp 107–119. Springer
56. Baraka K, Veloso MM (2018) Mobile service robot state revealing through expressive lights: formalism, design, and evaluation. Int J Soc Robot 10:65–92
57. Ciocirlan S-D, Agrigoroaie R, Tapus A (2019) Human-robot team: effects of communication in analyzing trust. In: 2019 28th IEEE international conference on robot and human interactive communication (RO-MAN), pp 1–7 . IEEE
58. Rohlfing KJ, Wrede B, Vollmer A-L, Oudeyer P-Y (2016) An alternative to mapping a word onto a concept in language acquisition: pragmatic frames. Front Psychol. https://doi.org/10.3389/fpsyg.2016.00470
59. Bruner J (1985) Child's talk: Learning to use language. Child Lang Teach Ther 1(1):111–114
60. Vollmer A-L, Wrede B, Rohlfing KJ, Oudeyer P-Y (2016) Pragmatic frames for teaching and learning in human-robot interaction: review and challenges. Front Neurorobot 10:10
61. Coronado E, Mastrogiovanni F, Indurkhya B, Venture G (2020) Visual programming environments for end-user development of intelligent and social robots, a systematic review. J Comput Lang 58:100970. https://doi.org/10.1016/j.cola.2020.100970
62. Paternò F, Wulf V (2017) New perspectives in end-user development
63. Huang J, Cakmak M (2017) Code3: a system for end-to-end programming of mobile manipulator robots for novices and experts. In: 2017 12th ACM/IEEE international conference on human-robot interaction (HRI), pp 453–462. IEEE
64. Pot E, Monceaux J, Gelin R, Maisonnier B (2009) Choregraphe: a graphical tool for humanoid robot programming. In: RO-MAN 2009 - The 18th IEEE international symposium on robot and human interactive communication, pp 46–51. https://doi.org/10.1109/ROMAN.2009.5326209
65. Liu X-WT (2005) An intuitive and flexible architecture for intelligent mobile robots. Master's thesis, University of Manitoba. Accessed: 2024-08-15. https://mspace.lib.umanitoba.ca/bitstreams/d52b3f6b-5698-47ac-9b32-993466769953/download
66. Orendt EM, Henrich D (2018) An architecture for intuitive programming and robust execution of industrial robot programs. In: Tagungsband des 3. Kongresses Montage Handhabung Industrieroboter, pp 31–39. Springer
67. Li M, Losey DP, Bohg J, Sadigh D (2020) Learning user-preferred mappings for intuitive robot control. In: 2020 IEEE/RSJ international conference on intelligent robots and systems (IROS), pp 10960–10967 . https://doi.org/10.1109/IROS45743.2020.9340909
68. Helenon F, Thiery S, Nyiri E, Gibaru O (2021) Cognitive architecture for intuitive and interactive task learning in industrial collaborative robotics. In: 2021 the 5th International conference on robotics, control and automation, pp 119–124
69. Billington D, Estivill-Castro V, Hexel R, Rock A (2009) Architecture for hybrid robotic behavior. In: Hybrid artificial intelligence systems: 4th international conference, HAIS 2009, Salamanca, Spain, June 10-12, 2009. Proceedings 4, pp 145–156 . Springer
70. Schneider FB (2005) The state machine approach: a tutorial. Fault-Toler Distrib Comput 448:18–41
71. Bohren J, Cousins S (2010) The smach high-level executive. IEEE Robot Autom Mag 17(4):18–20
72. Bryson JJ (2001) Intelligence by design: principles of modularity and coordination for engineering complex adaptive agents. PhD thesis, Massachusetts Institute of Technology
73. Colledanchise M, Ögren P (2017) Behavior trees in robotics and ai: an introduction. arXiv preprint arXiv:1709.00084
74. Erol K (1996) Hierarchical task network planning: formalization, analysis, and implementation. PhD thesis
75. Peterson JL (1977) Petri nets. ACM Comput Surv (CSUR) 9(3):223–252
76. Michaud F, Nicolescu M (2016) Behavior-based systems. In: Springer handbook of robotics, pp 307–328. Springer
77. Maes P, Brooks RA (1990) Learning to coordinate behaviors. In: AAAI 90:796–802
78. Kortenkamp D, Simmons R, Brugali D (2016) Robotic systems architectures and programming. Springer handbook of robotics, 283–306
79. Koenig N, Howard A(2004) 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 . IEEE
80. Van Rossum G, Drake Jr FL (1995) Python Reference Manual. Centrum voor Wiskunde en Informatica Amsterdam

81. Quigley M, Conley K, Gerkey B, Faust J, Foote T, Leibs J, Wheeler R, Ng AY (2009) Ros: an open-source robot operating system. In: ICRA workshop on open source software, vol 3, p. 5. Kobe, Japan
82. Saunders J, Syrdal DS, Koay KL, Burke N, Dautenhahn K (2015) "teach me-show me"-end-user personalization of a smart home and companion robot. IEEE Trans Hum-Mach Syst 46(1):27–40
83. Nau D, Cao Y, Lotem A, Munoz-Avila H (1999) Shop: Simple hierarchical ordered planner. In: Proceedings of the 16th international joint conference on artificial intelligence, vol 2, pp 968–973
84. Diprose J, MacDonald B, Hosking J, Plimmer B (2017) Designing an api at an appropriate abstraction level for programming social robot applications. J Vis Lang Comput 39:22–40
85. Flanagan D (2006) JavaScript: the definitive guide. O'Reilly Media, Inc
86. Franz M, Lopes CT, Huck G, Dong Y, Sumer O, Bader GD (2016) Cytoscape. js: a graph theory library for visualisation and analysis. Bioinformatics 32(2), 309–311
87. Reingold EM, Tilford JS (1981) Tidier drawings of trees. IEEE Trans Software Eng 2:223–228
88. Breslow N (1970) A generalized kruskal-wallis test for comparing k samples subject to unequal patterns of censorship. Biometrika 57(3):579–594
89. Franke T, Attig C, Wessel D (2019) A personal resource for technology interaction: development and validation of the affinity for technology interaction (ati) scale. Int J Hum-Comput Interact 35(6):456–467
90. Green P, Allen L, Astner K (1996) The word memory test: A user's guide to the oral and computer-administered forms, us version 1.1. Durham, NC: CogniSyst
91. Coucke A, Saade A, Ball A, Bluche T, Caulier A, Leroy D, Doumouro C, Gisselbrecht T, Caltagirone F, Lavril T et al (2018) Snips voice platform: an embedded spoken language understanding system for private-by-design voice interfaces, pp 12–16. arXiv preprint arXiv:1805.10190
92. Merkel D (2014) Docker: lightweight linux containers for consistent development and deployment. Linux J 2014(239):2
93. Bartneck C, Kulić D, Croft E, Zoghbi S (2009) Measurement instruments for the anthropomorphism, animacy, likeability, perceived intelligence, and perceived safety of robots. Int J Soc Robot 1(1):71–81
94. Bangor A, Kortum PT, Miller JT (2008) An empirical evaluation of the system usability scale. Intl J Hum-Comput Interact 24(6):574–594
95. Buschmeier H, Buhl HM, Kern F, Grimminger A, Beierling H, Fisher J, Groß A, Horwath I, Klowait N, Lazarov S et al (2023) Forms of understanding of xai-explanations. arXiv preprint arXiv:2311.08760
96. Roesler E, Manzey D, Onnasch L (2021) A meta-analysis on the effectiveness of anthropomorphism in human-robot interaction. Sci Robot 6(58):5425
97. Breazeal C, Dautenhahn K, Kanda T (2016) Social robotics. Springer handbook of robotics, 1935–1972
98. Lohan K, Gieselmann S, Vollmer A-L, Rohlfing K, Wrede B (2010) Does embodiment affect tutoring behavior. In: International conference on development and learning, vol 2010. IEEE Ann Arbor, MI, USA
99. Zolotas M, Demiris Y (2019) Towards explainable shared control using augmented reality. In: 2019 IEEE/RSJ International conference on intelligent robots and systems (IROS), pp 3020–3026 . IEEE

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Lukas Hindemith was a PhD student at Bielefeld University, working with the Medical Assistance Systems Group at the time this research was conducted. He now is a Research Engineer at STILL GmbH in Hamburg, Germany, where he focuses on intelligent automation and robotics solutions for industrial logistics.

Christiane B. Wiebel-Herboth is a Senior Scientist at Honda Research Institute Europe. She earned her Diploma in Psychology from Eberhard Karls University Tübingen in 2010 and completed her Ph.D. at Justus Liebig University Giessen in 2013, focusing on the visual perception of materials and their properties. Following a postdoctoral position at TU Berlin, she joined Honda Research Institute Europe in 2016. Dr. Wiebel-Herboth's research interests include enhancing human-machine interaction from a user-centric perspective and analyzing gaze behavior during active tasks.

Heiko Wersing is the Director of Science at Honda Research Institute Europe GmbH, based in Frankfurt, Germany. He earned his Ph.D. in 2001 from Bielefeld University, where he focused on neural networks and machine learning. Following his doctorate, Dr. Wersing has been instrumental in advancing research in neural networks, machine learning, object recognition, and incremental learning. In recognition of his contributions, he was appointed as an honorary professor at Bielefeld University's Faculty of Technology in 2017.

Britta Wrede is a Professor at Bielefeld University's Medical School OWL, leading the Medical Assistive Systems Group and representing the school on the scientific board of the Center for Cognitive Interaction Technology (CITEC). She earned her M.A. in 1999 and Ph.D. in 2002 from Bielefeld University. Following a DAAD fellowship at the International Computer Science Institute in Berkeley, she returned to Bielefeld, eventually heading the Applied Informatics Group in 2009. In 2019, she established the Medical Assistive Systems Group. Her research focuses on multimodal human-agent interaction, emphasizing social integration in assistive systems. Professor Wrede serves on the DFG Senate Committee for Graduate Schools.

Anna-Lisa Vollmer is a Professor at the Faculty of Medicine OWL at Bielefeld University, where she leads the Interactive Robotics in Medicine and Care research group. She holds a M.Sc. in Mathematics and a Ph.D. from Bielefeld University. Following her doctorate, she completed postdoctoral research as a Marie Curie Experienced Researcher at Plymouth University and at ENSTA Paris/INRIA Bordeaux. As a member of the Center for Cognitive Interaction Technology (CITEC), her research emphasizes co-development with stakeholders, explainable AI, and the deployment of low-code/no-code robot learning approaches to enable non-expert users-such as care professionals and patients-to interact effectively with assistive robotic systems.