Karlsruhe Reports in Informatics 2012,14
Edited by Karlsruhe Institute of Technology,
Faculty of Informatics
ISSN 2190-4782

Project Albert Service Experiments (PASE)

Sven R. Schmidt-Rohr, Gerhard Dirschl,
Fabian Romahn, Thorsten Mai, Laurenz Berger,
Tim Friedrich, Christian Wischnewski, Martin Seidel,
Jonas Stahl, Rüdiger Dillmann

2012

KIT – University of the State of Baden-Wuerttemberg and National
Research Center of the Helmholtz Association

Fakultät für **Informatik**

Technical Report

Project Albert Service Experiments (PASE)

Sven R. Schmidt-Rohr, Gerhard Dirschl, Fabian Romahn, Thorsten Mai, Laurenz Berger, Tim Friedrich, Christian Wischnewski, Martin Seidel, Jonas Stahl, Rüdiger Dillmann

Corresponding Author: Sven R. Schmidt-Rohr (srsr@ira.uka.de)

2012

Institute for Anthropomatics, Karlsruhe Institute of Technology (KIT)

1 Introduction

The document at hand presents experiment setups for learning and autonomous execution of selected service robot missions. Robots available for experiments, *Albert* and *Adero*, shown in Figure 1, incorporate the abilities of autonomous navigation, natural human-robot interaction and autonomous environment manipulation. In the focus of experiments is the most abstract level of autonomy, providing a robot with the ability to select abstract actions in a mission, based on a situation as perceived by the robot. In the given system, decision making is based on the framework of *partially observable Markov decision processes* (POMDPs), which enable to select approximately optimal action when facing uncertainty about action effects and the current state of the world [1]. Such a POMDP, schematically shown in Figure 2, is based on an abstract model of a service robot mission, with a discrete set of states S, the world can be in, a discrete set of actions A which can be chosen and executed and a discrete set of measurements M. Furthermore, the abstract model includes a stochastic action effect (transition) model T, a reward model R and an observation uncertainty model O. Based on such a model, a policy can be computed, which allows the robot to select an action, expected to maximize future rewards, in any given belief b of the current state of the world. Actual action selection takes place in the information processing system of the robot under the rational agent paradigm [2], shown in Figure 3. Each discrete time step, the robot collects measurements of aspects of the world using its sensors. Based on sensor data processing, an abstract belief b about the current state of the world is processed. Based on such a belief, reasoning queries a previously computed decision making policy and triggers execution of the selected action. That action has in turn an effect onto the physical world when performed by the actuators of the robot.

Figure 1: Service robots Albert (left) and Adero (right).

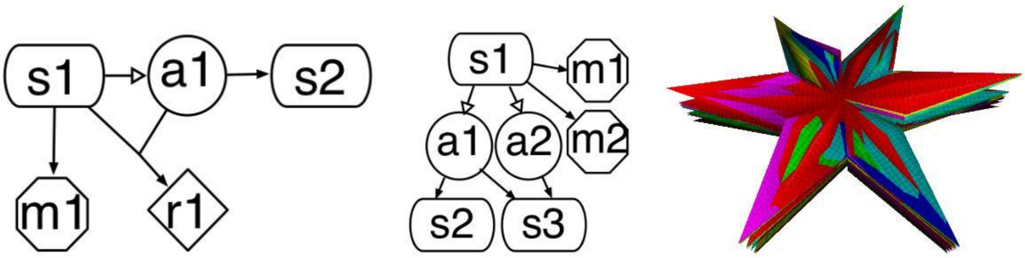

Figure 2: Schematic view of POMDP courses of events (left), a POMDP model (center) and as POMDP policy (right).

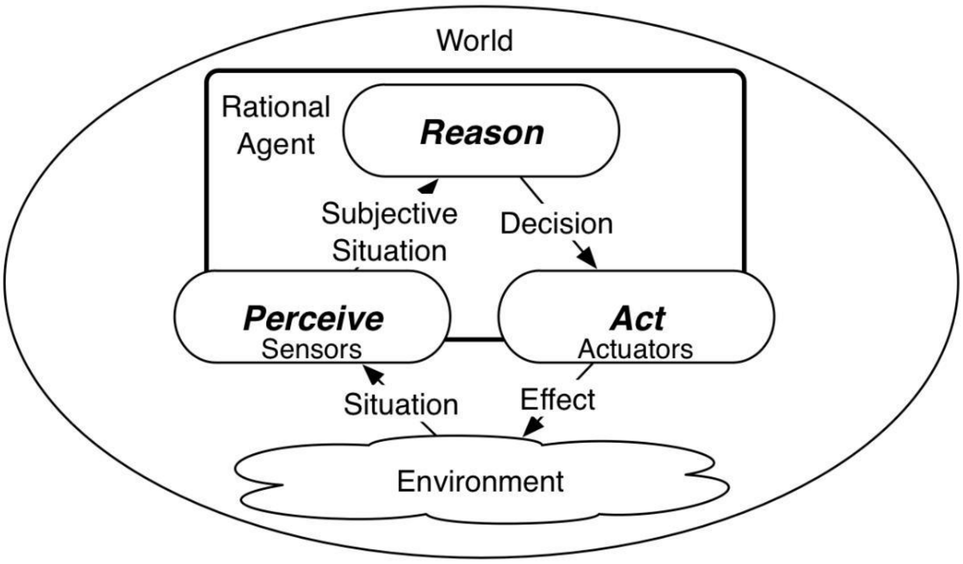

Figure 3: Schematic view of the rational agent paradigm.

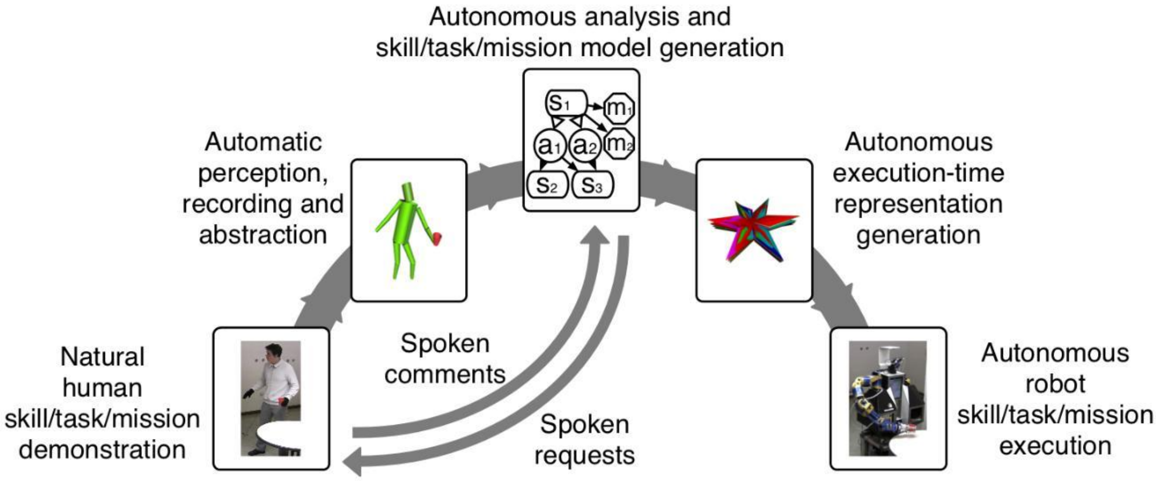

Figure 4: Schematic view of Programming by Demonstration from teacher observation (left) to autonomous robot execution (right).

In the given system, the following crucial questions are investigated:

- How to model a given, real service robot mission by means of an abstract POMDP model?
- How to comfortably acquire all parameters necessary for such a model?

The acquirement of model parameters, which can be interpreted as learning important knowledge concerning a service mission, is performed using learning based on analysis of natural human mission demonstrations in the given system. Such an approach, called *Programming by Demonstration* (PbD), is shown schematically in Figure 4. First, the robot captures the body movements of one or several human demonstrators (teachers) as well as optionally their speech and objects in the vicinity using its sensors and perception processing techniques. Based on a resulting, more abstract description of recorded courses of events of multiple sequences, an abstract model of the mission is generated. Finally, a policy is computed from such a model, which allows the robot autonomous execution of the demonstrated mission.

Before outlining the experimental setups and missions used for evaluation in Section 3, a brief overview on the execution-time information-processing system, providing autonomy and the system for generating decision making models from observation and analysis of human mission demonstrations is given in the next Section.

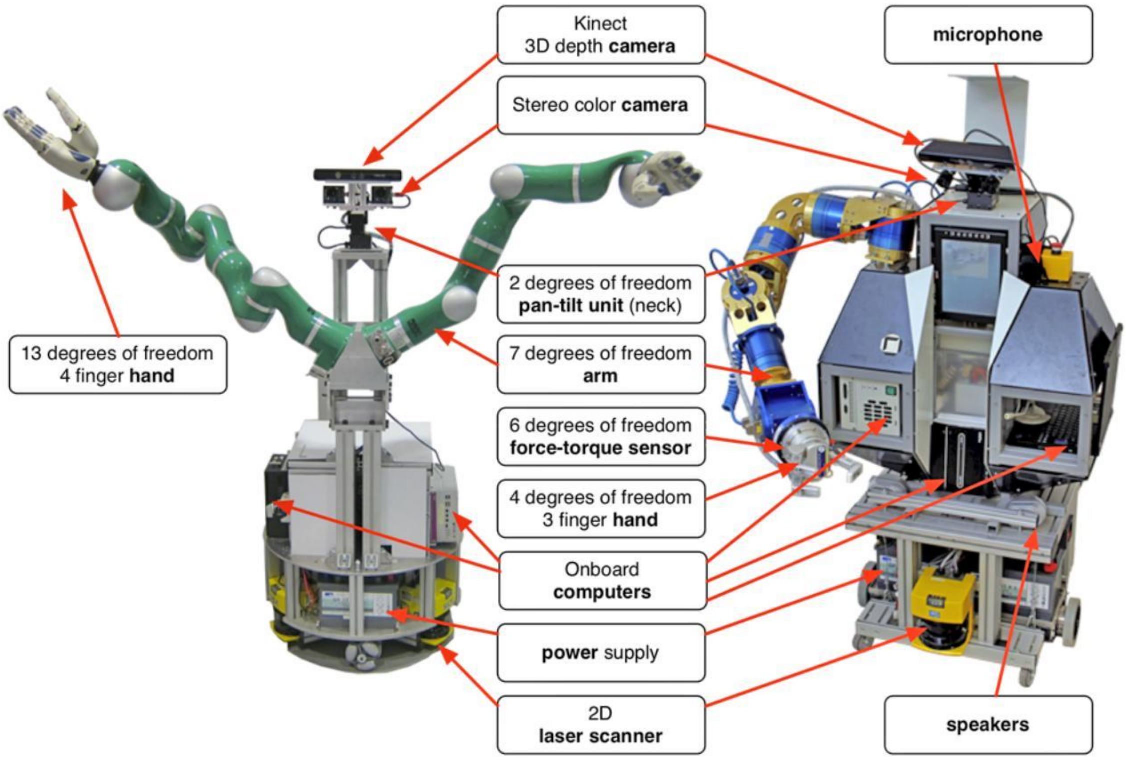

Figure 5: Service robots Adero (left) and Albert (right) with their main hardware components labeled.

2 System overview

Both execution-time and observation-time system are organized in a modular way as the robots are essentially a collection of skill components. These information-processing components typically relate to one or a few hardware components, which are labeled in Figure 5. By these means, skill domains, e.g. mobility, human-robot interaction and object manipulation are covered by several perception and actuation skill components [3].

For both learning and especially observation of human demonstrations, machine vision is crucial. During recording of human demonstrations, the robot can actively follow teachers with its head and track body movements and objects in the scene using stereo cameras and the Kinect RGB-D sensor, as shown in Figure 6.

However, there are some limitations for learning mission knowledge from human teachers. Some aspects, for instance the basic structure of a mission as well as typical human behavior in a mission, can be learnt from demonstrations while robot specific knowledge about its own perception and actuation peculiarities cannot be acquired from such demonstrations. A distinction of this kind can be made for example when analysing stochastic action effects, as shown in Figure 7. Primary action effect probabilities are independent of a particular executing agent and are thus valid for robot and human teacher in the same manner. Therefore, a robot can acquire them from observing human demonstrations. Secondary action effect probabilities, however, depend for instance on the motion planning technique, a robot uses or its arm/hand systems. For instance, relative frequencies of missed grasps for certain objects cannot be learnt from human demonstrations.

Based on this insight, the Programming by Demonstration approach used in the given sys-

Figure 6: Camera head of Albert for machine vision with the Kinect camera on top and a Guppy Stereo-camera setup below.

tem, as shown schematically in Figure 8, is organized in three stages: first, a preliminary model is generated from abstraction and analysis of recorded human mission demonstrations. Subsequently, missed aspects in demonstrated missions are estimated and demonstration sequences completed in an interactive process, including the teacher. Finally, model refinement stages, using description-logic, geometric analysis and learning from trials in physical dynamics simulation are applied to generate information about robot skill specific model aspects. By these means, model parameters, including both primary as well as secondary action effect probabilities, goal rewards and actions costs as well as information about observation uncertainty can be acquired.

This process is called *Probabilistic Mission Planning Model Programming by Demonstration* (PMPM-PbD).

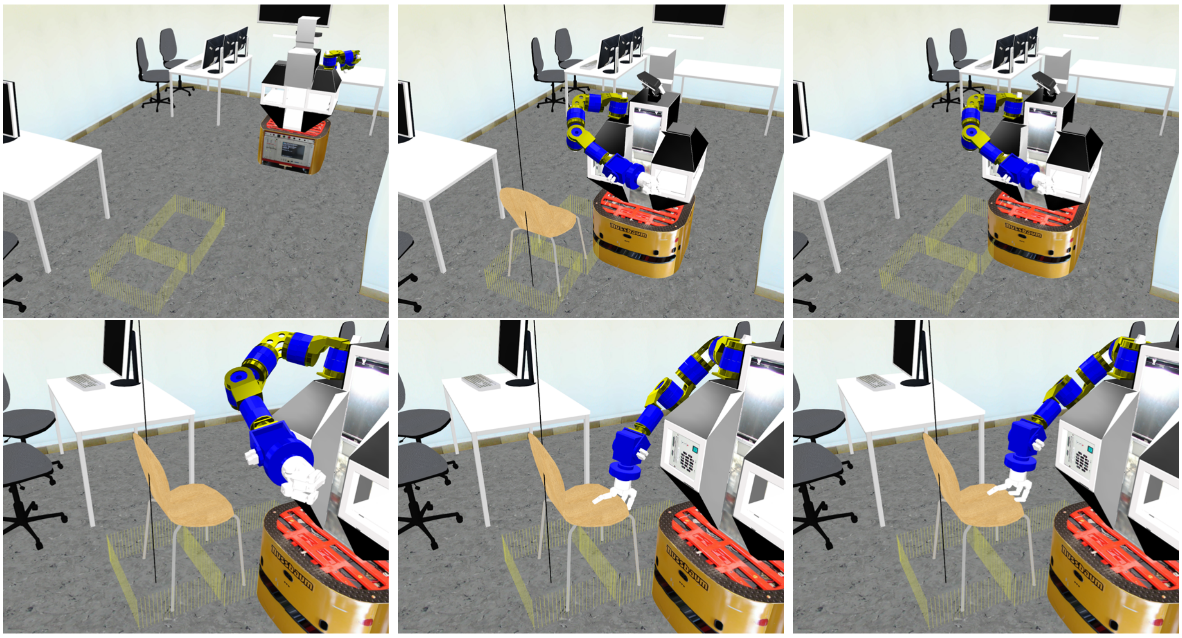

Figure 7: Primary action effects with a chair present or not (top) and secondary action effects with the robot grasping successfully or not (bottom).

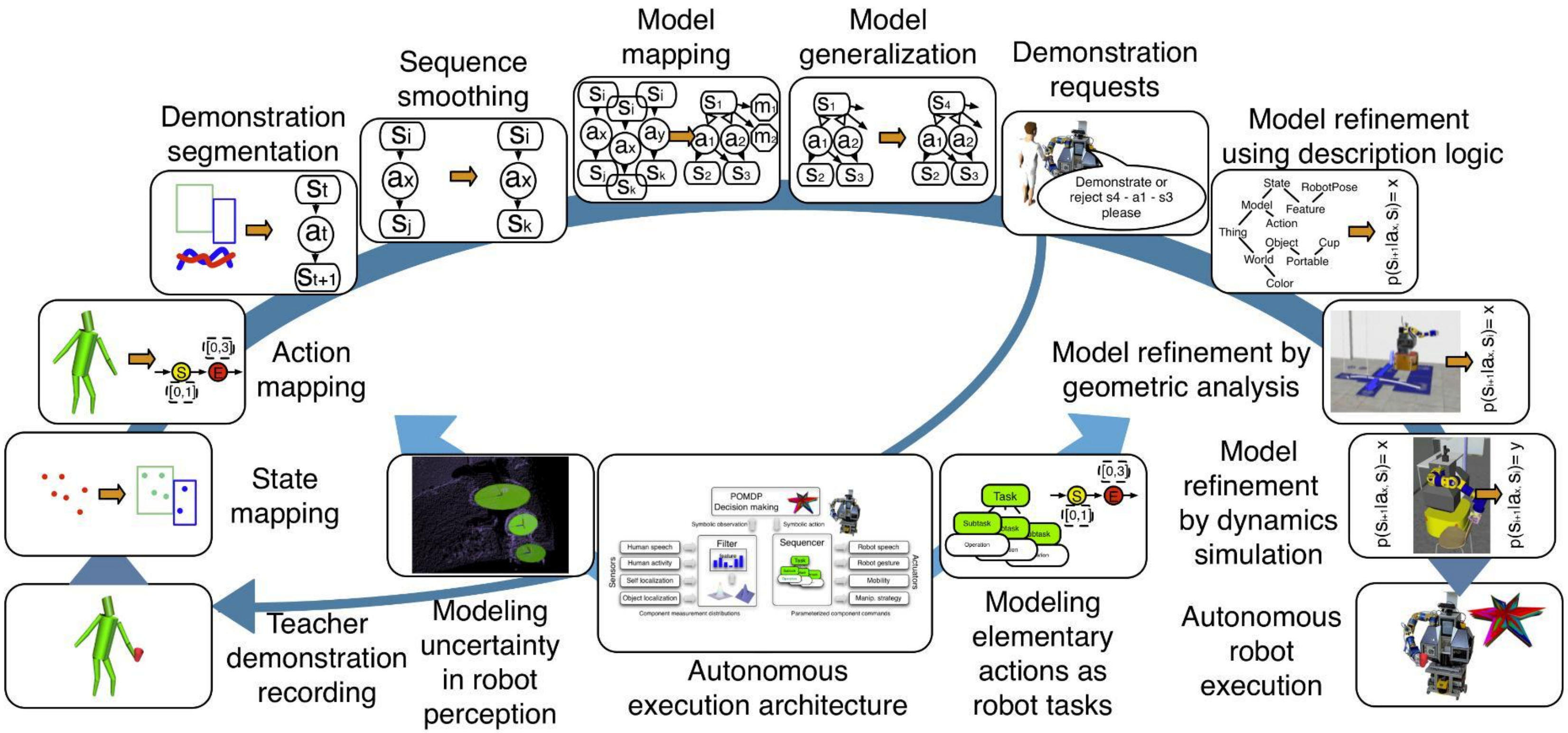

Figure 8: Schematic view of the PMPM-PbD process from recordings of human demonstrations (left) to autonomous robot execution policy (right).

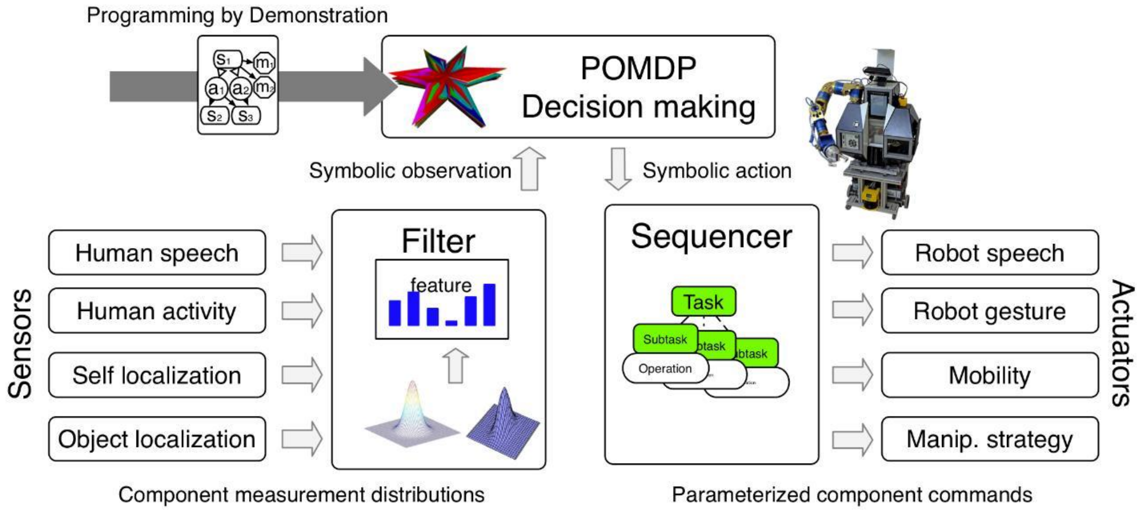

Figure 9: Schematic view of execution-time information flow with perception (left), autonomous decision making (top) and actuation commands (right).

2.1 Execution-time Architecture for Autonomy

The execution-time architecture is organized around the rational agent paradigm, as shown in Figure 3. Being a hierarchical, three layer architecture, as shown in Figure 9, basic perception skills process raw sensor signals and deliver refined, more abstract information to a filter component, which fuses this information into an abstract POMDP belief state b. Based on such a belief, a pre-computed policy is queried, selecting an action when the previous action has terminated. Such an abstract action is represented by a hierarchical task network (HTN), triggering one or several actuation skill commands in turn [4].

As the lowest layer in the execution architecture is represented by a collection of skills, these skills can run on several onboard computers and communicate by means of a computer network as shown in Figure 10.

The intermediate layer of the architecture is responsible for both mapping a certain, perceived world configuration onto an abstract state description as well as decomposition of an abstract action into actuator commands. World configurations are mapped onto abstract states by discretizing measurement values of individual perception skills along a concept, called *feature states* [4]. A feature f^i thereby covers a set of potentially continuous measurement values of one or several perception skill components which are mapped on a discrete, finite set of *feature states* c_i, as illustrated in Figure 11. Information processing from sensor data, followed by perception skill processing to the abstract feature discretization, as shown in Figure 12, can thus be interpreted as a channel which connects the subjective, abstract belief b of the robot with the real world. Such channels introduce both abstraction as well as additional observation uncertainties, that have to be considered in the observation model.

In the context of the given system, the following features are considered:

- Pose of the robot in the scene (*self-pose*).
- Pose of small objects, which can be carried, in the scene (*small-obj-state*).
- Pose of large objects in the scene (*furni-state*).

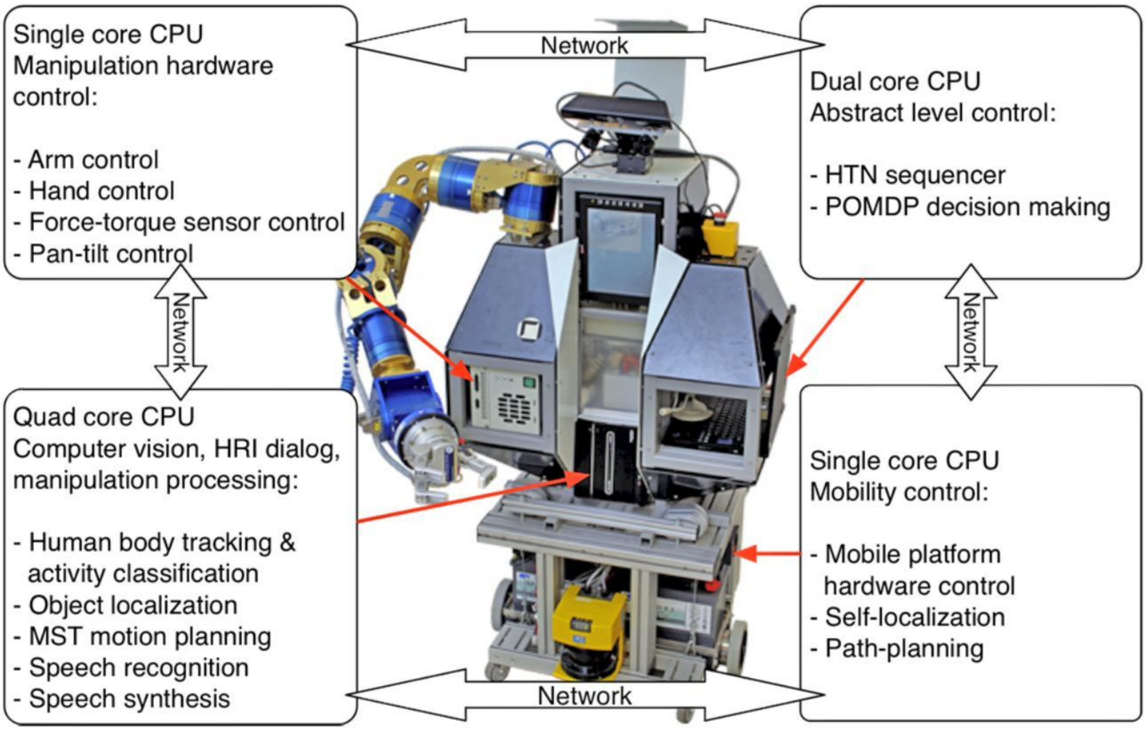

Figure 10: Description of allocation of processing components to different onboard computers on Albert.

- Pose of an interacting human in the scene (*human-pose*).
- Full body activity of an interacting human (*human-act*).
- Spoken utterance of an interacting human (*dialog-state*).

Exemplary discrete feature states are shown in Figure 13.

Abstract actions are decomposed into elementary actuation skill commands by means of a hierarchical task network called *Flexible Programs* (FPs) [5]. The root of the tree represents the abstract POMDP action and the tree is in turn executed by depth-first search as illustrated in Figure 14. Exemplary abstract action FPs are shown in Figure 15.

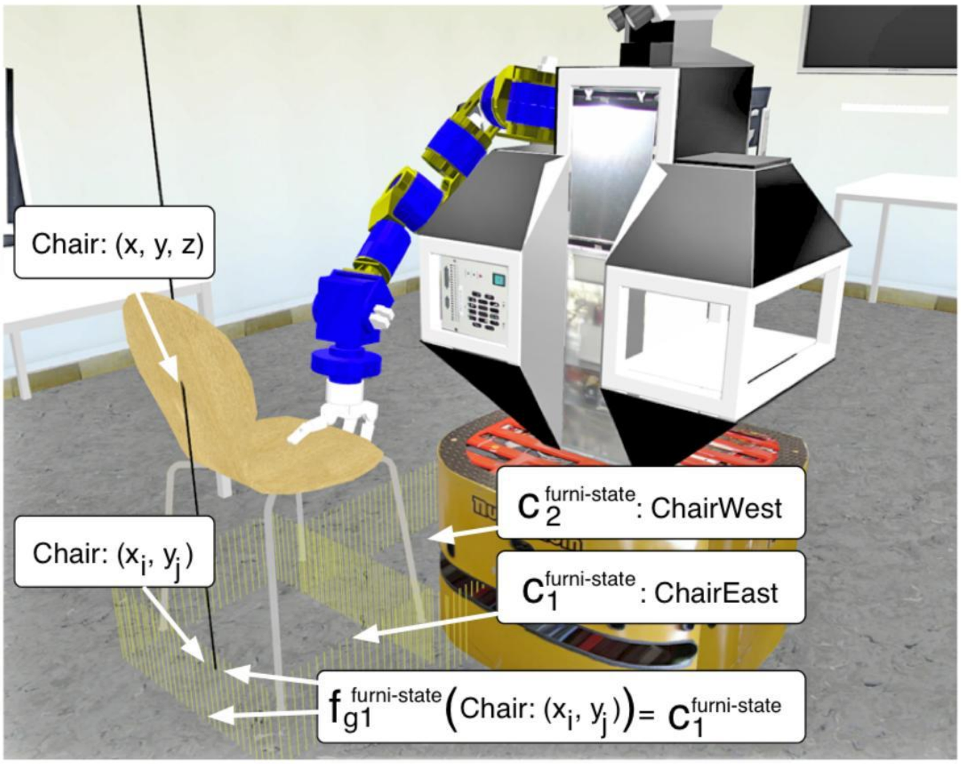

Figure 11: Illustration of the feature concept, in this instance based on the pose of the chair.

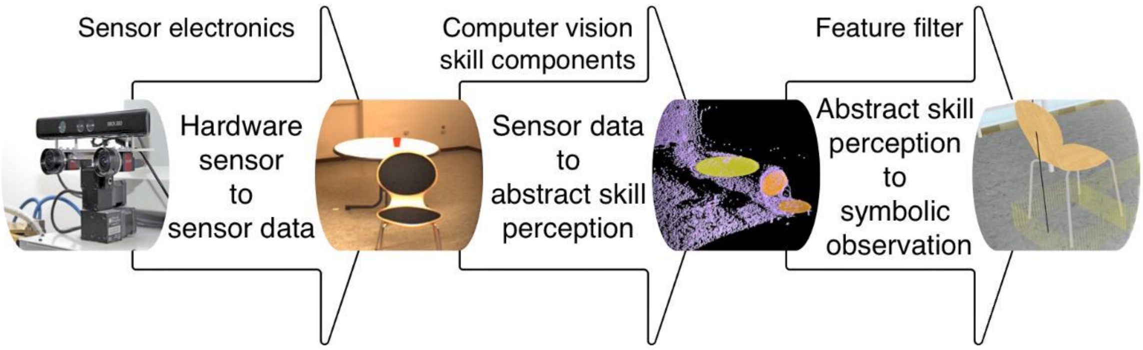

Figure 12: Schematic view of the information flow from the physical sensor to an abstract feature.

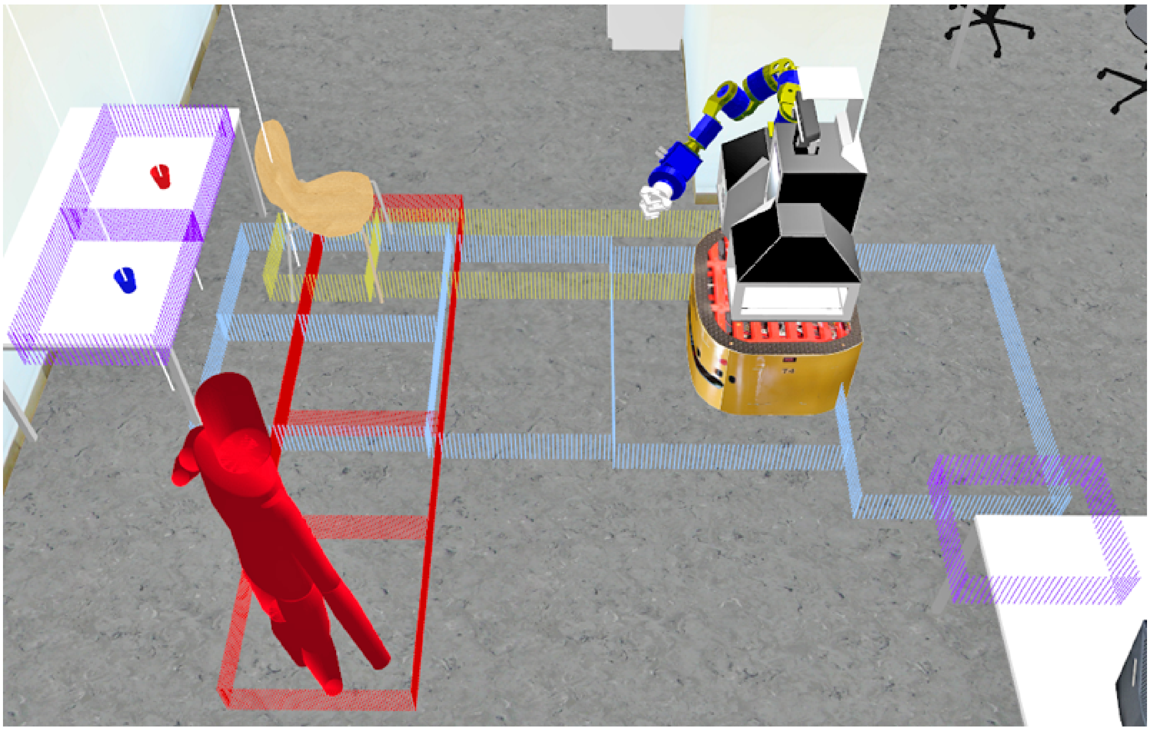

Figure 13: Illustration of exemplary categories for the features self-pose (blue), human-pose (red), small-obj-state (purple) and furni-state (yellow).

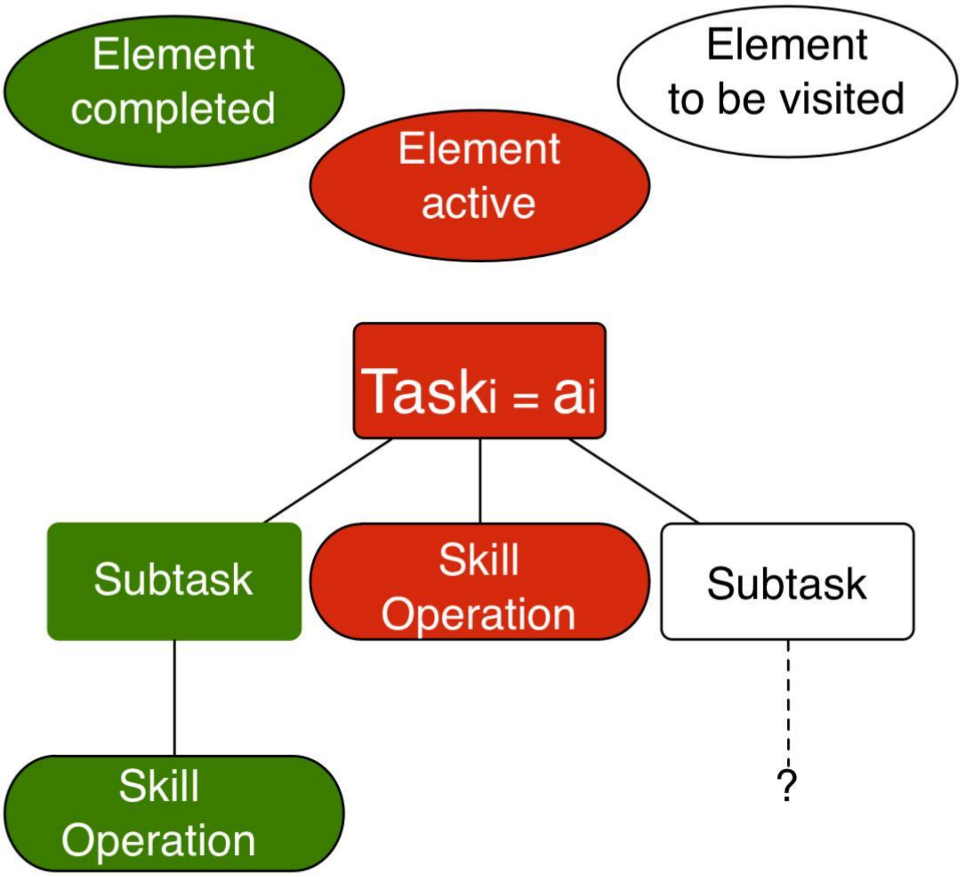

Figure 14: Schematic view of a Flexible Program.

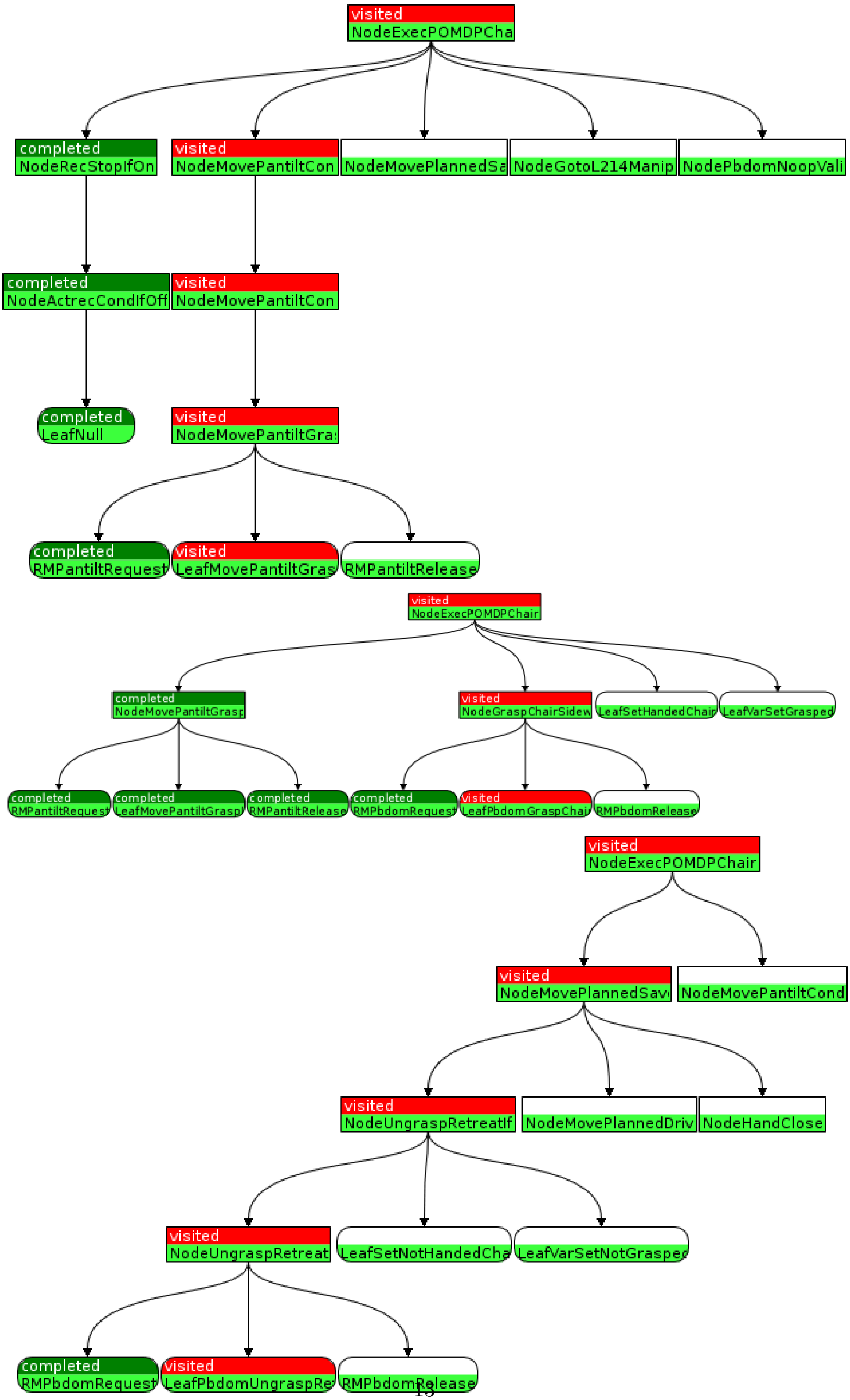

Figure 15: Exemplary Flexible Programs, representing elementary mission-POMDP actions: Goto (top), Grasp (center), Ungrasp (bottom).

2.2 Recording of Human Mission Demonstrations

To generate POMDP mission models usable for autonomous execution as described in Section 2.1, these models are generated from analysis of human teacher demonstrations of such missions followed by further refinement stages as sketched in Figure 8. The first stage in that process is recording human mission demonstrations by means of visual and acoustic observation by the robot. A crucial part are human full body motion tracking and symbolic full body motion classification [6], as shown in Figure 16. As the robot has only a limited field of view, it follows the human teacher, which represents the role of the robot when executing the mission, actively with its neck (pan-tilt unit). To focus on further interesting parts of the scene, for instance certain object poses, the human teacher may steer the robots attention further with special gestures as shown in Figure 17. Consequently, the same aspects of the scene as during autonomous execution, discussed in Section 2.1, can be recorded. However, human pose, human body activity and human utterances may be recorded for two interacting humans, one of them representing the role of the robot in the mission. Therefore recording-time features are the following [7]:

- Robot role human pose
- Robot role human full body activity type
- Robot role human utterance
- Interacting human pose
- Interacting human full body activity type
- Interacting human utterance
- Portable and non-portable object poses

Recorded data points can be stored as XML for each individual demonstration sequence and later be retrieved for further analysis stages.

2.3 Generation of State Definitions Based on Demonstrations

While perception skill components already deliver symbolic, discrete labels for human full body activity types and spoken utterances, human and object poses in the scene are recorded as continuous values. To generate feature discretization mappings autonomously from demonstrations, clustering methods are applied onto recorded data-points, as discussed in [8]. As a result, feature mappings are generated, which can map any recorded data point as well as any execution-time situation onto a discrete POMDP state.

2.4 Mapping Recorded Human Activities onto Robot-Executable Actions

Human full body activities as classified by the CHARM skill component [6] are not directly applicable as executable robot actions skill, which are represented by constraint-based motion planning *manipulation strategies* [9]. Therefore a mapping stage has to assign a suitable manipulation strategy to a given activity label in a certain recording situation. To achieve this, object-centered trajectory similarity analysis is performed as described in [10] and shown in Figure 30. As a result, recorded manipulation action symbols refer to robot-executable skill commands.

2.5 Segmentation and Preliminary Model Generation

After computing applicable situation discretization and action mapping is complete, recorded demonstrations can be segmented into symbolic, discrete state-action sequences [7]. Based on several segmented recordings, a preliminary (PO)MDP model of the mission can be generated as described in [7]. Such a model includes all courses of events present in demonstrations, according to their relative frequencies. However, courses of events, which were not demonstrated may be missing as well as robot-specific action effect probabilities. Hence, further processing stages are necessary.

Figure 16: Exemplary online human full body activity classification results for two pairs of human activities: grasping different types of objects with one hand (top) and grasping a table with two arms in two different ways (bottom).

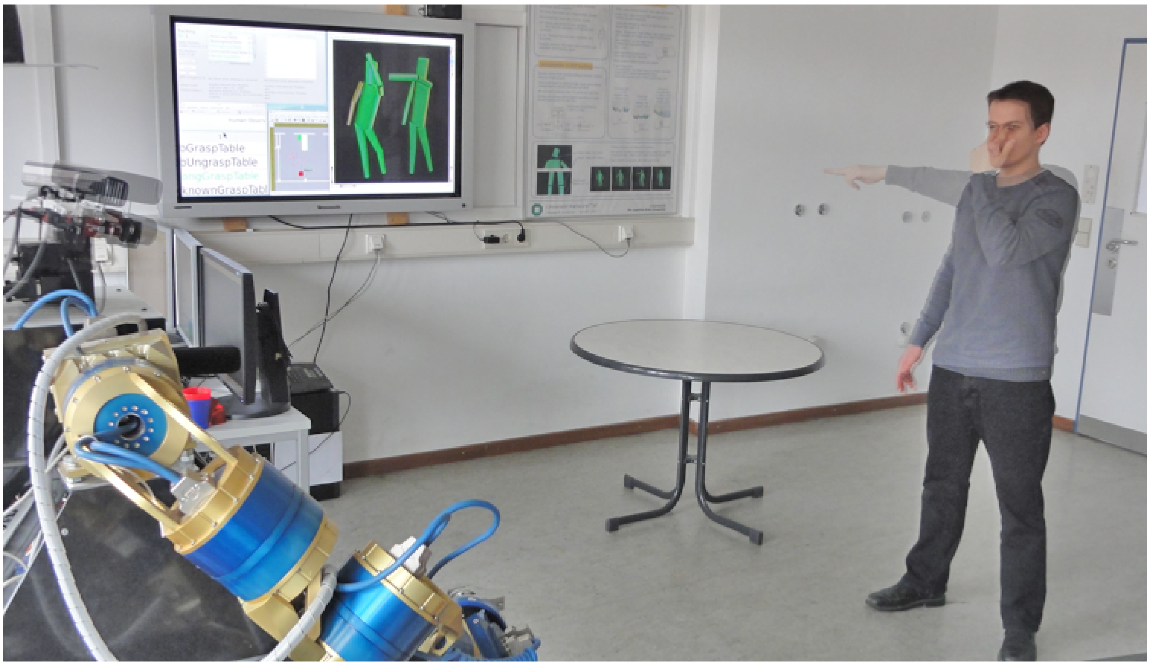

Figure 17: Illustration of the gesture-based attention mechanism.

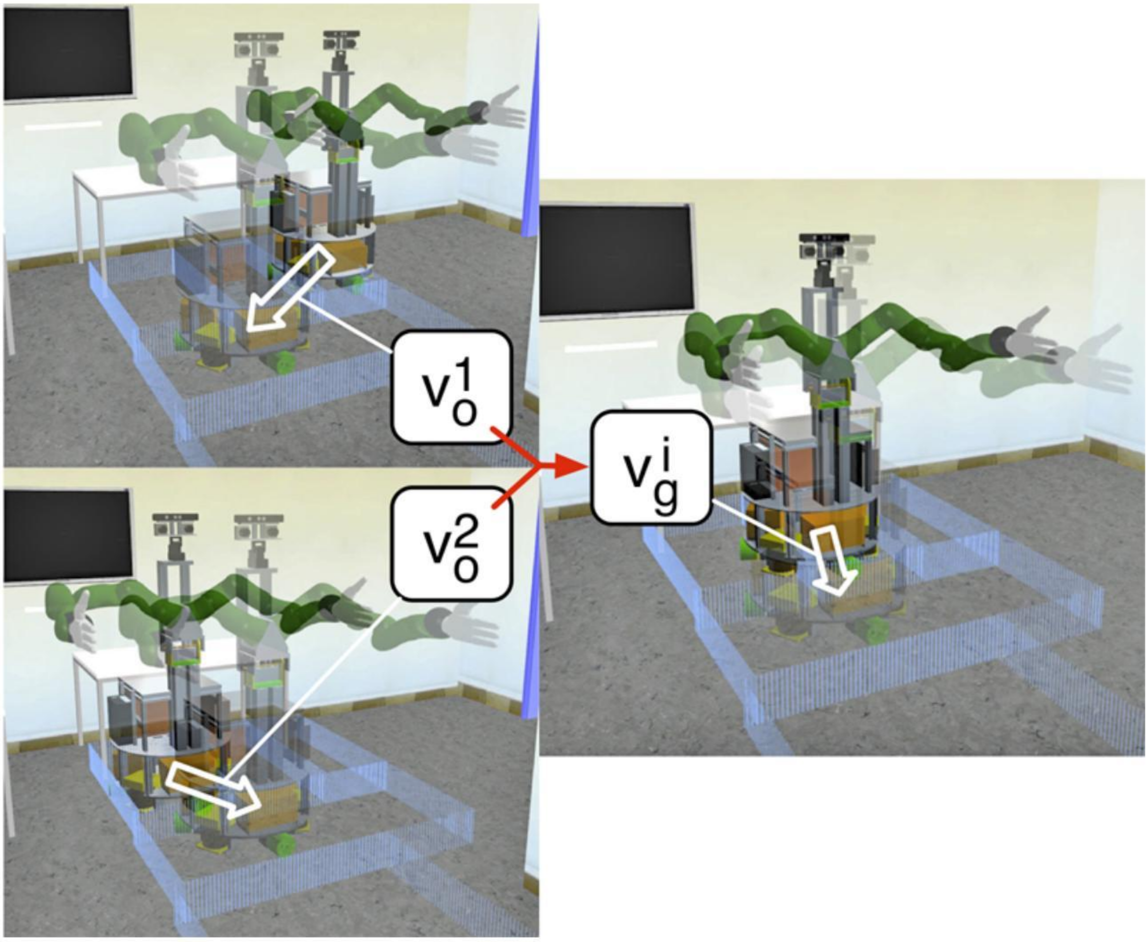

Figure 18: Illustration of transition generalization: two observed transitions (left) indicate a non-zero likelihood that a similar, not observed transition (right) is relevant.

2.6 Transition Model Generalization and Interactive Verification

To deal with demonstration bias and lacking information in demonstration sets in general, a model analysis and interactive verification stage is utilized as described in [11]. First, the transition model is analyzed for transitions which have not been observed in teacher demonstrations, but are very similar to other, observed transitions. Similarity is hereby defined by metrics in state and action spaces, concerning two transitions $v_1(s_1, a_1, s'_1)$ and $v_2(s_2, a_2, s'_2)$ as illustrated in Figure 18. Subsequently, models, attained by extending the original preliminary model with such similar, non-observed transitions are analyzed with respect to the impact of adding such non-observed transitions. Some transitions may allow very promising courses of events, not present in the original, preliminary model, for instance when the human teacher forgot to demonstrate them. However, transitions estimated by these means may as well be non-applicable or outright wrong considering the real setting. Thus, a verification stage is included, where human teachers are asked to perform further demonstrations, potentially including such courses of events. In turn, a teacher may either demonstrate such sequences, closing gaps in model knowledge, or decline a request as not applicable. To allow determination which transitions in a request are invalid, a binary request scheme, shown in Figure 19, is utilized.

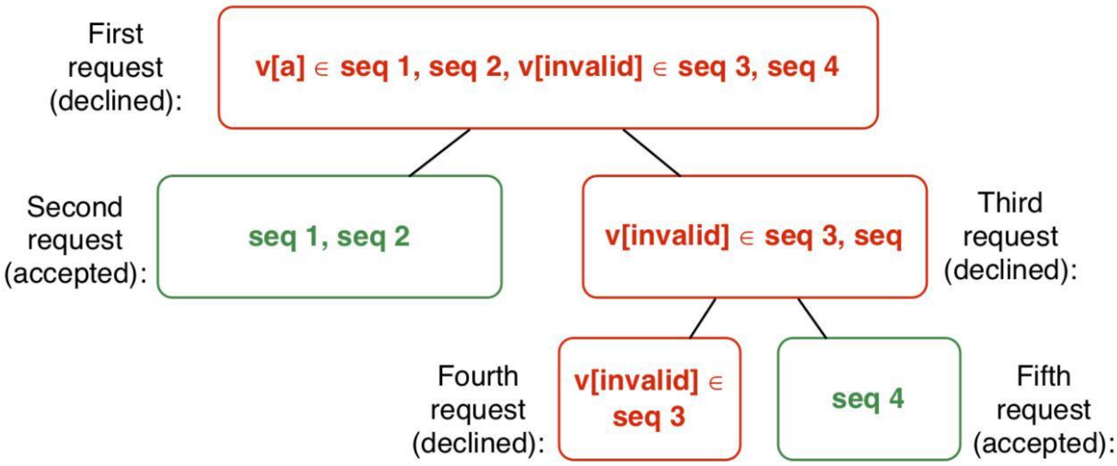

Figure 19: Schematic view of the binary-tree demonstration request concept.

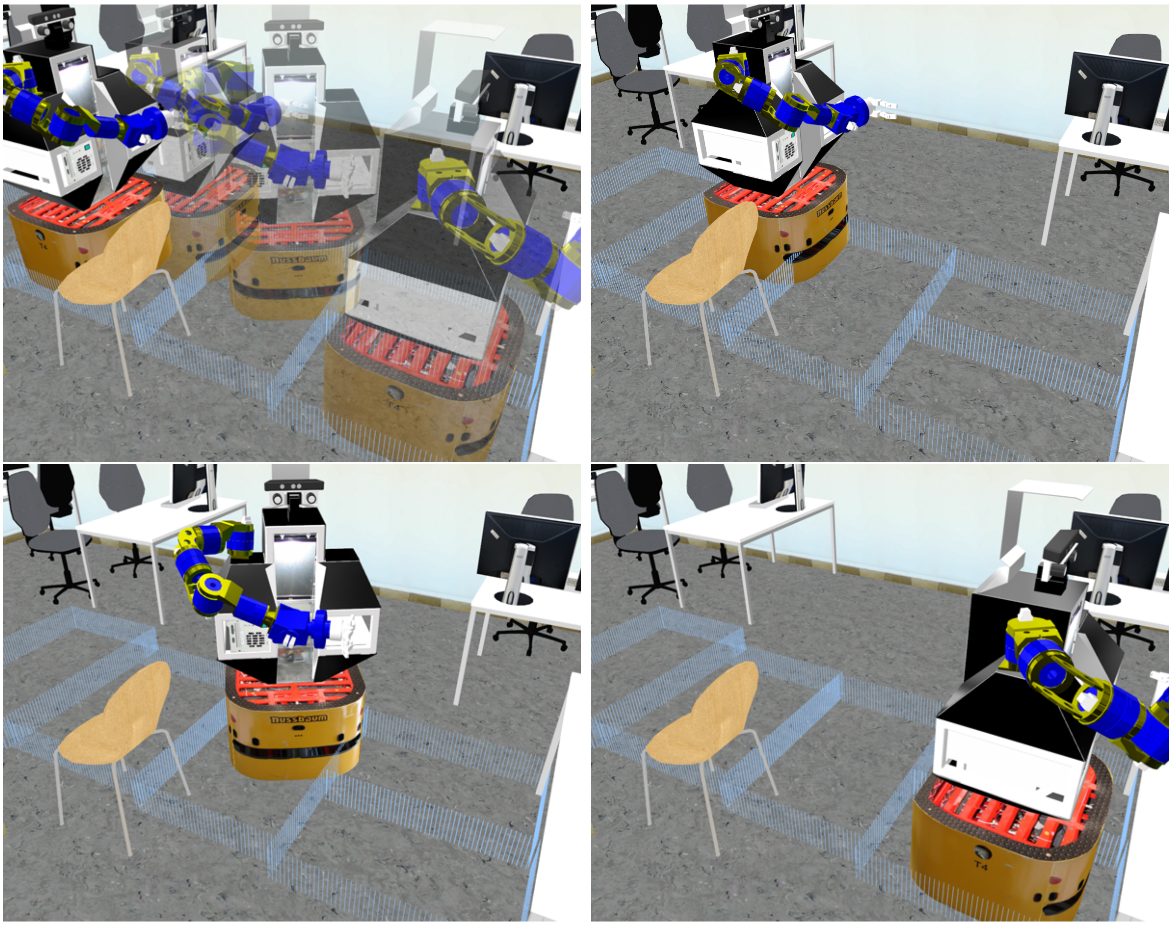

Figure 20: Illustration of action execution error transition with origin and potential result states (top left), error results (top right and bottom left) and the intended goal state of a goto action (bottom right).

2.7 Description Logic Based Inference of Additional Model Information

To extend the preliminary model with robot-specific mission information, a description-logic based inference framework is utilized [12]. For instance navigation errors may lead the robot to get stuck when executing a ”goto” action as illustrated in Figure 20. Such effects will never be demonstrated by human teachers, but may occur during autonomous execution and hence have to be considered by decision making. Further robot-specific information includes action effort costs in the reward model as illustrated in Figure 21 as well as observation uncertainty.

Model refinement is achieved by having persistent background knowledge in the form of a general scenario ontology as shown partially in Figure 22. Such an ontology contains general information about errors, efforts and robot skills in relation to known scenes, human behaviors and objects as abstract classes, enhanced by numeric information. After demonstration recording and preliminary model generation, new instances can be inferred in the ontology, based on analysis of teacher demonstrations and resulting model information added to the model.

To solve the problem of symbol correspondence between persistent background knowledge and entities generated for a specific mission demonstration, spoken teacher comments can be utilized as shown in Figure 23.

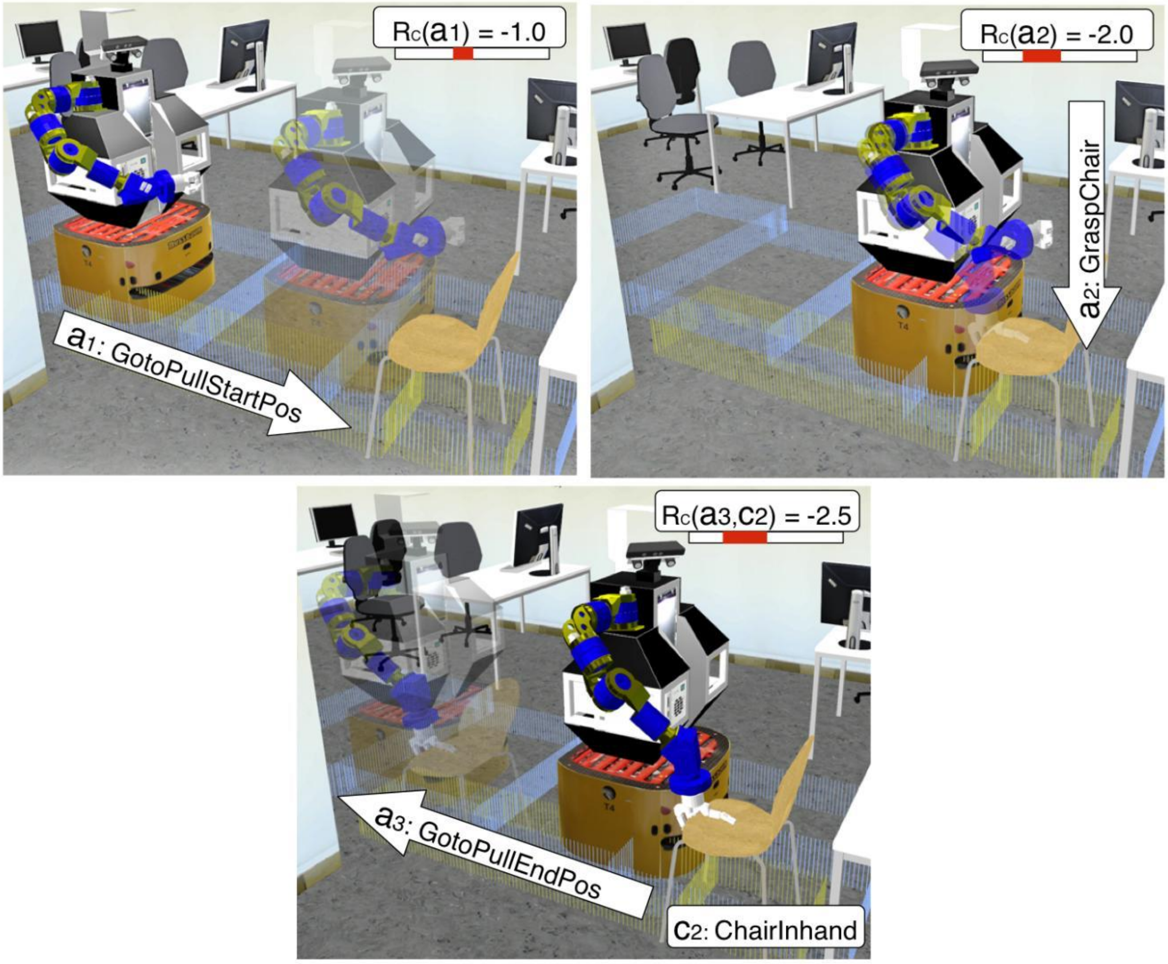

Figure 21: Illustration of action costs in the reward model with mobility (top left), manipulation (top right) and mobile manipulation (bottom).

2.8 Geometric Analysis of Navigation and Manipulation for Transition Model Refinement

Concerning transition probabilities, description-logic based inference of robot-specific action effect probabilities is often an imprecise estimation of real-world probabilities as it does not consider the specific scene geometry. In contrast, the scene is known from demonstrations. Hence, a virtual replication of demonstrated situations and action choices with instead of the human, the robot executing the mission can help to retrieve better estimates.

In the given process stage, both mobility navigation path planning and execution errors as well as manipulation action motion planning errors are considered and evaluated in a purely geometric fashion to retrieve better real-world approximation for the POMDP transition model [13].

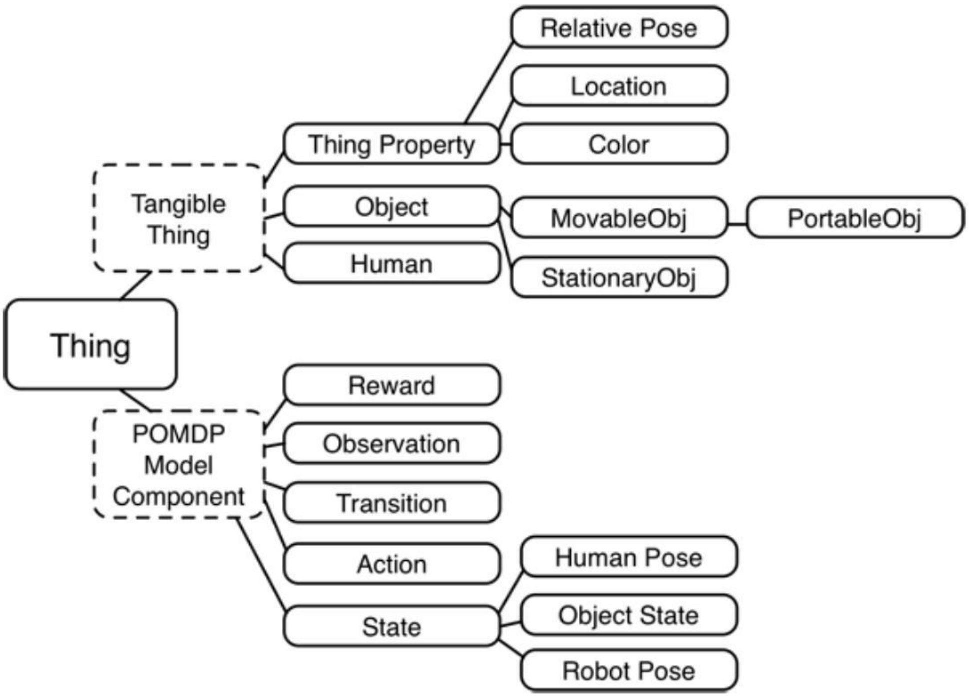

Figure 22: Schematic view of the upper portion of the knowledge base ontology.

Figure 23: Illustration of labeling states using spoken comments.

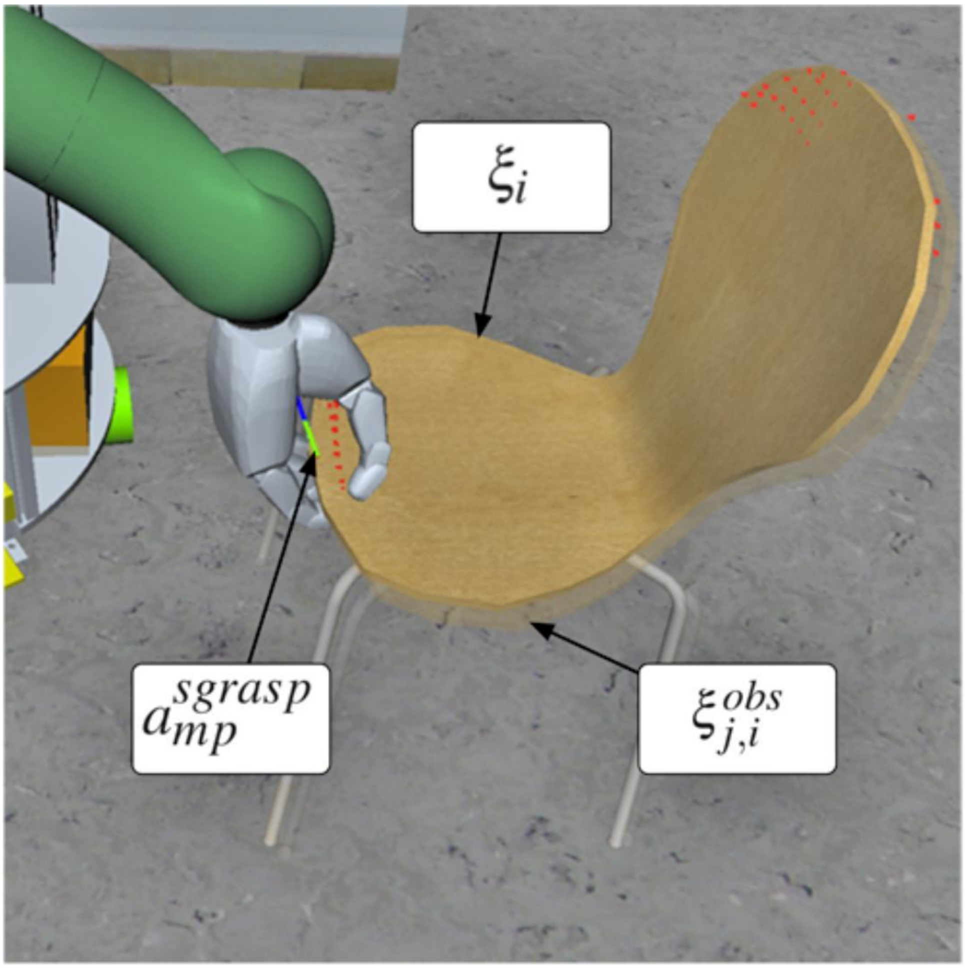

Figure 24: Illustration of observed world configuration $\xi_{j.i}^{obs}$ used for motion planning of action a_{mp}^{sgrasp} and intrinsic world configuration ξ_i, applied during simulated execution.

2.9 Transition Model Refinement Using Trials in Dynamics Simulation

While geometric analysis of mobility and manipulation motion planning is versatile, it cannot assess dynamic execution peculiarities, occurring when the robot interacts with physical objects as during object manipulation. Some manipulation strategies are more robust in the face of uncertainty than others when considering specific states and scenes they are executed in. To acquire information about realistic, robot-specific action effect probabilities in such cases, the robot has to learn them from experience. Such experience can be trials, replicating state-action pairs in real scenes in a manner similar to human teacher demonstrations. Effect frequencies can then be assessed, leading to good transition probability estimates. However, learning from real, physical trials is cumbersome, replicating initial states is difficult and precise effect state evaluation infeasible. Therefore, in the given system, learning from simulated trials in physical dynamics simulation is performed instead. As shown in Figure 24, observation uncertainty can also be considered with motion planning taking place on a virtually perceived situation while motion execution is then performed in a different, sampled scene configuration.

After execution of the motion in physical dynamics simulation, the resulting state can easily be evaluated, with special state configurations illustrated in Figure 25.

To replicate situations, relevant for demonstrated missions, trial configurations are sampled from feature descriptions [14], as shown in Figure 26.

After including these refined transition, a final POMDP mission can be used to compute an executable policy.

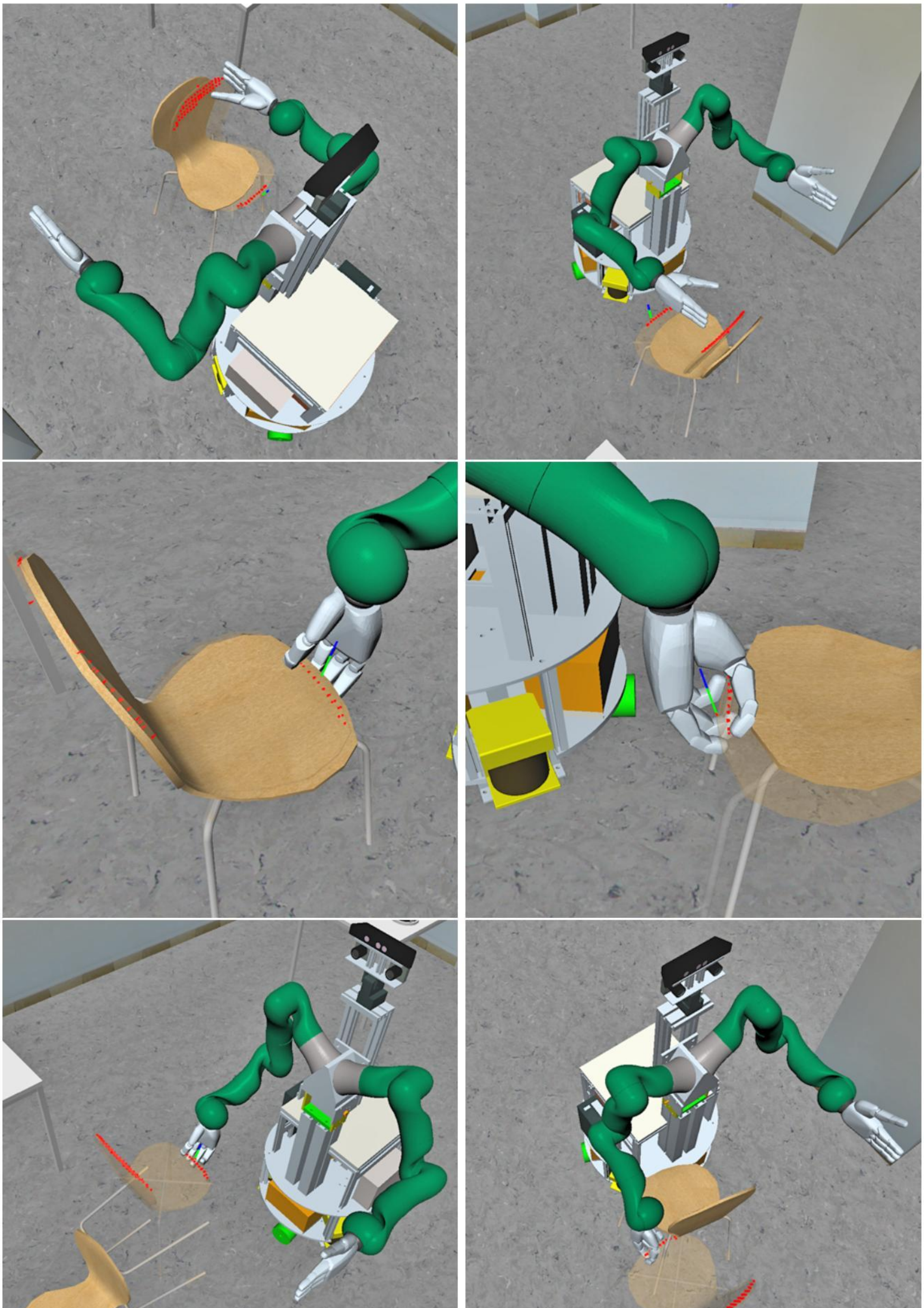

Figure 25: Illustration of states in learning from simulated trials: origin (top), success (center left), static scene (center right), chaos (bottom left) and jammed (bottom right).

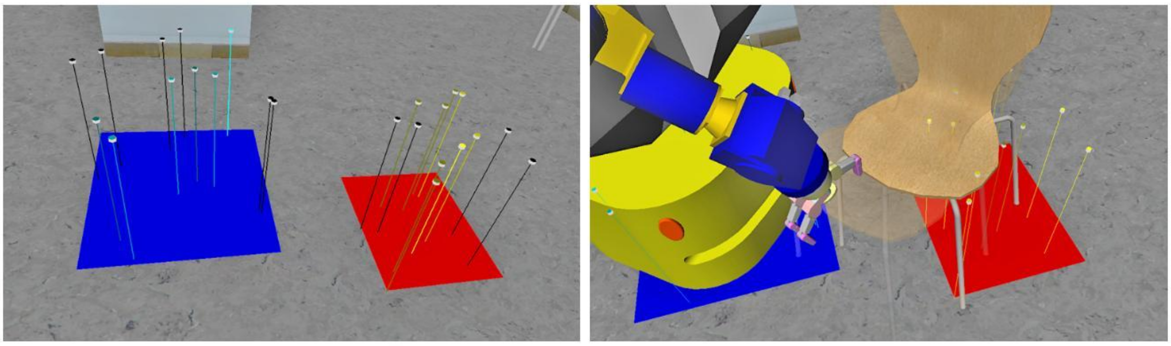

Figure 26: Illustration of configurations xi_i, sampled from state descriptions, generated from analysis of human demonstrations. The lighter the color of a sample, the more motions could be successfully planned.

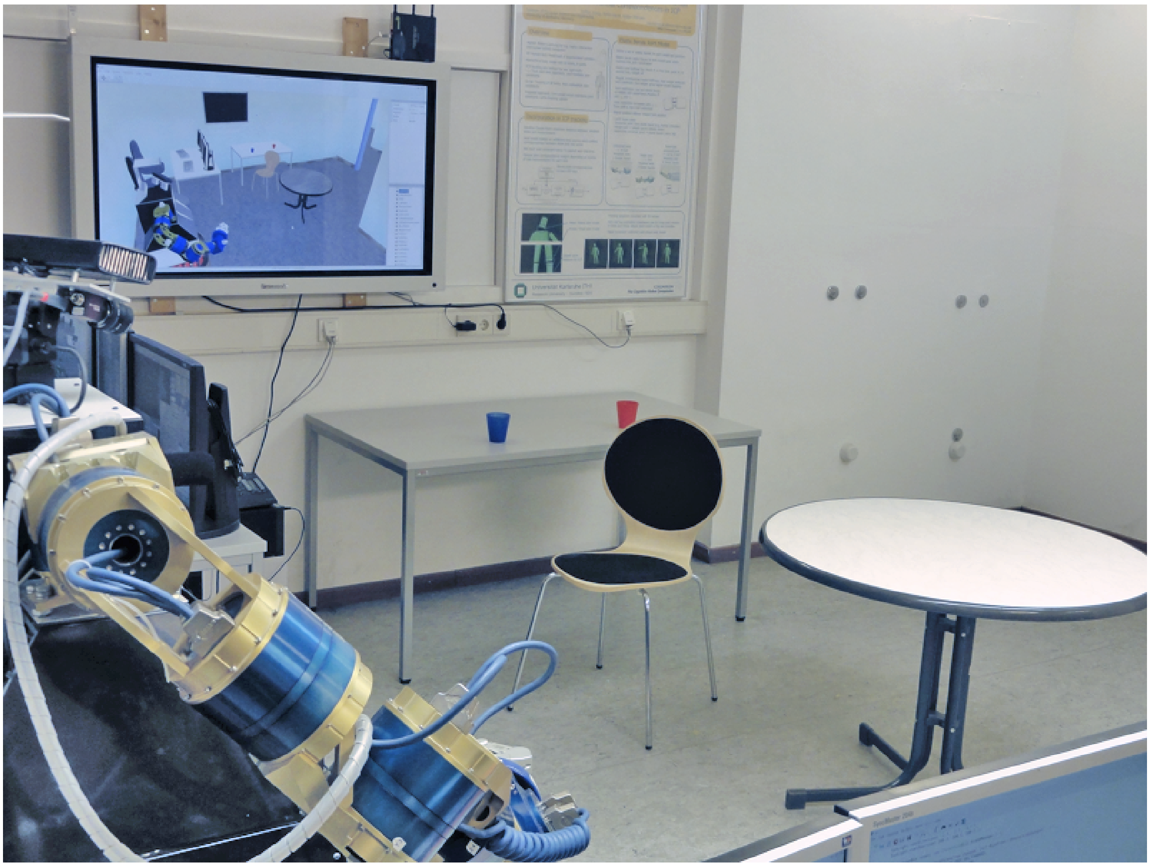

Figure 27: Picture of objects in a typical experiment scene, with visualization of an internal representation visualized on the screen in the background.

3 Experiments

The main purpose of this document is to present the setup for several experiments performed to evaluate performance aspects of the system presented in the previous Sections. In the following, first the general setup is discussed, followed by descriptions of missions used for experiments.

3.1 General Setting

For both robots Albert and Adero (see Figure 1), a dedicated laboratory environment was available for experiments. Each environment can simulate a small cafeteria setting with some open space for the robot to navigate around, some fixed furniture objects, some moveable furniture objects, small portable objects and potentially interacting humans. Internal geometric model representations of the scene and objects exist as shown on the display in the background of Figure 27.

Fixed furniture, for instance rectangular tables, as shown in Figures 32 and 34, as well as walls are not localized individually, but assumed at fixed poses in the world. Their poses relative to the robot are determined by means of robot self-localization based on a known laboratory map.

In contrast, moveable furniture objects, for instance round chairs and tables, shown in Figures 27 and 28, can be localized by the robots using a visual perception components [15]. Fur-

Figure 28: Picture of objects used in experiments.

thermore, those furniture objects can be pulled around by the robot. Similarly, small objects can be localized, transported or used for further manipulation actions. Such objects, shown in Figure 28, include cups, plates and a spatula.

Human body tracking can perceive body pose and activity type of one or two humans in the field of view. During recording, one or two interacting humans are considered, while during autonomous execution by the robot only one human, interacting with the robot is regarded.

Beyond the missions described in the following, similar simple service missions have been performed in the scope of parts of the presented system as described in [4] and [7].

The following missions were chosen for covering a varying, relevant and interesting combination of features in each mission, while resulting in state spaces of approximately 100 to 1000 states, one or two dozen action choices and non-trivial transition, reward and observation models. Slight variations of the missions, discussed in the following, were sometimes applied for different experiments, leading to slightly different state and action spaces as well as courses of events.

It has to be noted that larger missions are neither suitable for policy computation with current solvers, nor is the setup effort for autonomous learning and execution of such larger missions in a laboratory environment in the scope of quantitative evaluation reasonable.

In the missions described in the following, the elongated area in which the robot acts is necessary for the robot to be able to watch demonstrations in exactly the same layout, it has to act in autonomously later. Given the relatively little space in the lab at hand, that layout is the most versatile for different kinds of missions.

3.2 Mission PASE-I

The mission referred to as *PASE-I* centers around the robot monitoring an area were potentially entering humans may have the desire to sit down. The robot may proactively decide to fetch and pull a chair towards the human. The desire of the human to sit down hereby depends on the human pose. The following features are relevant in this mission:

- Robot self-pose
- Human-pose
- Furni-state

As illustrated in Figure 29, the geometric layout used has an area of human appearance on the one end and the initial location of a chair on the other end of an elongated area in the laboratory, the robot acts in. The robot typically waits near the center and pulls the chair towards the human. However, precise locations and state discretization depend on actual demonstrations. In different experiments, varying regions and orientations - and varying numbers of distinct feature states - for both human pose and chair pose were utilized, leading to different mission models.

Demonstrations typically include different transition probabilities of the human sitting down when presented with a chair, depending on both region and orientation after entering the scene.

No active dialog takes place between human and robot in this mission - neither by explicit gestures, nor spoken dialog - to underline the proactive nature of decision making.

Concerning grasping and pulling the chair, different manipulation strategy options are available, as shown in Figure 30, with different suitability depending on the initial pose of the chair.

Primary stochastic action effect probabilities arise from human intentions to sit down and initial chair pose placing. Secondary effect probabilities arise from robot navigation and manipulation action peculiarities. Crucial decisions encompass if to dare to pull the chair when a human is present and the belief about the human pose is unclear on the one hand and selecting the suitable grasp for the chair.

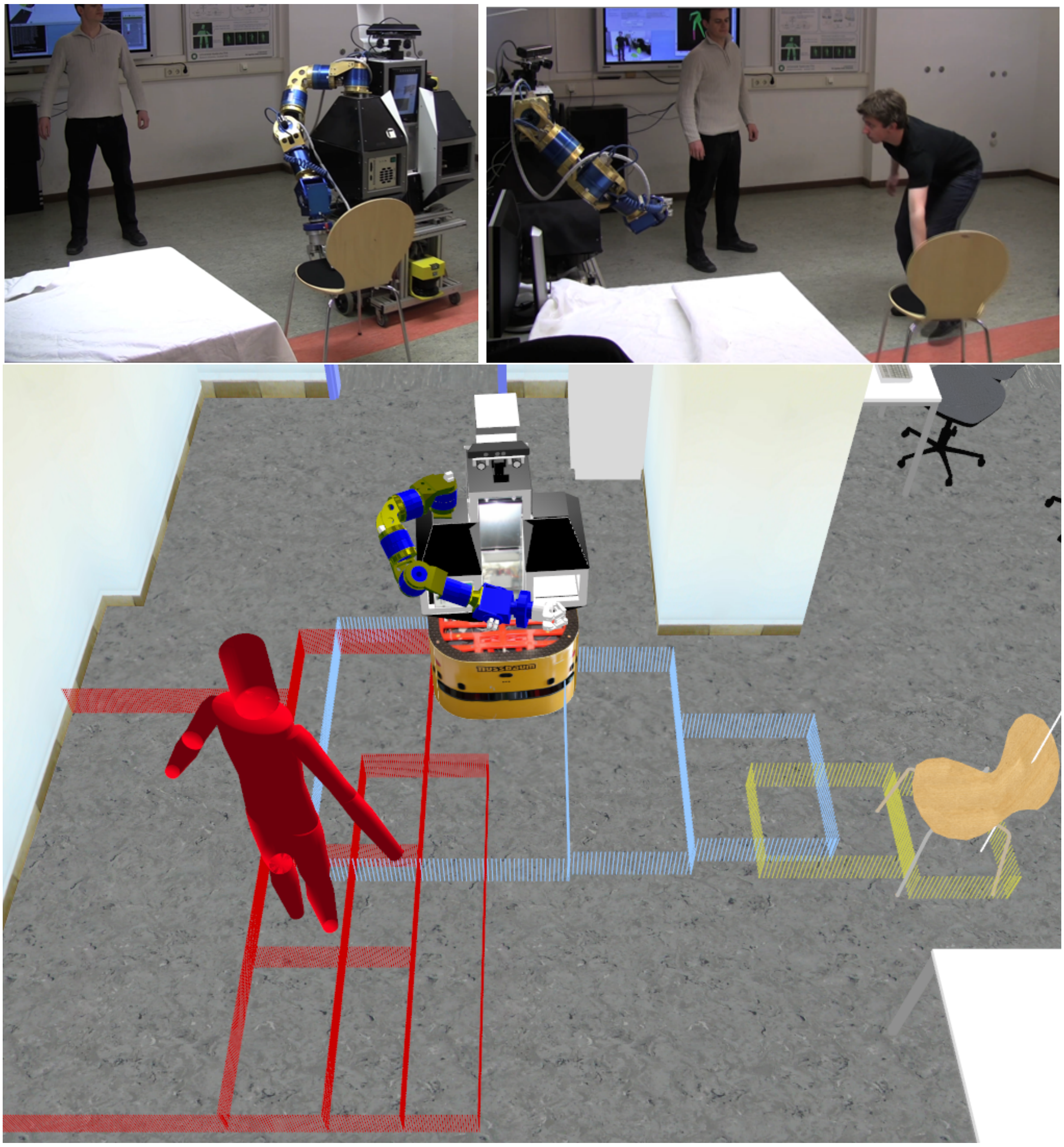

Figure 29: Pictures of exemplary situations in PASE-I: demonstration recording (top left), autonomous execution (top right) as well as exemplary feature state layout (bottom). [11]

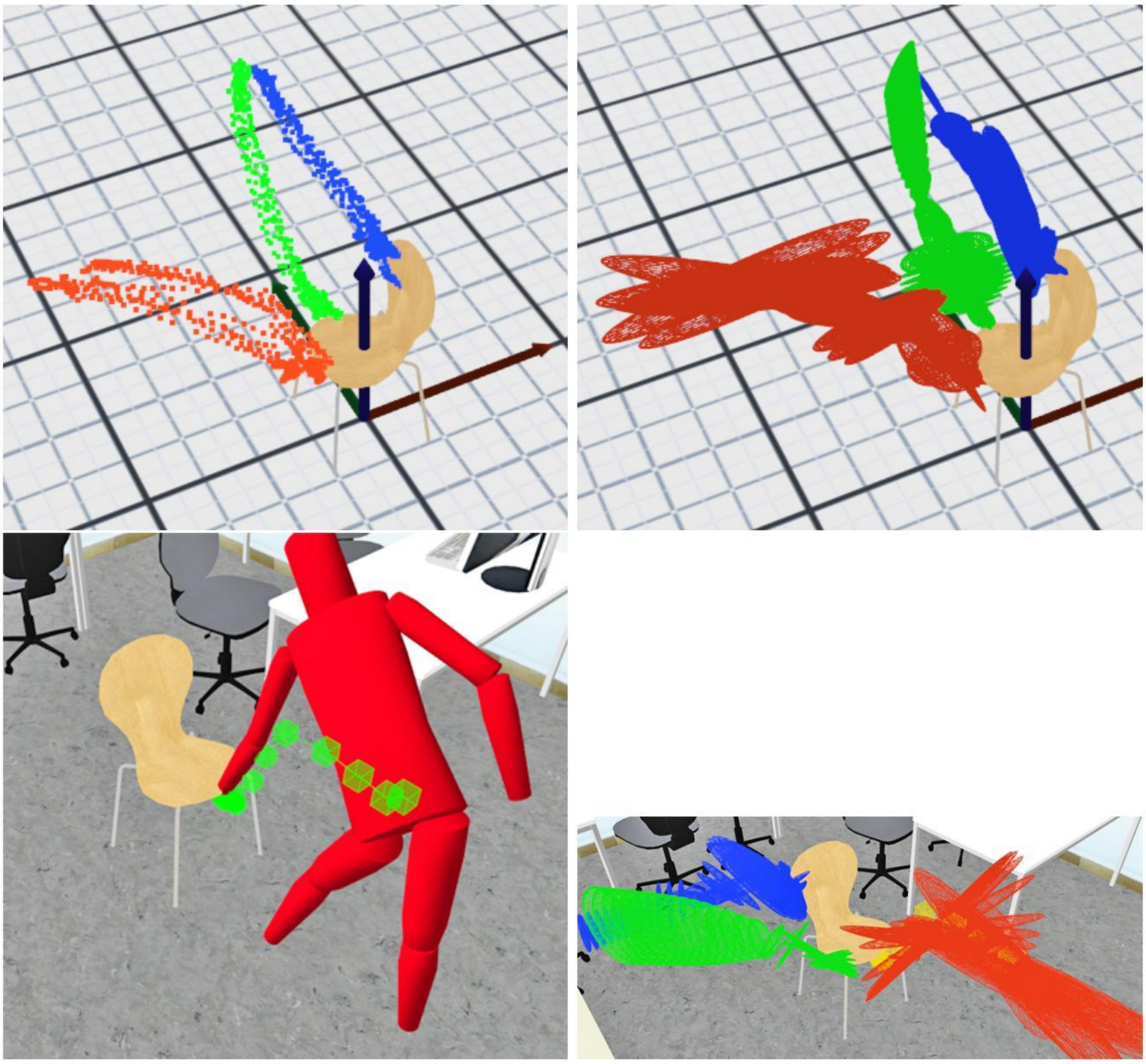

Figure 30: Manipulation action mapping in PASE-I: recorded manipulation strategies (top left) and exemplary recorded mission trajectory (bottom left), leading to a matching setup (bottom right).

3.3 Mission PASE-3.1

Referred to as *PASE-3.1*, this mission centers around a serving task. Hereby a human may interact with the robot by spoken dialog. The robot may bring a red cup or a blue cup to the human when requested for a cup. Those cups may initially be placed on different areas on a fixed table on one end of an elongated area, the robot acts in. This acting area of the robot is roughly similar to the area in PASE-I, however the object layout is different, as shown in Figure 32. On the opposite end of the fixed table with potentially some cups, another fixed table represents the place, the interacting human sits at. In front of the table with the cups, a moveable chair may block the access to one cup. The chair then has to be pulled away to allow access to the cup, as shown in Figure 31.

The following features are relevant in this mission:

- Robot self-pose
- Dialog-state
- Furni-state
- Small-obj-state

Proactive decision making centers around different options the robot has for instance when a red cup is requested, but blocked by the chair. There are the following options at this point:

1. Pull the chair away and fetch the red cup.
2. Ask if the blue cup is also ok and then react on the human reply.
3. Fetch the blue cup right away.
4. Decline and do nothing.

Hence, while the initial request of the human at first leads to reactive behavior - in contrast to PASE-I - subsequent choice selection is a proactive process. Naturally choices depend of transition probabilities of actions in individual choice sequences as well as action costs and goal rewards. Both primary and secondary transition probabilities as well as both action cost and goal reward values play an important role in this choice process. Therefore, this is a well suited example for POMDP decision making considering a service robot setting.

In this mission, there was little focus on varying state and action sets, but instead a quite similar layout was chosen each time and a emphasis made on analyzing varying transition probabilities and rewards.

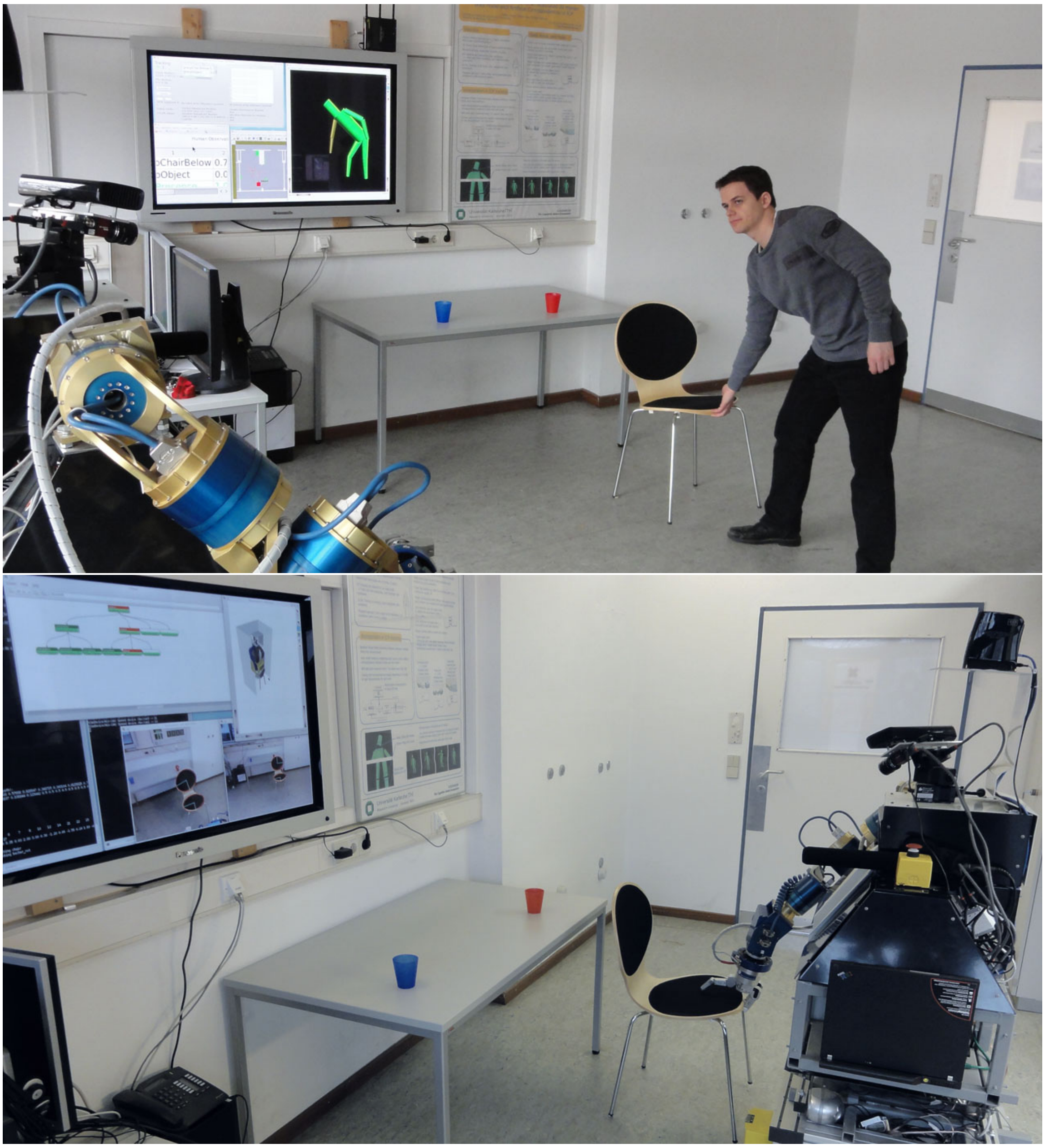

Figure 31: Pictures of exemplary situations in PASE-3.1: demonstration recording (top) and autonomous execution (bottom).

Figure 32: Pictures of exemplary PASE-3.1 situations: demonstration (top left), simulation (bottom left) and autonomous execution (right).

3.4 Mission PASE-3.2

Helping a human to pull a round table is the main objective of the mission, referred to as *PASE-3.2*. Basically, the robot observes the scene and proactively decides to help pulling the table, depending on the pose and fully body activity of the human.

The following features are relevant in this mission:

- Robot self-pose
- Furni-state
- Human-act

In case, the human activity seems to be to weak to succeed in pushing the table, the robot may approach proactively and help by pulling the table in the same direction as shown in Figure 33.

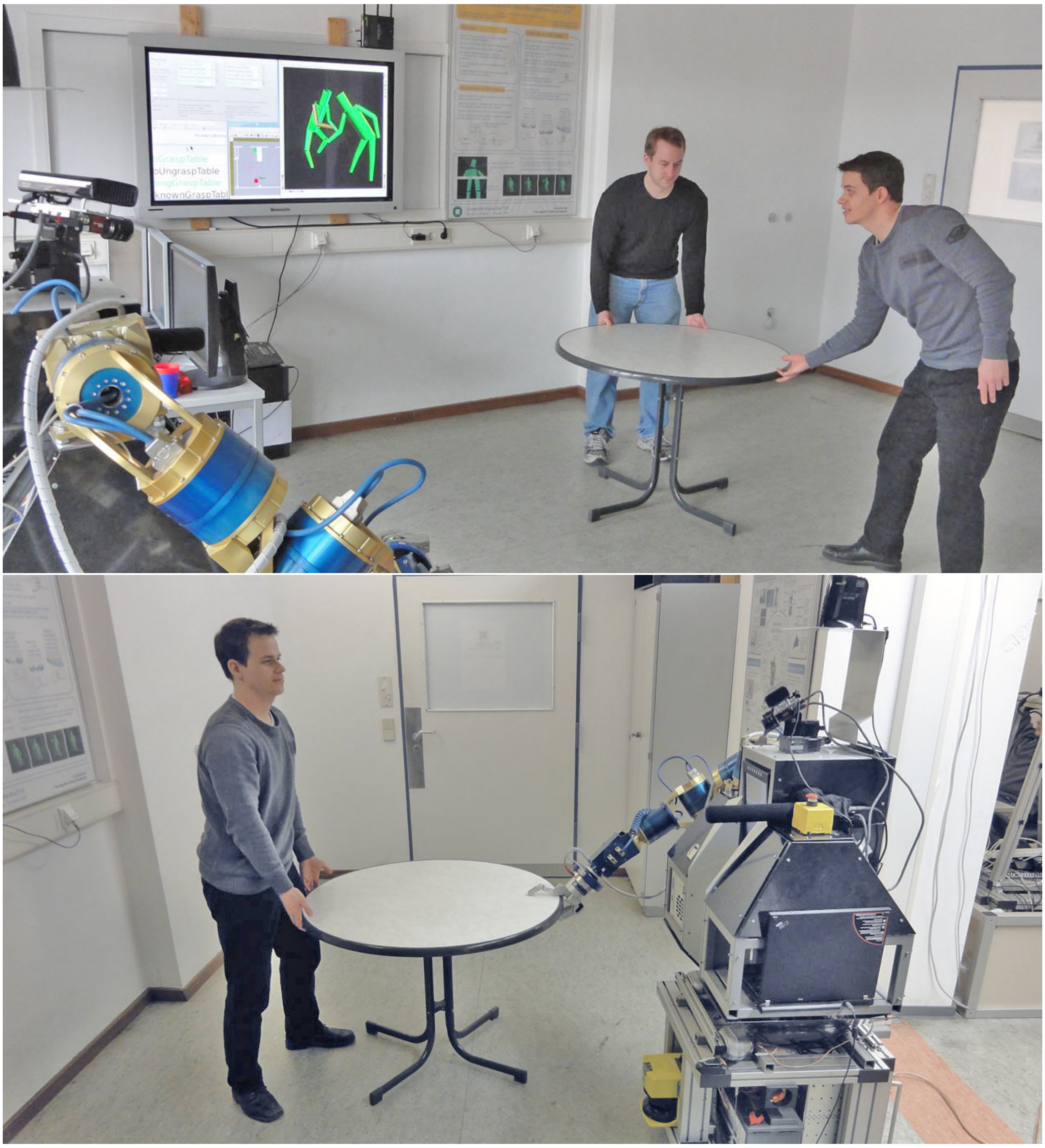

Figure 33: Pictures of exemplary PASE-3.2 situations: demonstration (top) and autonomous execution (bottom).

3.5 Mission AderoDexmart

In the mission alled *AderoDexmart*, the main objective is to swap a toast from one table to another. The layout of the mission is organized around two opposite fixed tables, the robot may navigate between. One of the tables may be substituted by a fixed cupboard and initially a deep plate may be placed there. On the other fixed table, most manipulation actions take place and the deep plate may also be located there. Furthermore, on this table, a spatula and a shallow plate are located as shown in Figure 34. The robot has to fetch the deep plate or push it into a suitable position on the main manipulation table, can then pick the spatula and swap a toast from the shallow plate onto the deep plate. If object poses are unsuitable for known manipulation strategies, the robot may call a human for help to improve object poses.

The following features are relevant in this mission:

- Robot self-pose
- Small-obj-state

Proactive decision making occurs when choosing in which order to collect the items, where to look for the deep plate first, when to look for object poses again, when to push the deep plate or spatula into a more suitable pose and when to call a human for help.

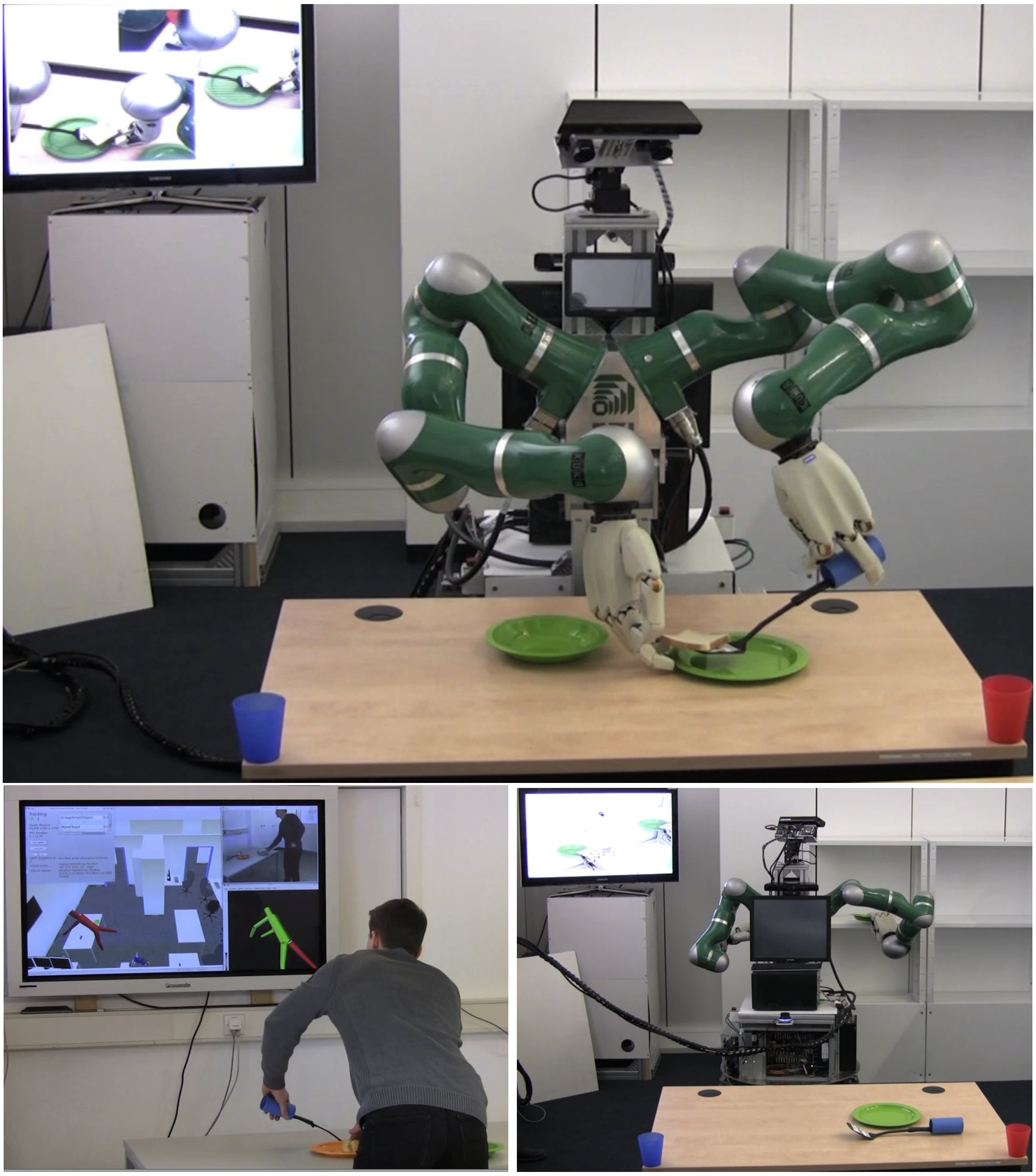

Figure 34: Picture of exemplary AderoDexmart situations: demonstration (bottom left) and autonomous execution (top, bottom right).

References

[1] A. R. Cassandra, L. P. Kaelbling, and M. L. Littman, "Acting optimally in partially observable stochastic domains," in *In Proceedings of the Twelfth National Conference on Artificial Intelligence*, 1994.

[2] S. Russell and P. Norvig, *Artifcial Intelligence - A Modern Approach*, 2nd ed. Pearson Education, 2003.

[3] S. R. Schmidt-Rohr, M. Lösch, Z. Xue, and R. Dillmann, "Hardware and software architecture for robust autonomous behavior of a domestic robot assistant," in *10th International Conference on Intelligent Autonomous Systems*, 2008.

[4] S. R. Schmidt-Rohr, S. Knoop, M. Lösch, and R. Dillmann, "Bridging the gap of abstraction for probabilistic decision making on a multi-modal service robot," in *Robotics: Science and Systems*, Zürich, Switzerland, 2008.

[5] S. Knoop, S. R. Schmidt-Rohr, and R. Dillmann, "A flexible task knowledge representation for service robots," in *The 9th International Conference on Intelligent Autonomous Systems (IAS-9)*, 2006.

[6] M. Lösch, S. Schmidt-Rohr, S. Knoop, S. Vacek, and R. Dillmann, "Feature set selection and optimal classifier for human activity recognition," in *Robot and Human interactive Communication, 2007. RO-MAN 2007. The 16th IEEE International Symposium on*, Jeju Island, Korea, 2007.

[7] S. R. Schmidt-Rohr, M. Lösch, R. Jäkel, and R. Dillmann, "Programming by demonstration of probabilistic decision making on a multi-modal service robot," in *In proceedings 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Taipeh, Taiwan, 2010.

[8] J. Stahl, "Modellierung und automatische Zustandsraumdiskretisierung für interaktives Lernen probabilistischer Entscheidungsmodelle von Servicerobotern," Master's thesis, Masterarbeit, KIT, Betreuer: Sven Schmidt-Rohr, Gutachter: Rüdiger Dillmann, 2011.

[9] R. Jaekel, S. R. Schmidt-Rohr, M. Loesch, and R. Dillmann, "Representation and constrained planning of manipulation strategies in the context of programming by demonstration," in *IEEE International Conference on Robotics and Automation (ICRA '10)*, 2010.

[10] T. Engelhardt, "Automatische Zuordnung von detaillierten Manipulationsstrategien auf entsprechende Teilaktionen von Missionsvorführungen bei Programmieren durch Vormachen von autonomen Servicerobotern," Master's thesis, Studienarbeit, KIT, Betreuer: Sven Schmidt-Rohr, Gutachter: Rüdiger Dillmann, 2011.

[11] S. R. Schmidt-Rohr, F. Romahn, P. Meissner, R. Jäkel, and R. Dillmann, "Learning probabilistic decision making by a service robot with generalization of user demonstrations and interactive refinement," in *12th International Conference on Intelligent Autonomous Systems*, 2012.

[12] S. R. Schmidt-Rohr, G. Dirschl, P. Meissner, and R. Dillmann, "A knowledge base for learning probabilistic decision making from human demonstrations by a multimodal service robot," in *The 15th International Conference on Advanced Robotics*, 2011.

[13] C. Wischnewski, “Arbeitsraumbezogene Analyse der Wahrscheinlichkeiten von Aktionseffekten für probabilistisches Entscheiden eines autonomen Roboters,” Master’s thesis, Diplomarbeit, KIT, Betreuer: Sven Schmidt-Rohr, Gutachter: Rüdiger Dillmann, 2012.

[14] M. Seidel, “Lernen aus Erfahrung in Dynamiksimulation für einen Serviceroboter auf Basis von Programmieren durch Vormachen,” Master’s thesis, Bachelorarbeit, KIT, Betreuer: Sven Schmidt-Rohr, Gutachter: Rüdiger Dillmann, 2012.

[15] P. Meißner, S. R. Schmidt-Rohr, M. Lösch, R. Jäkel, and R. Dillmann, “Robust localization of furniture parts by integrating depth and intensity data suitable for range sensors with varying image quality,” in *Proceedings of the 15th International Conference on Advanced Robotics (ICAR ’11)*, 2011.