Cognitive Semantics For Dynamic Planning In Human-Robot Teams

Alexandre Angleraud[1], Quentin Houbre[1], Metodi Netzev [1] and Roel Pieters[1]

***Abstract*— Robots are making their way into our society and are foreseen to become an important part in our everyday life, at work or at home. Industrial factory layouts are moving robots out of enclosures bringing them side by side with human workers. As for service robots they are by definition meant to perform tasks in our immediate proximity. To be performed successfully, these tasks, also referred to as joint actions, require coordination and trust. Coordination implies that the robot needs to account for his actions and their effects on the environment but also for changes that the user introduces. Therefore, flexible planning capacities allowing on-the-fly adaptation to what a human is requesting or doing, together with a shared mental representation of the task, are needed. In this paper we present (i) a symbolic knowledge system and the way it translates into simple temporal networks (STN) to generate actions plans, and (ii) interaction models based on natural language. First results indicate the robot can build plans for a joint action according to several parameters given its conceptual semantic description. Furthermore, a human can interactively either modify the plan or ask for explanations about it. By several experiments we demonstrate the generation and adaptation of these dynamic human-robot collaboration plans.**

I. INTRODUCTION

A traditional industry floor consists of distinct areas where robots and humans cannot interact for safety reasons [1]. However, the current trends ask industries to show more and more versatility as opposed to mass production. Subsequently, the need for more intelligent machines capable of performing tasks together with a human is equally increasing [2]. Similarly, using robots as personal assistants in our homes or workplaces points to developing machines capable of taking human actions into account when reasoning about the world and the tasks to be carried out.

We believe that those new trends are intertwined with our understanding of Cognitive Semantics (CS). CS is part of the Cognitive Linguistics (CL) field of research. Its main underlying assumptions are as follows: 1. Language is not an isolated cognitive mechanism, i.e., it supports the idea that rather than having a module handling the knowledge and another one handling language production: 2. Grammar is conceptual, which compares language to a symbolic system and links to the study of semiotics and the emergence of coordination from the aptitude to communicate semantic information through a symbol system. [3] 3. Learning the language actually comes through using it [4].

Previous results showed that human teams rely heavily on communication to establish shared plans [5] and react to changes while performing collaborative tasks. In our preceding work we addressed the problem of maintaining a conceptual model about tasks and explored interactive processes allowing to solve ambiguities in the given commands [6]. We also showed how it can be used by a human user to teach new knowledge (see Fig. I).

[1]Department of Automation Technology and Mechanical Engineering, Cognitive Robotics, Tampere University, 33720, Tampere, Finland; `firstname.surname@tuni.fi`

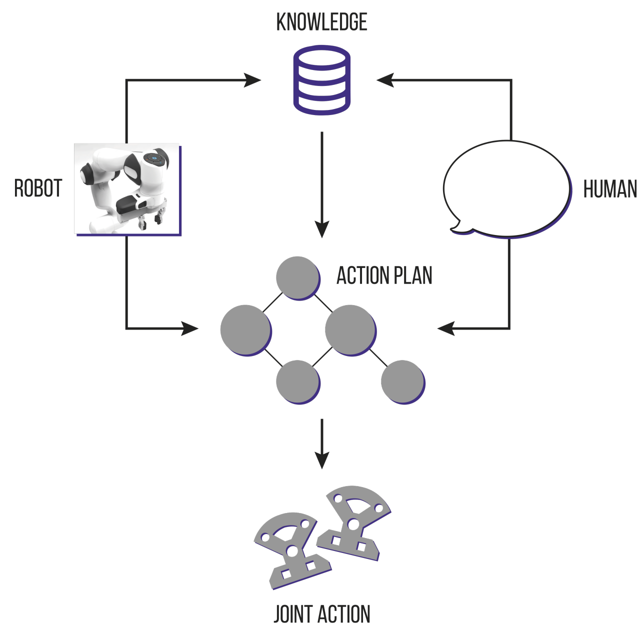

Fig. 1. Robot and human share a symbolic representation of a task. As a result human can teach new skills to the robot (via a web interface). This knowledge is then transformed into an action plan using the Simple Temporal Network (STN) formalism. While the action is being performed the human can still interactively modify the plan using natural language.

As continuation, in this paper we introduce a mechanism to translate the knowledge automatically into a form suitable for task planning. Additionally, we present dialog patterns allowing the user to access and modify the plan on-the-fly. The following section will compare those features with the most similar systems in the literature and through this comparison stress the problem addressed in this paper. Section III will discuss the model adopted to represent the robot's skills and its translation into an action plan as well as the different policies that can be used to do so. Following, the mechanisms using the semantic information about the plan under their symbolic forms are discussed in Section IV. Finally Section V takes as an example an industrial benchmark in which human and robot can collaborate (but also act independently) to perform an assembly task.

II. RELATED WORK

The study of joint actions is not new and has deep roots in the cognitive sciences field. The underlying cognitive mechanisms taking place in the process such as joint attention, action observation, task-sharing, action coordination, and perception of agency have been identified as core elements of cooperation [7]. Implementations of those concepts usually evaluate the ability of the robot to reason about the task to perform in a cooperative fashion and its reactivity, should the human fail to carry out a step [8]. Additionally it has been discussed whether cooperation should be seen as an atomic concept or if it could be broken down into sub concepts for researchers to explore. For instance, by utilizing the concepts of planned and emergent coordination [9].

In this paper, planned coordination is addressed. This type of behavior implies knowledge about the different entities involved in the tasks to perform and brings the semantics problem to the equation. In order to successfully coordinate its actions with a human, not only does the robot need to reason but also communicate about its perception of the task. It is convenient to use a model close to our understanding based on a symbolic system [3], [10] that builds upon semantic networks [11]. Moreover, as robots are physical agents, designed to interact with humans, their knowledge needs to be grounded in the real world in such a way that they are able to bridge their perceptual inputs [12].

Reasoning thereafter lifts the interaction to achieve as natural collaboration as possible [13]. Borrowing again concepts from cognitive sciences, a Shared Cooperative Activity (SCA) can also be seen as a mechanism in which agents dynamically mesh their sub-plans together [14]. In other words, this paradigm argues that a shared plan cannot be reduced to the sum of individual plans of the agents involved. Following this idea, collaboration then exists as a way to reach a common goal and natural language plays an important role as it allows to share beliefs and intentions between agents.

Natural language can also be used to offer an intuitive programming interface [15] after a task has been taught to the robot by demonstrating it. All in all it appears to be a way to bring humans and robots closer by sharing information in the most natural way for us and thus facilitate integrating robots in our society [16]. To ensure mutual understanding is therefore an important aspect of communication. This problem transposes to robots that need to be able to explain their choices when asked about it [17]. Ultimately being capable of such transparency is getting closer to the hope that we could trust our robotic partners [18].

III. KNOWLEDGE, BELIEFS AND INTENTION MODELS

This section describes the semantic representation of a skill and its equivalent in the task planning space before discussing ways to share models between agents.

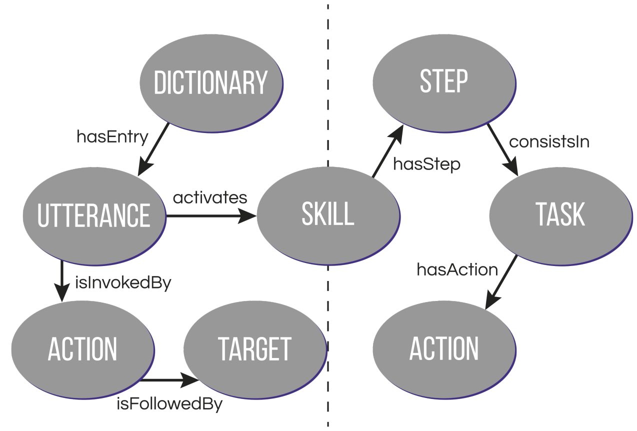

Fig. 2. High level ontology used to represent knowledge in a symbolic fashion. On the left the structure to handle the semantic event chain related to a skill. On the right the structure to handle the communication from human to robot about it.

A. Semantic skill representation

At the root of the knowledge representations for tasks and robot capabilities lies Object Action Complexes (OACs) [19]. They were introduced to overcome the differences between high level representations for cognitive reasoning and lower level representations used when controlling robots. We generalized this representation by wrapping it around the broader concepts of tasks and steps (see Fig. 2)

Doing so allows us to describe the Semantic Event Chain (SEC) [20] of the skills available on the robot.

B. Simple temporal network

We review here the main concepts related to Simple Temporal Networks (STN). An STN is a graph $\mathcal{S}$ in which each node represent a timepoint and the edges are temporal constraints between them [21].

$$\mathcal{S} = (\mathcal{T}, \mathcal{C})$$

Where $\mathcal{T}$ is a set of time-point variables : $\{t_1, ..., t_n\}$ and $\mathcal{C}$ temporal constraints : $t_j - t_i \leq \delta$ with δ being a real number.

When constructing the graph it is convenient to take a zero point to serve as a reference when expressing the constraints. A solution to an STN is a full assignments to the timepoints in $\mathcal{T}$: $\{t_1 = w_1, t_2 = w_2, ..., t_n = w_n\}$

The traditional algorithm to solve an STN involves building the distance matrix of the network. The constraints that are intervals are rewritten as a pair of directed edges such that the original one takes the value of the upper bound and the second one the opposite of the lower bound. The All Pairs Shortest Paths (APSP) form obtained using for example the Floyd-Warshall Algorithm [22] is further necessary to transform the graph into a dispatchable form [23]. After this step the STN is ready to be solved, but to make execution more efficient pre computations are often made to lower the cost of updating the distance matrix.

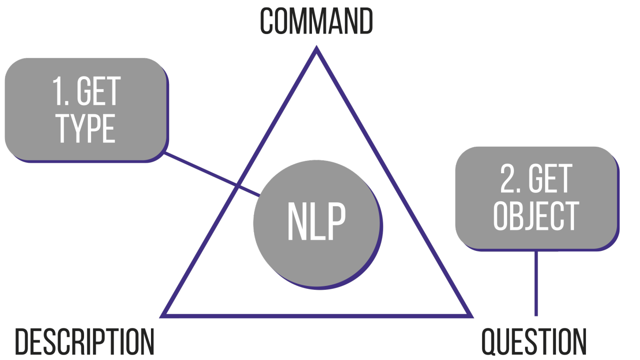

Fig. 3. Parsing a sentence occurs in two steps. First the robot determines the type of the sentence. It can be a command, a description or a question. Then specific parsing rules are applied respectively to decide the subject of the sentence.

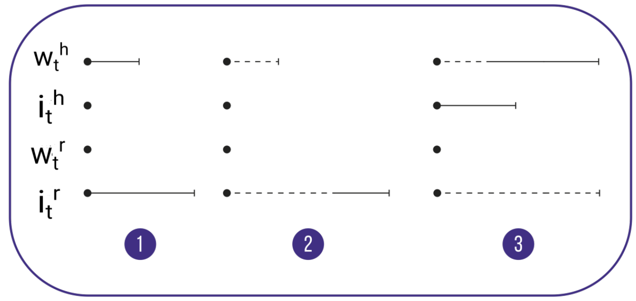

Fig. 4. Beginning of the timeline from a simulation example. 1. Human and robot both start performing an action. The human is a bit faster than the robot 2. The human however has to wait for the robot to complete its next action. It will thus enter a waiting state. 3. When the robot is done, work and idle times are updated. The human leaves the waiting state to get a chance to reevaluate his available steps. The execution then starts again normally.

Since their first apparition in 1990, STNs have been extended to gain more and more flexibility [24]. We build upon those using a pair containing a base solution and a set of differences. The base solution consists in a relaxed form of the network while the set of differences holds the various effects that choosing a certain step over another step can have.

C. Model sharing

To be successful the two agents need to be coordinated. Language is an important factor to achieve coordination in human teams [14].

1) Dialog patterns: We explored a two steps parsing (see Fig. 3) in which the type of an utterance is first matched against predefined patterns. To each type is then associated a certain number of heuristics that will infer the subject of the utterance. Questions, description and commands are the available types. Questions can be about the location of an object for instance. Locations that could have been taught through a description. Descriptions also serve to transmit the information about the completion of an action. Lastly, commands are the (action, target) pairs used to trigger the robot's skills.

2) Beliefs about the human: Throughout the interactions, information will be gathered to build a certain model of the human. The knowledge base will attribute a context to any new agent known by the robot. These contexts allow to obtain perspective taking, at least to some extent by being able to maintain distinct states for the same entities withing the same reasoning module.

IV. DYNAMIC AND INTERACTIVE PLANNING

This section describes the different planning policies explored and the mechanism allowing a human to use natural language to modify the task plan during its execution.

A. Planning policies

The initial model includes disjunctions with binary constraints over the robot and the human performances. It thus needs to be projected to come back to a regular STN. In other words it is the moment tasks are attributed to an agent and it can be done following various policies. The distribution relies on a simulation of the task execution (see Fig. 4). Utilizing the worst case scenario times for each action results in simulation output that specifies the working times of human and robot respectively together with their idle times (see Algorithm 1).

while *not all timepoints stamped with execution time* **do**
  if *robot time ahead and not human waiting* **then**
    if *has available task* **then**
      human working_time = simulate_step(step);
      update_list_steps;
      if *robot is waiting* **then**
        robot_waiting = False;
        update_robot_idle_time
    else
      human_waiting = True
  else if *human time ahead and not robot waiting* **then**
    if *has available task* **then**
      robot working_time = simulate_step(step);
      update_list_steps;
      if *human is waiting* **then**
        human_waiting = False;
        update_human_idle_time
    else
      robot_waiting = True
end

Algorithm 1: Simulation algorithm calculating the working and idle times of the human and the robot

The algorithm runs a simulation of the skill until full completion when all the nodes of the plan are marked as done. It uses the simulated working timelines of the robot and the human, trying to attribute a task to the least advanced in time given that this agent has available steps in the current distribution being evaluated. Otherwise the current agent enters a waiting state. When the partner completes his next step, it changes back the state of the first worker to active to reevaluate the available steps at the next iteration. Doing so, the idle time for the waiting agent is updated by the difference between the working times of the two agents and the working time is set to match the working time of the teammate.

Also summarized in Table I, the task distribution can be done according to the following policies:

1) Even distribution: Attribute tasks aiming at having even working times :

$$\arg \min_{work_i, work_j} (\|\Sigma work_i^r - \Sigma work_j^h\|)$$

2) Capacity based: Attribute tasks aiming at minimizing a working time while keeping idle time as low as possible:

$$\arg \min_{work_i, work_j} (\Sigma work_i^r + \Sigma work_j^h)$$

3) Activity based: Attribute tasks aiming at minimizing an idle time while keeping working time as low as possible:

$$\arg \min_{work_i, work_j} (\Sigma idle_i^r + \Sigma idle_j^h),$$

where in each case superscript r and h denotes robot and human, respectively.

B. Regular execution

The overall process governing the excution of a task is explained in Algorithm 2:

while *not all timepoints stamped with execution time* **do**
  perform one;
  if *success* **then**
    mark as done;
    update links;
  else
    warn human;
  end
end

Algorithm 2: Execution to reach a solution for the STN

When it starts, the algorithm computes all the valid plans for the task, picks one and start executing it. In parallel the human is also going to perform actions and can warn the robot by telling it after completing one. Every time the list of available plans is updated to take into account only the temporally consistent distributions matching the past events. If the robot cannot perform any action it will explicitly inform the human that he or she needs to complete something before the robot can move on.

C. Interactive plan modifications

Using the dialog patterns previously described, the user can, while performing a task, interrupt the current execution and attribute a task to the robot or adopt a task it had initially planned to carry out.

TABLE I

PLANNING POLICIES AND THEIR EXPECTED INFLUENCE ON THE TIME VARIABLES OF THE TASK EXECUTION.

Policy	Optimization target	Expected result
Balanced distribution	$work_t^r, work_t^h$	Equal amount of time working regardless of the time spent inactive.
Capacity based	$work_t^r, work_t^h$	Low working time but the idle time might increase
Activity based	$idle_t^r, idle_t^h$	Low idle time but individual working time might be longer

1) Trigger plan modification: A plan can be modified in two ways. The human can decide to carry out a step that was initially planned for the robot and vice versa. Modifying the plan in reality means imposing a constraint on the task distribution. The robot then updates the list of valid plans similarly to the update occurring after completing a task, retaining only the alternatives matching with the human request.

2) Backpropagation rules: Given that the graph is in a dispatchable form, modifications need only to be propagated to neighbouring timepoints and can be resolved using triangular reductions. Applying these rules after the completion of a task allows to check the consistency of a plan by taking into account the real execution time of the timepoints in the STN. We implemented two rules also refered to as the precede and the follow case [25].

V. PRACTICAL SCENARIO

As assembly task we consider the Cranfield Assembly Benchmark [26], which is composed of nine steps. This task is interesting because the steps are not sequential. Some can be done independently while others require previous steps to be executed (see Fig. 5). In particular, steps 1-2, are the assembly of the round small pegs in the holes of the back faceplate, steps 4-5 are the assembly of the square ones, while step 3 brings the shaft. Step 6 is the assembly of the pendulum on top of the shaft and step 7 completes it with the pendulum head. Finally in step 8 the separator is assembled on top of the square pegs before the front plate can be adjusted as step 9. Regarding our implementations, all our developments are available open source for the robotics community[1].

Running the tests for our three policies leads to the task plans depicted in Fig. 6. Fig. 7 describes a successful plan execution for a balanced plan, where the human interrupts the plan and changes task allocation step 9 from human to robot. In addition, the means of work and idle times for the human and robot, respectively, are collected in Table II according to the policy used to plan for the task. They were calculated

[1]https://zorrander.gitlab.io/franka-web-app/

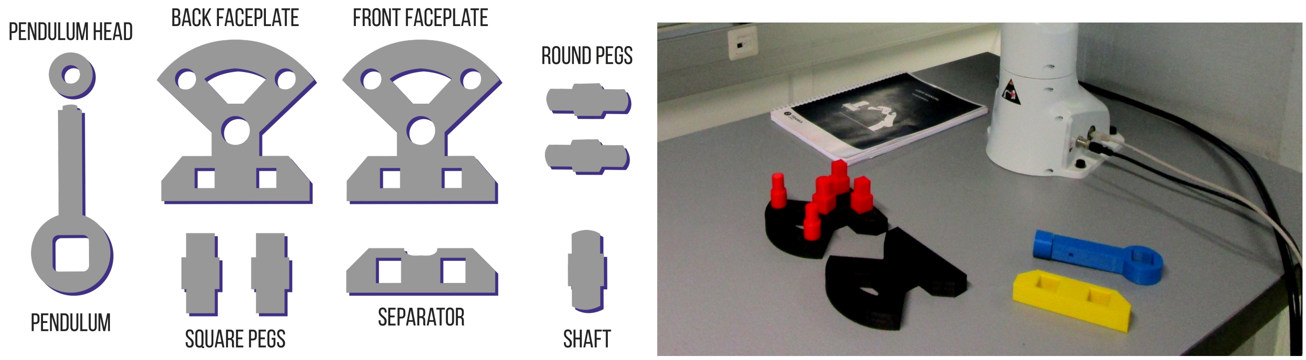

Fig. 5. Parts involved in the Cranfield assembly benchmark. Left figure shows the descriptive shapes for each part and their names. The assembly steps relate to the parts as follows. Steps 1-2: assembly of the round pegs in the holes of the back faceplate, steps 4-5: assembly of the square pegs in the back faceplate, step 3: assembly of the shaft in the back faceplate, step 6: assembly of the pendulum on top of the shaft, step 7: assembly of the pendulum head on the pendulum, step 8: assembly of the separator on top of the square pegs, Step 9: assembly of the front plate on the pre-assembled part. Right figure shows the real parts manufactured by 3D printing, next to a robot.

TABLE II

WORK AND IDLE TIME MEANS FOR THE HUMAN AND THE ROBOT GIVEN A PLANNING POLICY.

Time group	Balanced policy [sec]	Capacity policy [sec]	Activity policy [sec]
$work_t^h$	160	118	119
$idle_t^h$	73	9	12
$work_t^h$	160	121	121
$idle_t^r$	19	14	12

using default times of 20 seconds for a human and 30 seconds for the robot. These are indicative times obtained by trials; typically, the human is slightly faster in the assembly tasks. This shows that the balanced policy offered less performance with regard to both work and idle times. Nonetheless it seems like the equilibrium is maintained when optimizing for lower working or idling times. Regarding these two cases the first results would suggests that with the current implementation they achieve similar performance.

VI. DISCUSSION AND FUTURE WORK

The work presented here relies blindly on the correct statements from a human to assist in an assembly task. If the human utters a wrong statement a part, a task would need to be restarted. One solution to this is to take visual detection into this task planning and only rely on spoken language when the task plan is changed. This is motivated by the positive experience in experiments, where language proved to be useful for fast and reliable interaction [6].

On the other hand, language processing is complex and can not include all possible requests a human could perform. However, a combination of dialog patterns and even simplistic nlp algorithms seems viable, in particular in industrial settings where the environment and the tasks are predictable.

Deciding on a planning policy depends on many factors, that are often out of control for a human operator. Adopting the fastest policy for execution is not always the best option and changing policies might not even be in the best interest of the task. In our view, a solution to this should be sought per case, therefore, future work will explore how properties of an operator and a robot can be taken into account in the planning policies. For example, knowing that a robot should do the heavy lifting and a human should do the fine compliant assembly effects a shared task plan considerably [2]. Automated reasoning to achieve such a plan can, again, be taken into account by a knowledge and reasoning system.

VII. CONCLUSION

In this work we propose (i) a symbolic knowledge system that translates a task into simple temporal networks (STN) to generate actions plans, and (ii) an interaction model based on natural language resulting in direct modifications of the previously created action plan. This action plan generation and action plan adaption is integrated in our conceptual knowledge base, where users can define new concepts for robots in an intuitive way. Action planning utilizes a language model based on object-action complexes (OACs) that have also proven themselves useful when solving ambiguities arising from commands given using natural language. The system is demonstrated using a benchmark for an assembly task in which human and robot can collaborate closely but also act individually at times. By defining three planning policies (balanced, capacity and activity) we demonstrate how the robot uses task knowledge to generate and adapt plans on-the-fly automatically.

REFERENCES

[1] R.-J. Halme, M. Lanz, J. Kämäräinen, R. Pieters, J. Latokartano, and A. Hietanen, "Review of vision-based safety systems for human-robot collaboration," *Procedia CIRP*, vol. 72, no. 1, pp. 111–116, 2018.

[2] A. Hietanen, R.-J. Halme, J. Latokartano, R. Pieters, M. Lanz, and J.-K. Kämäräinen, "Depth-sensor-projector safety model for human-robot collaboration," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS) Workshop on Robotic Co-workers 4.0*, Madrid, Spain, 2018. [Online]. Available: http://vision.cs.tut.fi/data/publications/iros2018ws.pdf

[3] T. Taniguchi, T. Nagai, T. Nakamura, N. Iwahashi, T. Ogata, and H. Asoh, "Symbol emergence in robotics: a survey," *Advanced Robotics*, vol. 30, no. 11-12, pp. 706–728, 2016.

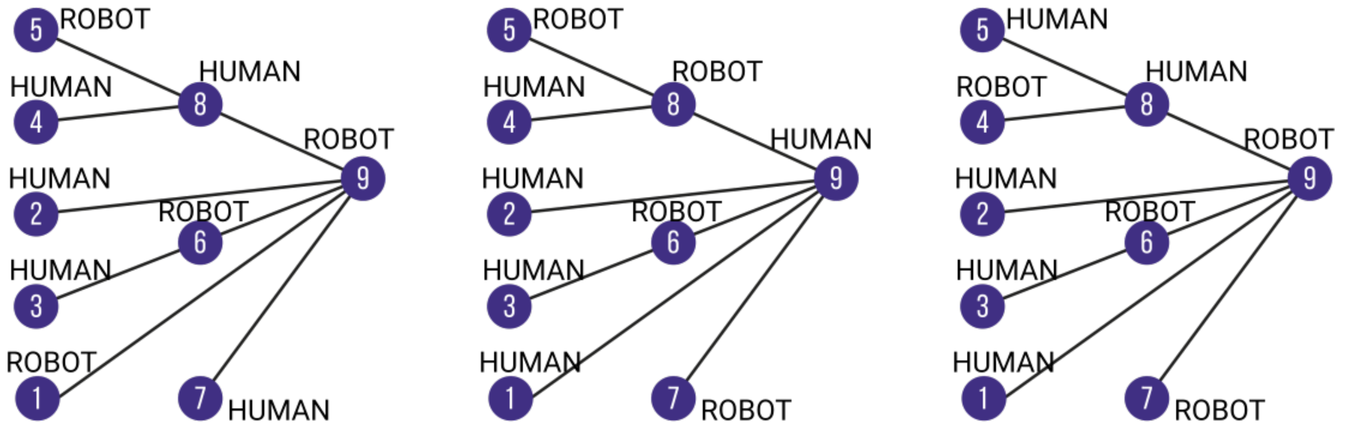

Fig. 6. Example assembly plans for the Cranfield assembly benchmark. Left: Initial plan for the balanced policy (Equal amount of time working regardless of the time spent inactive). Middle: Initial plans according to the capacity based policy (low working time but idle time might increase). Right: Initial plans according to the activity based policy (Low idle time but individual working time might be longer).

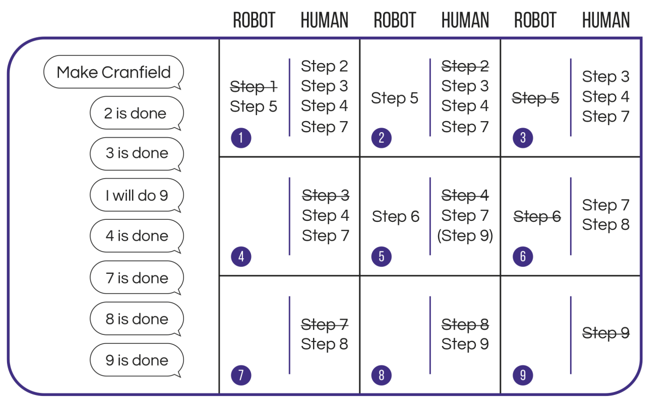

Fig. 7. Plan execution for the initial balanced plan (Fig. 6-left). On the left are the spoken utterances from the human indicating either the completion of a step or a change in the plan. On the right is the status of the plan, indicating the available steps at different timepoints during the plan. During task execution the human changes the plan (step 9 executed by the human) which is taken into account by the symbolic knowledge system and planner.

[4] W. Croft and D. A. Cruse, *Cognitive Linguistics*, ser. Cambridge Textbooks in Linguistics. Cambridge University Press, 2004.

[5] P. R. Cohen and H. J. Levesque, "Confirmations and joint action." in *IJCAI*, 1991, pp. 951–959.

[6] A. Angleraud, Q. Houbre, V. Kyrki, and R. Pieters, "Human-robot interactive learning architecture using ontologies and symbol manipulation," in *Proceedings of the 27th IEEE International Symposium on Robot and Human Interactive Communication*, Nanjing, China, 2018, pp. 384–389.

[7] B. Mutlu, A. Terrell, and C.-M. Huang, "Coordination mechanisms in human-robot collaboration," pp. 1–6, 2013.

[8] P. F. Dominey and F. Warneken, "The basis of shared intentions in human and robot cognition," *New Ideas in Psychology*, vol. 29, no. 3, pp. 260–274, 2011.

[9] G. Knoblich, S. Butterfill, and N. Sebanz, "Psychological Research on Joint Action," in *Psychology of Learning and Motivation*. Elsevier, 2011, vol. 54, pp. 59–101.

[10] D. Beßler, M. Pomarlan, and M. Beetz, "Owl-enabled assembly planning for robotic agents," in *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*, 2018, pp. 1684–1692.

[11] H. Helbig, *Knowledge representation and the semantics of natural language*, ser. Cognitive technologies. Berlin ; New York: Springer, 2006.

[12] N. Mavridis and D. Roy, "Grounded Situation Models for Robots: Where words and percepts meet," in *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Beijing, China, 2006, pp. 4690–4697.

[13] S. Lemaignan, M. Warnier, E. A. Sisbot, A. Clodic, and R. Alami, "Artificial cognition for social humanrobot interaction: An implementation," *Artificial Intelligence*, vol. 247, pp. 45–69, 2017.

[14] C. Breazeal, G. Hoffman, and A. L. Thomaz, "Teaching and working with robots as a collaboration," *Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems, 2004.*, pp. 1030–1037, 2004.

[15] W. Wang, R. Li, Y. Chen, Z. M. Diekel, and Y. Jia, "Facilitating Human-Robot Collaborative Tasks by Teaching-Learning-Collaboration From Human Demonstrations," *IEEE Transactions on Automation Science and Engineering*, pp. 1–14, 2018.

[16] A. Sciutti, M. Mara, V. Tagliasco, and G. Sandini, "Humanizing Human-Robot Interaction: On the Importance of Mutual Understanding," *IEEE Technology and Society Magazine*, vol. 37, no. 1, pp. 22–29, 2018.

[17] B. Hayes and J. A. Shah, "Improving robot controller transparency through autonomous policy explanation," in *Proceedings of the 2017 ACM/IEEE International Conference on Human-Robot Interaction*, New York, NY, USA, 2017, pp. 303–312.

[18] S. H. Huang, K. Bhatia, P. Abbeel, and A. D. Dragan, "Establishing appropriate trust via critical states," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 3929–3936.

[19] N. Krüger, C. Geib, J. Piater, R. Petrick, M. Steedman, F. Wörgötter, A. Ude, T. Asfour, D. Kraft, D. Omrčen, *et al.*, "Object–action complexes: Grounded abstractions of sensory–motor processes," *Robotics and Autonomous Systems*, vol. 59, no. 10, pp. 740–757, 2011.

[20] E. E. Aksoy, A. Abramov, J. Dörr, K. Ning, B. Dellen, and F. Wörgötter, "Learning the semantics of object–action relations by observation," *The International Journal of Robotics Research*, vol. 30, no. 10, pp. 1229–1249, 2011.

[21] R. Dechter, I. Meiri, and J. Pearl, "Temporal constraint networks," *Artificial Intelligence*, vol. 49, no. 1, pp. 61 – 95, 1991.

[22] T. Cormen, C. Leiserson, and R. Rivest, *Introduction to Algorithms*, 3rd ed. MIT Press, 1990, ch. All-Pairs Shortest Paths, pp. 684–707.

[23] N. Muscettola, P. H. Morris, and I. Tsamardinos, "Reformulating temporal plans for efficient execution," in *Proceedings of the Sixth International Conference on Principles of Knowledge Representation and Reasoning*, 1998, pp. 444–452.

[24] J. A. Shah, "Fluid coordination of human-robot teams," Ph.D. dissertation, Massachusetts Institute of Technology, 2011.

[25] P. Morris, N. Muscettola, and T. Vidal, "Dynamic control of plans with temporal uncertainty," 2001.

[26] T. R. Savarimuthu *et al.*, "Teaching a robot the semantics of assembly tasks," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 48, no. 5, pp. 670–692, 2018.