STAIRS 2016
D. Pearce and H.S. Pinto (Eds.)

doi:10.3233/978-1-61499-682-8-3

Human Robot Collaboration to Reach a Common Goal in an Assembly Process

Sharath Chandra AKKALADEVI[a,b,1], Matthias PLASCH[a], Andreas PICHLER[a] and Bernhard RINNER[b]
[a] *Department of Robotics and Assistive Systems, Profactor GmbH*
[b] *Institute of Networked and Embedded Systems, Alpen-Adria-Universität Klagenfurt*

Abstract. Enabling robotic systems to collaborate with humans is a challenging task, on different levels of abstraction. Such systems need to understand the context under which they operate, by perceiving, planning and reasoning to team up with a human. The robotic system should also have perspective taking capabilities in order to efficiently collaborate with the human. In this work an integrated cognitive architecture for human robot collaboration, that aims to develop perspective taking capabilities using human preferences, is proposed. This is achieved by developing a 'mental model' that takes human preferences, the knowledge of the task (including the objects), and the capabilities of the human and the robot. This mental model forms the basis of the cognitive architecture, to perceive, reason and plan in the human-robot collaborative scenario. The robotic platform guided by the cognitive architecture, performs 'picking', 'showing', 'placing' and 'handover' actions on real world objects (of interest in the assembly process) in coordination with the human. The goal is to answer the 'how' (how a manipulation action should be carried out by the robot in a dynamically changing environment) and the 'where' (where the manipulation action should take place) of the assembly process considering/given varying human preferences. We show that the proposed cognitive architecture is capable of answering these questions through various experiments and evaluation.

Keywords. Human Robot collaboration, Common Goal, Human Robot Interaction, Knowledge representation, task planning, perception and vision in robotics

1. Introduction

The concept of robots cooperating with humans has gained a lot of interest in recent years, in both domestic and industrial areas. Combining the cognitive strength of humans together with the physical strength of robots can lead to numerous applications [3]. For example, in industrial scenarios a certain assembly processes requires the worker's strenuous effort of lifting heavy objects, operating in non-ergonomic positions etc., which lead to negative long-term effects. This is becoming increasingly important also because of the aging work-force [3] and the fact that there is a trend to automate such work-places even if it does not lead to additional short-term profits [10]. The preferred solution for these work-places is a robotic assistant to interact and aid a human operator rather than a fully automated system. Combining the flexibility of adapting in humans with the physical strength and efficiency of the robots/machines will potentially

[1] Corresponding Author, Profactor GmbH, Im Stadtgut A2, 4407 Steyr-Gleink, Austria | Alpen-Adria-Universität Klagenfurt, Klagenfurt, Austria; E-mail: sharath.akkaladevi@profactor.at

transform life and work practices, raise efficiency and safety levels and provide enhanced levels of service [3][6].

Human robot interaction (HRI) is a challenging field that combines robotics, artificial intelligence, cognitive science, computer science, engineering, human computer interaction, psychology and social science [9]. One of the primary goals of this research is to find an intuitive way in which humans can communicate and interact with a robot [2]. The essential components of HRI include evaluating the capabilities of humans and robots and designing the technologies and training that produce desirable interactions between them [12]. Humans, in general have perception and cognitive functions and are able to act and react with respect to a given situation. Characterizing and understanding a situation, to describe and detect a situation and often predict the next steps, comes naturally to humans. Whereas, to develop a robotic system with 'Context Awareness' or in other words to enable a robotic system to understand the circumstances under which they operate and react accordingly in a cooperative fashion is a challenging task [16].

Human robot interaction can be realized in various forms. A binary input (e.g. yes or no) from the human to the robot can be seen as an interaction in its simplest form. Depending on the kind of interaction [22], HRI in industrial scenarios can be partitioned into: a) human robot coexistence – where both agents (human and robot) operate in a close proximity on different tasks; b) human robot assistance – where the robot passively aids the human in a task (helping in lifting heavy objects); c) human robot cooperation – where both agents simultaneously work on the same work piece (each agent has their own task to do on the work piece); and d) human robot collaboration – where both agents perform coordinated actions on the same task (for e.g., robot handing over a work piece to the human operator, who then completes the coordination by taking the work piece).

In order to intuitively interact with humans, the robots should know not only the properties of objects but also the capabilities of other agents (humans) in their environment [2]. For a close collaboration with humans, the robotic systems should also attribute meaning to beliefs, goals and desires of humans during a particular task. These set of (meta) representational abilities can be collectively called as "mental models" [18]. This would not only allow the robotic system to understand the actions and expressions of humans within an intentional or goal-directed architecture [2], but also for the human operators to better understand the capabilities of the robotic system [18].

In this paper we propose an architecture that combines state of the art object tracking [15], action recognition [14] approaches, together with a real-time robotic path planning system. To facilitate a human robot collaboration scenario with a common goal, the architecture is enabled with cognitive capabilities, where the cognition arises from the reasoning, simulating and planning behavior of the architecture.

The main contributions of this paper are to present an integrated cognitive architecture, that combines state of the art perception (object tracking, human action recognition) and planning algorithms, to model, reason and interact in an assembly process that involve

- real world object manipulations, see **Figure 1**,
- a common goal between human and robot to complete an assembly process, and
- preliminary results of an integrated cognitive architecture, evaluating the aspects 'where' (what is the suitable location for the task) and 'how' (how should the task be carried out) of the assembly process.

The remaining part of the paper is structured as follows: In section 2 we review the state of the art approaches in human robot interaction which consider the perspective of the human ('Perspective Taking'), followed by the problem statement description in section 3. Section 4 explains then the cognitive architecture in detail. The experimental setup and the evaluation of the proposed cognitive architecture in dealing with the 'where' and 'how' are presented in section 5, followed by conclusion and future steps in section 6.

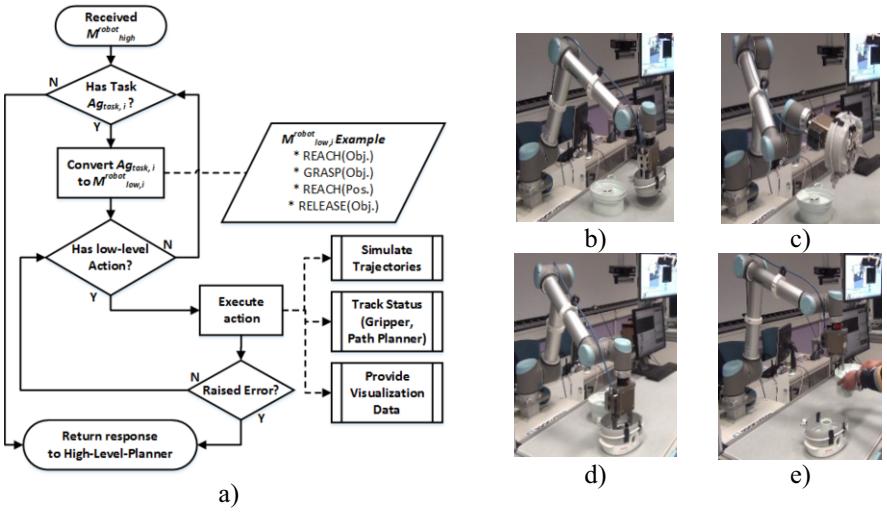

Figure 1. a) Flow chart describing the reasoning and planning in Low-level planner. Robotic manipulations/assistance during the assembly process b) '**Picking'** – robot reaching for *Heater/Tray*, grasping *Heater/Tray* and lifting up, c) '**Showing'** – Presenting the grasped *Heater* to the human, d) '**Placing'** – Putting down the compound *Heater-Base* object and releasing gripper, e) '**Handover'** – Presenting the *Tray* to the human. The robot releases the gripper when human reaches for the *Tray*

2. Related Work

Human robot interactions are demonstrated in different levels of abstraction in literature. The abstraction ranges from close proximity simultaneous task execution between human and robot to collaborative execution of a task with a common goal. A detailed survey on human robot interaction and emerging fields is given in [12]. In this section we would like to review HRI approaches where the robot interacts with the human by considering the 'mental model' of the human. This allows the robotic system to take actions from the perspective of human to facilitate a 'natural' interaction.

Schrempf et al. [17] present a system architecture for human robot cooperation that allows the robot to plan its actions depending on the user's intention, using a probability density function. A planning framework that allows a human and a robot to perform simultaneous manipulation tasks safely in close proximity is proposed in [8]. The framework generates a prediction of human workspace occupancy. The motion planner then plans a cost based trajectory with minimum penetration into human workspace to enable simultaneous manipulations.

A cognitive system capable of handling ambiguous situations where the robot can perceive two similar looking objects, but where one of the objects is occluded by the human is presented in [7]. When asked for the object, the robotic system [7] takes the

visual perspective of the human to determine which object the human referred to. Gray et al. [4] present an architecture that aims to manipulate the mental states of a human through robot actions. The framework demonstrates a competitive game scenario in which the robot's actions influence the human's mental states through their visual perception.

Pandey et al. [1] presented a human robot interaction framework that enables the robotic system to take the visual-spatial perspective of the human and decide the effort level involved for the human in performing a particular task. This information of varying effort levels in doing a task depending on the current position of the human is used by the robotic system to initiate an interaction in a manner that has the least possible effort for the human. However, they consider artificial objects (with markers) for manipulation and a motion capture system (with complicated calibration process [20]) to capture the visual perspective of the human. Our approach considers manipulation with real world objects and relies on rgb-d sensors to derive the current status of the involved entities.

These approaches show the importance of modeling the perspective of the human (goals, beliefs, desires) by the robotic system for an 'intuitive' human robot interaction. However, we differentiate from them by integrating state of the art object tracking and human action recognition approaches into the cognitive framework. We focus on human robot collaboration in a real world assembly process. Our system also takes into consideration the dynamic changes in the environment while developing the mental models, which describe the human perspective to achieve the goal.

3. Problem Statement

Given the 'mental model' that includes knowledge of the assembly process, involved objects, human capabilities (and preferences), robot capabilities and the environment (all entities in the workspace – objects, robot/s, human/s, etc.,), the main prerequisite for the architecture is to decide 'what' (which step) should be done 'when' (at which state instance in the assembly process) in order to attain the common goal. Given the 'what' the next question is to decide 'who' (the human or the robot or both together) is better suited for the task. These questions to some extent were answered in [4][7][8][17], but there is also an interesting research question that asks 'where' (at which location in the environment) should the task happen. The authors in [1] proposed an approach to answer this question. We extend the above research questions and pose the question 'how' (how should the task be carried out, i.e., collision free object manipulation considering human preferences) and propose a solution that can deal with real world objects. When dealing with real world objects, it is also important to 'reason' about collision free (with the environment, other objects, human) manipulation (See **Figure 1** b, c, d, e – 'Picking', 'Showing', 'Placing' and 'Handover' resp.) to ensure successful execution of the task. A mathematical formulation of the problem is given below.

The assembly process (AP) is a collection of a set of States (S) , a set of Events (V) and a set of Relations (R). The terms State and task state are used analogously in this work. The set of States (S) define the individual steps of the assembly process. The set of Events (V) drives the progress of the assembly process from one step to another. The Relations (R) specify the effect of a given Event V_m on a given State S_i in progressing the assembly process. At any instance of time, the assembly process is said to be in a given State $(S_{instance})$ on which an Event (V_{occur}) can occur (or is occurring) that could change the progress of the assembly process from one State to the other, whose

relations are defined in R. In this work, the knowledge about the assembly process (S, V, R) is assumed to be known a priori.

Depending on the current State $S_{current}$, the architecture triggers an Event $V_{trigger}$ to progress the assembly process. This triggering of an Event is defined by a manipulation plan M^p.

The set of States $= \{S_1, S_2, S_3 \dots S_n\}$, where each State instance S_i corresponds to an assembly step, are known a priori (n defines the no. of assembly steps). An instance of a State S_i corresponds to an individual step of the assembly process (AP). A State instance of the assembly process can be described with a set of tuples $\langle Actors, Objects, Constraints \rangle$.

A State instance is said to be a collection of *properties*, where each tuple $\langle Actors, Objects, Constraints \rangle$ is called a *property* of that State instance, and together the set of tuples describe the State instance. In other words, the assembly process is said to be in a State instance, when all the *properties* of that State instance are satisfied. The set of $Actors$ is defined as $A = \{Human, Robot\}$ and it describes the actors involved in the assembly process. The set of Objects (O) consists of $\langle id,\ pose \rangle$ pairs that define the object identification and the pose estimation parameters (position and orientation) of all objects in the scene. The collection of constraints in the set of $\langle Actors, Objects, Constraints \rangle$ tuples, define the conditions for each *property* of the State instance and are designed in such a manner that, when satisfied, the assembly process is said to be in that State instance. The *properties* of a State instance can either be observed directly or inferred. The constraints of a State instance can be composed of the following:

- **Actor Constraints**: For a single tuple describing a *property* of the State instance, either the set of Actors or the set of Objects, could have a null value when they are deemed unnecessary. This is the case when it is only sufficient to constrain either the Actor or the Object set to describe that *property* of the State instance

- **Object Constraints**: For example, a required *property* of a State instance could be, say an object O_l should be on top of the table. In such cases it is sufficient to only constrain object O_l to be on the top of the table and ignore the Actors, for that particular *property*

- **Actor-Object Constraints**: This constrains the status of the actor and status of the object w.r.t each other and the assembly process. If the constraint is that the human actor has to be holding a particular object, then it is not sufficient to constrain just the human status or the object status but both of them applied in relation with each other.

The set of Events $V = \{V_1, V_2, V_3, \dots V_m\}$ comprises m events that can occur during the assembly process. An Event occurs during the assembly process only because of the activities performed by the human or the robot. An Event is described as an activity performed by an agent (Ag) on a $target$ (a $target$ could be other agents (Ag) or objects (O)). An instance of an Event V_m can be defined as a tuple $\langle Ag,\ Acts \rangle$. Agents (Ag) can have a value $\{Human, Robot,\ both\}$, while $Acts$ is a pair defined as $\langle ActivityType,\ target \rangle$. $ActivityType$ describes the type of activity performed by the agent. This activity could be a single action independent of the $target$ or actions that involve interaction with a $target$.

The set of Relations $= \{R_1, R_2, R_3, \dots R_q\}$, where each Relation R_q consists of a tuple $\langle S_{instance},\ V_{occur},\ \{S_{result}: result(S_{instance},\ V_{occur})\}\rangle$, describes the effect of an Event V_{occur} occurring on a given State instance $S_{instance}$ and provides the resulting State S_{result} in the assembly process. In case of the final State of the assembly process, the set of Relations (R) is a null set. The following holds during the assembly process: $\forall\ S_i\ \exists\ R^i$, where $R^i \subseteq R; i = 1, \dots, n$. For every State instance S_i, there exists a set of Relations (R^i) that defines the set of possible Events (V^i) that could occur on that State instance which then result in a set of next States (S^{i+1}), where $V^i \subseteq V$and $S^{i+1} \subseteq S$.

Given the knowledge about the assembly process $AP\ (States, Events, Relations)$, the purpose of the cognitive architecture (See **Figure 2**) is to generate a manipulation plan M^p that facilitates the assembly process to proceed from the current State to the next, in order to achieve the common goal (of completing the assembly process). The manipulation plan is abstracted in two layers: namely M^P_{high} and M^P_{low}. M^P_{high} denotes the 'what', 'who' and the 'when', while M^P_{low} denote the 'where' and 'how'. The reason for differentiating the manipulation plan is twofold a) to enable the system to consider dynamic changes while performing the task b) to enable the cognitive architecture to be robot embodiment agnostic (for more details see 4).

The manipulation plan M^p_{high} is described as a tuple $\langle Ag_{task},\ Params\rangle$, where, Ag_{task} describes the manipulation task that should be carried out by the respective agent (human, robot, or together). $Params$ consists of the necessary information that involves the required object on which a manipulation should occur and in which fashion 'how'. The 'how' (In what manner a manipulation task should occur/ is expected to occur) is based on the 'mental model' of the cognitive architecture, which considers human preferences, knowledge of the AP, the current status of AP and the environment.

Once the manipulation plan M^p_{high} is generated it is converted into a set of 'low level' executions M^p_{low} that consider the dynamic changes in the environment (object configuration, human position, robot status). Given the 'what' (M^p_{high}) the aim is to find the 'where' (what is the best possible location for a task) and 'how' (what is the best way to carry on the task) -M^p_{low}, considering the current sensor input, the target State and known human preferences. Human preferences are communicated to the system via direct user input, who at the time of task abstraction also creates a preferred manner in which a task needs to be done. Human preferences can also be inferred by observing how a human operator executes the task. If there are no preferences mentioned for a task execution, the system presents different alternatives possible for doing the task and then infers the preferences from the human execution. It is not in the focus of this work to deal in detail with human preference modeling (creating mental models), but to concentrate on the 'where' (and 'how') of the manipulation execution, given the knowledge about the assembly process and human preferences.

4. Architecture

The cognitive architecture proposed for a human robot collaboration scenario with a common goal of completing the assembly process is as shown in **Figure 2**. The reasoning capabilities of the architecture are twofold: a) On the one hand, the High-level planner with the knowledge of the assembly process perceives, reasons and initiates the necessary cooperative behavior of the robotic system. Its output is a manipulation plan M_{high}^{p} describing 'what' needs to be done, 'when' and by 'whom'. These instructions are provided to the Low-level planner for realization of the common goal. b) Though the

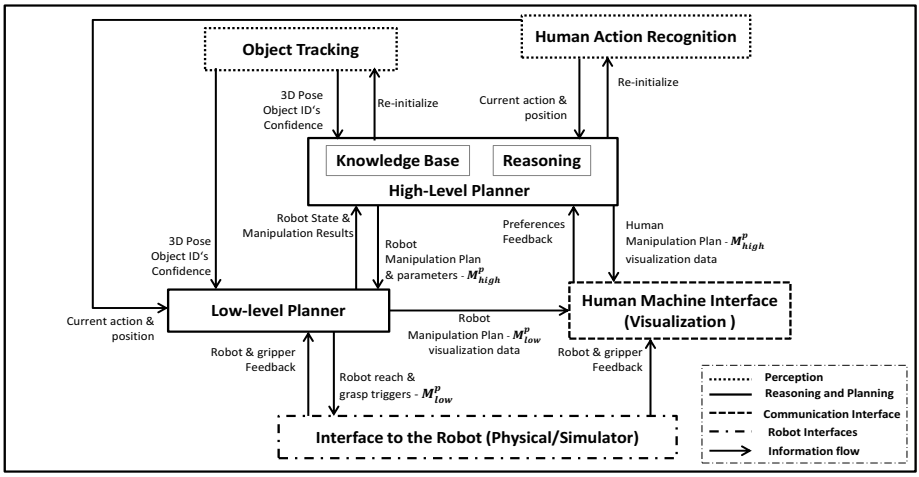

Figure 2: The Cognitive Architecture capable of dealing with the 'where' and 'how' of the collaboration

High-level planner provides the manipulation plan, it is equally important to decide 'how' and 'where' a manipulation task (M_{low}^{p}) is to be carried out. The individual modules of the architecture are briefly described below.

4.1. Human Action Recognition and Object Tracking

Perception and classification of human actions as well as object recognition and tracking are important prerequisites to establish a human robot collaboration system. A state of the art action recognition framework [14], using skeleton tracking, classifies actions by applying a multi-class random forest classification technique. The module is capable of providing the current action executed by the human and the position (skeleton tracking) in real-time. Object tracking in 3D can be defined as the problem of estimating the trajectory (6 DOF) of an object in the 3D point cloud as it moves around a scene. The tracking approach currently developed and used [15] relies on depth data only, to track multiple objects in a dynamic environment. It builds on random-forest based learning techniques to deal with problems like object occlusion, motion-blur due to camera motion and clutter.

4.2. High Level Planner

The High-Level Planner provides the main reasoning functionalities to the cognitive architecture. Its major purpose is to reason about the current state $S_{current}$ of the

assembly task and to generate a manipulation plan of actions M_{high}^{p} to be carried out, in order to reach the next attainable assembly state. This is done by considering the current situation of the surrounding environment, a-priori knowledge, as well as experiences from previous perceptions and assembly task execution cases. The upcoming sub-sections describe the functional components of the High-Level Planner in detail.

4.2.1. Knowledge Base and Knowledge Management

Knowledge within the High-level planner (See **Figure 2**) is organized using two separated databases. An a-priori knowledge base is used to store permanent knowledge, including assembly process descriptions (AP) (see section 3), environmental knowledge (e.g. CAD models of workspace and objects of interest), and configuration data (e.g. capability description of functional system components, human workers and robots).

And the other is the online database (the online version of knowledge base), that is used to share data which is created during assembly process execution. These data sets are transient and include perception data (e.g. object configurations and human actions), human worker preferences, instances of planned or perceived actions and their parameterization, and identified assembly task variations (i.e. an adaptation of an assembly task description, based on newly classified State/s and Event/s). After assembly task execution, newly experienced data can be moved into the a-priori knowledge base thus extending permanent knowledge. The data structures of the knowledge bases build on the knowledge processing framework KnowRob [13]. In this work, the basic ontologies of KnowRob were extended in order to cover the description of assembly process variants as well as pre-, operational- and post-operational conditions for Events (V). Functional system components including their capabilities are described using the Semantic Robot Description Language (SRDL) [11]. Combined with perception data, the databases provide major input for the Reasoning system.

4.2.2. Reasoning System

The reasoning system accesses data from both the a-priori and the online database, in order to create qualitative hypotheses on object configurations (the way the objects are spatially arranged) and ongoing human activities involving objects, by considering the assembly task context and perception data. Based on these hypotheses the reasoning system reasons about the current status of the assembly process. As described earlier in section 3, the assembly state instance $S_{instance}$ is described by a set of tuples, including $Actors, Objects\ and\ Constraints$. The reasoning system checks if the recent perception results (hypotheses), match the required constraints. This process is triggered once the system expects a stable assembly State, or when an unexpected Event occurs.

Given the current State of the assembly process, the reasoning system deduces the next Event V_{next} (including its sub actions), which needs to occur in order to proceed towards the common goal. In case of the next required Event being detected, the required action instances are created, stored in the online database. To commence the execution of action instances, manipulation plans M_{high}^{p} for robotic and human actions are generated, also observing user preferences stored in the online database. Manipulation plans containing robot actions are sent to the Low-level planner for the physical execution. For human actions, task relevant information to guide the human is sent to the visualization module. The execution of human actions is observed by the reasoning system.

4.3. Low Level Planner

The main goal of the Low-level planner is to convert high level manipulation plans M_{high}^{robot} for robotic actions, into low level plans M_{low}^{robot} and to execute those. **Figure 1** a) depicts a flow chart describing the execution workflow. Considering a received plan M_{high}^{robot} (see description in section 3), the Low-level planner converts and executes each manipulation task pair $\langle Ag_{task,i},\ Params_i \rangle$ into a low level task one by one. For a single converted task $M_{low,i}^{robot} \subseteq M_{low}^{robot}$, the low level actions (e.g. REACH, GRASP,…) are executed iteratively. According to the given parameters $Params_i$, the current object configurations of interest, human activity status, or target positions within the workspace, are considered during execution. With this information, the Low-level planner simulates collision-free reach and grasp operations of the robot using [5], also satisfying user preferences. The trajectories which suit best are executed and visualized, to show the intention of the robotic system to the human operator. This status information is communicated to the High-level planner, once the manipulation plan M_{high}^{robot} was fully executed (no manipulation task pairs pending) or one single task failed.

4.4. Visualization for Human Machine Interfaces

The visualization module not only conveys the intent of the robotic system to the human operator but also guides the human operator in performing a required manipulation task. The manipulation plan $(M_{high}^{p}, M_{low}^{p})$ received from the High-level and Low-level planner respectively aids the visualization module to display the information necessary during the AP. It provides a direct platform for the human operator to communicate with the cognitive system (communicating user preferences) and vice versa. The cognitive system however, is actively monitoring the complete environment (including the human operator) to infer the current situation of the assembly process and characterize it. The information about the cognitive system state (what the cognitive system thinks about the current status of the assembly process), and intention (that suggest what should happen next), the required assistance from the human operator and the current status of the robotic platform are visualized by the visualization module.

4.5. Robotic Platform Components

The robotic platform consists of a UR10 [21] robotic arm with 6 degrees of freedom. It also consists of a SCHUNK 2 finger electric parallel gripper [19]. These components provide interfaces to the Low-level planner to send reach (trajectory plan) and grasp triggers to the robot arm and the gripper respectively. The status information that is returned by the robot (joint angles) and the gripper is used by the Low-level planner to determine success or failure of the associated manipulation execution.

5. Experiments and Evaluation

The experiments carried out, focus on an evaluation of the capabilities of the Low-level planner, specifically dealing with the aspects of ‘where’ and ‘how’ the robotic assistance should be provided. **Figure 1** (b, c, d, e) depict the type of robotic assistance possible in

the chosen assembly process of a steam cooker. **Figure 3** (a) depicts the robotic system setup, including the human operator. For the considered common goal of assembling a steam cooker device, the assembly task states and Events, as given in **Table 1**, are known a-priori to the system. A video describing the robotic system and the assembly task can be found **here**[2]. It is recommended to watch the video to better understand the assembly process and the functional components involved. Having the common goal of assembling a steam cooker device, the performance indicator of the integrated cognitive architecture is defined as the ability of the system to provide assistance to the human operator during the assembly process, at a position that is suitable (preferred by human). In this work, user preferences are specified prior to the execution of the assembly task (assumption).

Table 1. Description of states of the Steam-Cooker assembly task, according to the notation, introduced in section 3. Events to proceed to the next assembly state are described in the last column. Example: In State S_1, the human actor is HOLDING the *Base*, the robot is IDLE, the *Heater* is located OnTop of the Table. In order to proceed to the next state S_2, perform Event V_2: Robot executes a PickAndShow action on the *Heater* involving the 'showing-location' defined by the human preference.

State	Actors	Objects	Constraints	Event V_{next}
S_0	Both		*RS,HA::IDLE*	V_1<*human, <PickAndShow, Base>>*
		Base	*SR::OnTop<Table>*	
S_1	Human	*Base*	*HA::HOLDING<Base>*	V_2<*robot, <PickAndShow, Heater,*
	Robot		*RS::IDLE*	*Loc::Human-Preference>>*
		Heater	*SR::OnTop<Table>*	
S_2	Human	*Base*	*HA::HOLDING<Base>*	V_3<*human, <Assemble, Base,*
	Robot		*RS::IDLE*	*Loc::Heater>>*
S_3	Both		*RS,HA::IDLE*	V_4<*robot, <Placing, H-B,*
				Loc::EmptyPlace>>
S_4	Both		*RS,HA::IDLE*	V_5<*human, <PutObjectInto,*
		Ring	*SR::OnTop<Table>*	*Ring, H-B>>*
		H-B	*SR::OnTop<Table>*	
S_5	Both		*RS,HA::IDLE*	V_6<*robot, <PickAndHold, Tray>>*
		Tray	*SR::OnTop<Table>*	
		H-B-R	*SR::OnTop<Table>*	
S_6	Robot		*RS::IDLE*	V_7<*robot, <Release, Tray>>*
	Human	TCP	*HA::REACHING*<TCP>	
S_7	Robot		*RS::IDLE*	V_8<*human, <Assemble, Tray, H-B-R>>*
	Human	*Tray*	*HA::HOLDING<Tray>*	
		H-B-R	*SR::OnTop<Table>*	
S_8	Both		*RS,HA::IDLE*	no Event
		SC	*SR::OnTop<Table>*	

RS: Robot State**; HA:** Human Action**; SR:** Spatial Relation**; *H-B:*** *Heater-Base* object; ***H-B-R:*** *Heater-Base-Ring* object; **SC:** Steam-Cooker; **TCP:** Robot Tool Center Point; **Loc:** Location

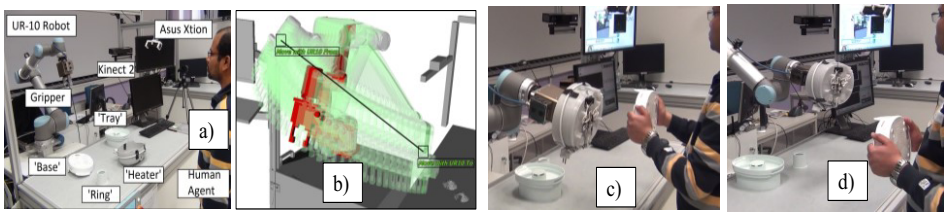

Figure 3. (a) System setup including robotic platform (robot, gripper), 3D sensors, steam cooker parts and the human agent. (b) Visualization showing planned robot movements. Presenting objects for compound object assembly, where human preference of location is (c) right hand side (d) left hand side

[2] Assembly process video: https://www.youtube.com/watch?v=URfJUMNc9SY

In the given case, the heater part is presented with respect to the detected "spine-shoulder" joint [14]. An offset position to this human joint can be chosen in any direction, as shown in **Figure 3** (b) and (c). For performance evaluation, a number of trials with different configurations of the assembly parts, where the Low-level planner also considers the two different user preferences, were carried out (see **Table 2**). An assembly process trial is defined successful if the system was able to 'pick', to 'place', to 'present' and to 'handover' objects appropriately for the user and without collisions. In case of failures, as shown in experiments 2 and 6, the Low-level planner communicates the failure to the High-level planner to enable re-planning accordingly. For these experiments the feedback loop to the High-level planner was not considered, as recovering from failure was not in the focus of this evaluation.

Table 2. Evaluation table of the cognitive architecture performance, in dealing with the 'where' and 'how' of the assembly process (considering human preferences). Experiments (1-5) in grey, were performed considering the human preference 'present heater left'. Experiments (6-10) with human preference 'present heater right'

Experiment	'Picking'	'Showing'	'Placing'	'Handover'	'Success'
1	Yes	yes	yes	yes	yes
2	Yes	yes	yes	no	no*
3	Yes	yes	yes	yes	yes
4	Yes	yes	yes	yes	yes
5	Yes	yes	yes	yes	yes
6	Yes	yes	no	no	no**
7	Yes	yes	yes	yes	yes
8	Yes	yes	yes	yes	yes
9	Yes	yes	yes	yes	yes
10	Yes	yes	yes	yes	yes

* Robot failed to grasp the *Tray* object ('handover' step) due to a pose estimation error.
** Robot failed to 'place' the '*Heater-Base*' object due to a colliding movement path.

6. Conclusion and Future work

Human-robot collaboration in industrial environments is gaining lot of attention to improve flexibility in production. Conventionally, robotic systems were used in highly automated production scenarios behind closed fences to guard humans. The recent developments in 'collaborative robotics' [10], are allowing these robotic systems to break the fences, and move towards working hand in hand with humans.

In this paper we presented an integrated cognitive, modular system architecture for a robotic system collaborating with a human operator to complete an assembly task. The architecture combines state of the art object tracking, action recognition and path planning approaches together with a knowledge representation framework to perceive, reason, plan and execute an assembly process in a human-robot collaboration scenario. By conducting several real-world experiments, we evaluated the ability of the architecture (Low-level planner) in answering the questions 'where' and 'how', by also considering varying human preferences in a dynamic environment.

As future work, we plan to extend the reasoning capabilities to learn/classify previously unknown assembly task states and events during runtime. These extensions can help us also in recovering from failures. In case of human-robot collaboration one could see two forms of failures. One which is caused due to the robot's action execution (grasp, reach failures) and the other concerns with deviations due to unexpected human behavior. For a more dynamic management of human preferences, we plan to extend the human-machine-interface to enable the human operator to specify preferences or to

accept/reject proposals from the cognitive architecture on how to proceed the assembly process.

Acknowledgement

This research is funded by the projects KoMoProd (Austrian Bundesministerium für Verkehr, Innovation und Technologie), SIAM (FFG, 849971) and CompleteMe (FFG, 849441).

References

[1] A. Pandey, M. Ali and R. Alami, "Towards a Task-Aware Proactive Sociable Robot Based on Multi-state Perspective-Taking," *International Journal of Social Robotics*, vol. 5, no. 2, pp. 215-236, 2013.
[2] B. Scassellati, "Theory of mind for a humanoid robot," *Autonomous Robots*, vol. 12, no. 1, pp. 13-24, 2002.
[3] euRobotics aisbl, Robitcs 2020 Strategic Research Agenda for Robotics in Europe 2014 - 2020, 2013.
[4] J. Gray, & C. Breazeal,"Manipulating Mental States Through Physical Action," *International Journal of Social Robotics*, vol 6, pp. 315-327, 2014.
[5] A. Ioan et al., "The Open Motion Planning Library," *IEEE Robotics & Automation Magazine*, vol 19, no 4, pp. 72–82, 2012.
[6] J. A. Corrales Ramon et.al., "Cooperative tasks between humans and robots in industrial environments," *Int. Journal of Adv. Robotic Sys.*, vol. 9, no. 94, pp. 1-10, 2012.
[7] J. G. Trafton et al., "Enabling Effective Human-Robot Interaction Using Perspective-Taking in Robots," *IEEE Trans. System, Man, and Cybernetics - Part A Syst. Humans*, vol. 35, no. 4, pp. 460–470, 2005.
[8] J. Mainprice and D. Berenson, "Human-robot collaborative manipulation planning using early prediction of human motion," *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems,* Tokyo, pp. 299-306, 2013.
[9] K. Dautenhahn, "Methodology and themes of human-robot interaction: a growing research field," *Int. Journal of Adv. Robotic Sys.*, vol. 4, no. 1, pp. 103-108, 2007.
[10] L. L. P. PricewaterhouseCoopers, "The new hire: How a new generation of robots is transforming manufacturing," 2014.
[11] L. Kunze, T. Roehm and M. Beetz, "Towards semantic robot description languages," *in Proc. IEEE Int. Conf. on Robotics and Automation (ICRA)*, Shanghai, pp. 5589-5595, 2011.
[12] M. A. Goodrich and A. C. Schultz, "Human-robot interaction: a survey," *Foundations and trends in human-computer interaction*, vol. 1, no. 3, pp. 203-275, 2007.
[13] M. Tenorth and M. Beetz, "KnowRob: A knowledge processing infrastructure for cognition enabled robots," *The International Journal of Robotics Research*, vol 32, pp 566-590, 2013.
[14] S. C. Akkaladevi and C. Heindl, "Action recognition for human robot interaction in industrial applications," *in Proc. IEEE Int. Conf. Computer Graphics, Vision and Inf. Security (CGVIS)*, pp. 94-99, 2015.
[15] S. Akkaladevi, M. Ankerl et al., "Tracking multiple rigid symmetric and non-symmetric objects in real-time using depth data," *in Proc. IEEE Int. Conf. on Robotics and Automation (ICRA)*, pp. 5644-5649, 2016.
[16] S. Albrecht, "Implicit human computer interaction through context," *Personal technologies*, vol 4, no .2-3, pp. 191-199, 2000.
[17] O. Schrempf et al., "A novel approach to proactive human-robot cooperation," *IEEE Int. Symp. Robot and Human Interactive Communication (Ro MAN)*, pp 555–560, 2005.
[18] S. Lee et.al, "Human Mental Models of Humanoid Robots," *Proc. of the 2005 IEEE Int. Conf. Robotics and Automation*, pp. 2767-2772,2005.
[19] SCHUNK GmbH & Co. KG, "Schunk PG70 Parallel Gripper," [Online]: http://us.schunk.com/us_en/gripping-systems/#/product/2493-0306095-pg-70, last accessed: 09.05.2016, 2016.
[20] T. Fischer and Y. Demiris, "Markerless Perspective Taking for Humanoid Robots in Unconstrained Environments," *in Proc. IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2016.
[21] Universal Robots A/S, "UR10 robot - A collaborative industrial robot," [Online]: http://www.universal-robots.com/products/ur10-robot/, last accessed: 09.05.2016, 2016.
[22] Y. Shen, "System fur die Mensch-Roboter-Koexistenz in der Fliessmontage (Forschungsberichte / IWB 305), " Munich, 2015.