



HAL
open science

Adapting Cobot Behavior to Human Task Ordering Variability for Assembly Tasks

Belal Hmedan, Dorilys Kilgus, Humbert Fiorino, Aurelie Landry, Damien
Pellier

► **To cite this version:**

Belal Hmedan, Dorilys Kilgus, Humbert Fiorino, Aurelie Landry, Damien Pellier. Adapting Cobot Behavior to Human Task Ordering Variability for Assembly Tasks. FLAIRS-35, May 2022, Jensen Beach, United States. 10.32473/flairs.v35i.130645 . hal-03662945

HAL Id: hal-03662945

<https://hal.science/hal-03662945>

Submitted on 8 Jun 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Adapting Cobot Behavior to Human Task Ordering Variability for Assembly Tasks

Belal Hmedan, Dorilys Kilgus, Humbert Fiorino, Aurélie Landry and Damien Pellier

Univ. Grenoble Alpes – LIG
Grenoble, France

{firstname.secondname}@univ-grenoble-alpes.fr

Abstract

In this paper, we address flexible assembly plans generation to accommodate human task variability. Unlike existing approaches, the proposed approach is based on a *free-style* human-robot interaction (HRI) that does not impose any task order on the participants and furthermore can accommodate their errors and help them to correct the errors without stopping the whole assembly process. Our approach is implemented in a real robotic architecture that combines sensory-motor modules with Hierarchical Task Networks (HTN) to endow the cobot with the necessary adaptability to correspond to human actions dynamically. We show experimentally with 56 participants on a simulated industrial assembly task that the cobot increases the task performance (reduction in the number of errors and of gestures) without increasing the participants' cognitive load.

Introduction

Manufacturing assembly automation is a permanent challenge for the industry as innovation is a real lever in order to keep production performance in a worldwide economic market. However, this automation has obliged the skilled operators, who cannot be completely substituted by robots, to follow specific operation paradigms to cope with the robots' operation procedures. This type of constrained operation increases the cognitive load of the operators [11, 12], reduces the room for maneuver in their task performance and causes musculoskeletal disorders [1, 23]. This is due to a lack of human variability consideration in the design of the automation process. It is therefore important to take into account human variability, e.g., level of experience or fatigue, size, etc. very early in the robotic architecture design, and to endow the robot with the ability to adapt to human beings in order to prevent these risks.

Among the human variabilities that must be taken into account, *human task variability* is one of the most important. It is a question of being able for a robot to adapt the order in which it performs the tasks to the order chosen by a human and not the other way around. The aim is to give the human a great deal of freedom in carrying out a collaborative task with a robot and not to constrain him/her more than necessary and to alleviate his/her cognitive load by not imposing

specific task orders, leaving the human free to delegate more or fewer tasks to the robot.

Accommodating robot's behaviors to human task variability is a challenging issue due to the uncertainty in human behavior [4] and the probability of conflicts between human and cobot actions [2]. To deal with this issue, we propose in this paper a robotic architecture bridging sensory and action elements with task planning reasoning capabilities [19] implemented on the cobot YuMi to ensure a smooth workflow in collaborative tasks where humans and cobots work together in close vicinity and share the same environment. Unlike other approaches e.g., [4, 5, 18, 21], we do not impose any order on the operator. Moreover, the robot and the operator can simultaneously act in the same area to accomplish a shared assembly task.

Our contribution in this paper is twofold: (1) a *free-style* human-robot interaction (HRI) that does not impose the task-order on the participants and furthermore can accommodate their errors and help them to correct the errors without stopping the whole assembly process, and (2) an extensive evaluation methodology that highlights the benefits of integrating a cobot into the assembly process using not only a classical robotics metric to evaluate task performance but also metrics from ergonomics. This methodology handles the problem from different aspects, including the subjective cognitive load measured through two phases with and without the cobot, of experiments performed with 56 participants.

The rest of the paper is organized as follows: Section II presents the related work; section III presents the architecture proposed; Finally, section IV introduces the experimental setup and the results showing that our approach reduces the cognitive load of the operators and increases the task performance.

RELATED WORK

Robotic architecture bridging sensory and action elements with task planning reasoning capabilities in the context of industrial human-robot collaboration was discussed mainly from the performance point of view, where the time of planning and scheduling is the main criteria for evaluation [4, 8, 10, 24]. In contrast, human aspects such as acceptability, usability and cognitive load were less discussed [2, 5, 21]. Furthermore, most works limit the freedom of ac-

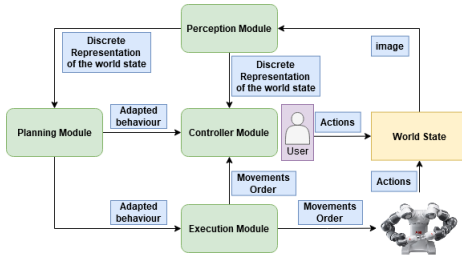


Figure 1: The proposed modular architecture

tion of humans in the execution of their task, for example by assigning specific tasks to humans [5, 8, 21], or limiting their choices in terms of assembly zones and assembled block colors [4].

To relax these constraints it is necessary to grant the human a higher degree of liberty to make the interaction with the cobot smooth, natural, and intuitive. Hence, it is better not to impose task order on the human, but instead, adapt the task plans with respect to the human actions. A number of works have tried to deal with the problem of adapting a robot’s behavior to the order of execution of a task by a human and many techniques have been proposed to generate behaviors more or less dynamically. These are the techniques (in non-exhaustive list): Planning and Scheduling [4, 14, 15, 17], AND-OR Graphs [8], Behavior-Tree [18, 22], Hierarchical Task Networks (HTN) [2, 6, 13], Game Theory [5], Markov Decision Process (MDP), or a combination of them [21]. Among these techniques, task planning and more precisely hierarchical planning is widely used and very effective to generate symbolic plans in a similar way of the human decomposition of complex tasks into simpler actions [3].

Subsequently, the challenge of the tasks order adaption is tackled in this paper with a real industrial cobot YuMi to best imitate an industrial assembly task and not only on simulation [4, 14]. The robot’s behavior is made explicit to the human, through an interface so as to consolidate his/her understanding and his/her acceptability of its behavior [16]. An explanation of the cobot next action, and human errors are provided for error correction while the assembly keeps uninterrupted.

Adaptive Robotic Architecture

Our approach in tackling human task variability stands on a modular architecture consisting of four modules (Fig 1):

1. *Perception module* to perceive the workplace (including the human task variability) and encode the perceived information into a discrete representation.
2. *Planning module* to generate the adaptive behavior that corresponds to the human task variability.
3. *Execution module* to execute the intelligent behavior on the cobot.
4. *Controller module* to manage the interaction between the operator and the cobot, and explain the intelligent behav-

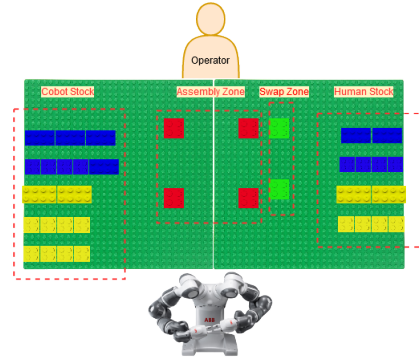


Figure 2: Workplace zones

ior to the operator in order to augment his/her social acceptance.

PERCEPTION MODULE

This module, using classical image processing techniques generates a discrete representation of the workplace composed of position and color information. This representation is a dictionary where the key is the discrete position encoded in row column indices, and the value is the color information, e.g., {13 : blue, 21 : green, 34 : yellow}.

The workplace in our scenario is divided into four zones (Fig 2): cobot stock zone, human stock zone, shared assembly zone, and swap zone where the cobot can provide blocks from its stock to the operator.

PLANNING MODULE

To adapt the cobot behavior to the human task variability we take advantage of the flexibility and the intelligence that Automated Planning (AP) can generate. Specifically, Hierarchical Task Networks (HTN) technique allows to compute action sequences i.e. plans with respect to given world states, and to provide explanations on the causal links of these actions at different levels of abstractions [16].

For this purpose we have developed a conceptual model composed of a *domain*, and a *problem* in the Hierarchical Domain Definition Language (HDDL) [9]. The planner used in our architecture is the Totally Ordered Fast Downward (TFD) Planner [20]. The domain in our model is static and encompasses the objects in our scenario and the relations (primitive actions) between them, as well as the higher-level tasks, and different methods to decompose the tasks into primitive actions according to triggering preconditions. On the other hand, the problem in our case is dynamic in the sense that the initial state is obtained from the discrete representation generated by the vision system, which varies along the cobot actions, and the operator variable actions (Fig 3). Re-planning is done after each perception phase sequentially for each block until a plan is found for a given block, otherwise if there is no plan found for all blocks, hence planning stops and the user is informed to do the task alone.

The predicates in HDDL define the status of each object, for instance the *gripper* position, orientation, fingers status

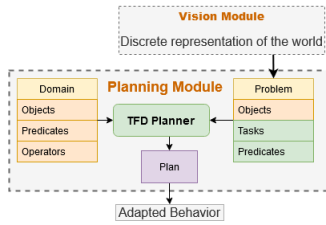


Figure 3: Planning module

(open/close), and whether it is loaded or not. Furthermore, the *points* are described in terms of occupancy and neighboring points. Finally, *blocks* orientation, gripping position, and whether the block is attached to the workspace or not. The tasks in our domain are limited to *Pick*, and *Place*, and the parameters are the gripper which will be used, the block to be picked/placed, and the direction of the placement:

```
(:task pick
  :parameters (?g - Gripper ?x - Block ?d -
    Direction))
(:task place
  :parameters (?g - Gripper ?x - Block ?pos
    - Point_Workspace))
```

These tasks are combined with the necessary primitive actions, and the hierarchical decomposition methods to provide a flexible way to execute the same task in various available approaches. An example of the actions done here is holding a block with the gripper, this action changes the status of the cobot fingers from open to close, and the gripper becomes loaded with the block:

```
(:action Hold_gripper
  :parameters (?x - Block ?g - Gripper)
  :precondition
  (and (is_open ?g)
    (unloaded ?g))
  :effect
  (and (loaded ?g ?x)
    (not (is_open ?g))
    (not (unloaded ?g))))
```

The method to decompose the pick task in the case of 2x2 block will have the selected gripper, the block to be picked, the direction of the placement, and the waypoint where the gripper will move to after picking the block as parameters. This method decomposes the pick task into a series of sub-tasks/actions in the case that all the pre-conditions are satisfied:

```
(:method pick_2x2
  :parameters (?g - Gripper ?x - Block_2x2 ?d
    - Direction ?pa - Point_Air)
  :task (pick ?g ?x ?d)
  :precondition
  (and (not (left_direction ?d))
    (not (right_direction ?d)))
  :ordered-subtasks
  (and (Open_gripper ?g)
    (Rotate_empty_gripper_V ?g)
    (Move_open_gripper_V_To_Stock ?g ?pa)))
```

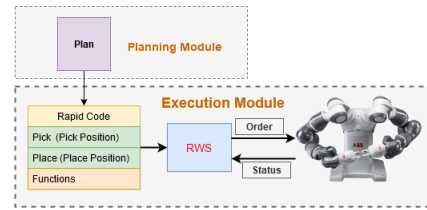


Figure 4: Execution module

```
(Hold_gripper ?x ?g)
(take_V ?x ?g ?pa))
```

EXECUTION MODULE

The execution of the adapted behavior represented in the generated plan is done in our scenario on the cobot YuMi (for the sake of simplicity we used one arm only). Pick/Place locations are passed from the plan to the low-level language (RAPID) to execute the corresponding motion. We have developed a special Application Programming Interface (API) to use the Robot Web Services (RWS) platform provided by the robot manufacturer in order to execute the plan on the cobot, and get the execution status (Fig 4).

USER INTERFACE MODULE

It is important to express the cobot behavior to the operator to ensure workflow smoothness in the collaborative joint task, hence a user interface is used to explain: i) Assembly progress. ii) Cobot's next movement, to ensure human awareness of the cobot behavior in order to avoid conflicts. iii) Operator mistakes, to make the cobot adaptation clear and understandable for the operator and to increase the operator acceptance.

Therefore, we have developed a user interface to demonstrate the cobot behavior, where the elements inside this interface emulate the situation of the workplace. It contains three distinctive areas (Fig 5): 1. Assembly pattern selection and control area. 2. Assembly and swap zone area. 3. Messages area.

The first area is used to control the experiments, while the rest are interactive areas, area (2) visualises the location and the color signals, while area (3) is used to display the interactive messages.

The color signals used in the graphical interface are as follows: I) Green-colored squares to represent the green base-plate to indicate unoccupied positions. II) Red-blinking squares to represent the positions that the cobot will use to place a block. III) Yellow/Blue colored squares to represent the occupied positions. IV) Red cross on top of the Yellow/Blue squares to represent any incorrect placement.

Moreover, the interactive messages are: i) "I'm taking a photo to understand the situation." ii) "I'm placing a block in the blinking area." iii) "I can't help you anymore, I'll let you finish." iv) "The task is solved, thank you very much"

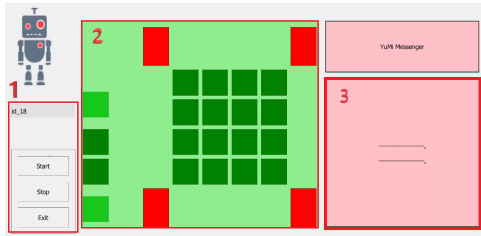


Figure 5: Graphical interface module

Experiments

The objective of this experimentation is to evaluate the impacts of integrating intelligent cobot equipped with an adaptive action-selection mechanism and free-style user interaction on runtime and operators' well-being. The former (runtime) represents productivity, while the latter (well-being) is interpreted as the physical and cognitive demands of the task. Thus, we consider the performance with an extended vision that takes into account not only runtime but also human ergonomics.

The experiments were performed in two phases, the first with 33 participants, and the second with 23 participants different from the first phase. They all were informed about the experiments and signed a consent form beforehand.

Experimental Scenario

The experiments are meant to simulate a collaborative assembly task in an industrial environment, hence LEGO blocks were used to construct four patterns (Fig 6) each of them is 4×4 blocks of two different colors yellow/blue (Fig 6), pattern (D) is used for demonstration purpose, while the rest are used for the experiments with the participants. All the experiments have been designed by ergonomists (co-authors) expert in industrial workplaces to specifically investigate free-style human-robot interaction in realistic situations.

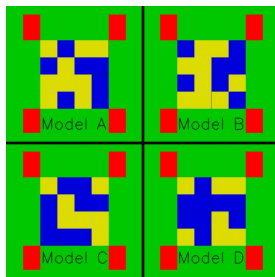


Figure 6: Assembly patterns

The hardware of the experimental setup is as follows: I) Raspberry Pi 3B+ module with wide-angle camera. II) PC. III) Screen to display the user interface. IV) YuMi cobot. V) Lego blocks and base-plate for the assembly.

In this workspace divided into four spaces (Fig 2), the participant is supposed to use the stock from his/her dedicated area, and the swap area, while the cobot is programmed to

use the stock from its dedicated area. Initially, the swap and the assembly areas are empty, while the stock of both the participant and the cobot is filled with blocks of two different sizes 2×2 and 2×4 , and two different colors: yellow, and blue. The user interface is used to start the experiment, where the cobot starts by perceiving the environment, and the user is informed through a message displayed on the user interface "I'm taking a photo to understand the situation.", the perceived environment is interpreted into a discrete representation exploited by our conceptual model to generate the adaptive response. After that the cobot executes the sub-task (single block assembly) and the participant is informed of the cobot action and place of execution by the user interface: the message "I'm placing a block in the blinking area." appears on the user interface, and the red-blinking squares, which reflects the positions that the cobot needs to do the placement, appear on the screen (Fig 5). The user at this stage is supposed to do the assembly at the same time as the cobot, with respect to the red-blinking positions that the cobot will use. This loop of perception, planning, collaborative action is repeated until the cobot cannot place any block because of the size of the robot's fingers. Then the user is asked to complete the task alone in the user interface: "I can't help you anymore, I'll let you finish.", and this message is presented until the perception module verifies the task completion, thus the last message appears: "The task is solved, thank you very much". The assembly task complexity stems from the fact that the task is changing according to the selected pattern, thus the size and the color of the blocks vary from one pattern to another making the task cognitively demanding.

Moreover, this scenario reflects actual collaborative assembly tasks in manufacturing where the operator and the cobot execute teamwork. The code used is released online¹ for experiment reproducibility.

Evaluation Criteria

Four factors were used for evaluation in our case: *time* of the task completion, to evaluate the impact on the productivity, *errors* committed during the task (any misplacement is considered an error, even if it was corrected later), to evaluate the impact on the quality, count of *gestures* per task (each movement of the participant's hand to do a sub-task is counted as a gesture, even if he did not place a block), which in turn reflects the physical demands of the task, and *cognitive load* measured using the NASA-TLX [7] form.

Methodology

The experiments were conducted in two separate phases with different groups of participants in each phase: in the first phase, one person fulfills the block assembly task, this constitutes our baseline. In the second phase, the assembly task is jointly realised by the cobot and a human operator. It is noteworthy that the experiments focus on the Human-Robot adaptation in joint task completion, not manipulation skills. From this perspective, comparable setups can be met

¹<https://github.com/bhomaidan1990/Legos>

in the industry without significant changes because the assembly process is similar, though the size and the shape of the assembled parts vary with the considered application.

First Phase: Human Alone

The first phase of the experiments was performed with humans only (without the cobot), in order to compare the results with the second phase. This phase of the experiments consists of four steps:

1. *Introduction of the tasks to be carried out:* a short introduction of the tasks and the framework is given to familiarize the participant with the experiment. The setup is as in Fig. 2, except that the cobot, the camera, and the user interface are not present in the first phase.
2. *A demonstration of assembly pattern:* a demonstration of "Model D" assembly is carried out to eliminate any ambiguity, then the participant is asked if he has any further questions about the task assembly to be clarified.
3. *The assembly tasks:* The user is asked to assemble the other three models (A, B and C) according to a given document in a specific order² one by one, and the experiments are video recorded to be analyzed later.
4. *The survey:* Finally, the participant is asked to answer the NASA-TLX test to measure his/her cognitive load.

Second Phase: Human and Cobot

The second phase involves the cobot with the human to achieve a shared task. Furthermore, this phase has three additional components: the *cobot*, the *camera* to capture the world, and the *user interface* projected on the external screen beside the cobot. The second phase of the experiment is executed in four steps as follows:

1. *Introduction of the tasks to be carried out:* a short introduction of the cobot, the tasks, and the framework is given to familiarize the participant with the experiment. The setup that the participant is introduced to consists of:
 - The user interface (Fig 5) including the signals and the messages that the participant will encounter during the experiment.
 - The camera, and the captured workspace.

And the workplace:

 - (a) The *cobot* abilities and the high safety constraints to make the participant feels confident that there is no harm in working in collaboration with the cobot.
 - (b) The *pattern* which has to be assembled in collaborative manner with the cobot.
 - (c) The *stock to be used by the participant*.
 - (d) The *stock of the cobot*, which the participant should not use.
 - (e) The *assembly zone* where the tasks are carried out.
 - (f) The *swap zone*, where the cobot handles a block to the participant to assist him in doing the task in case the user is out of stock of a specific size/color.

²The sequence introducing the patterns is randomized to avoid any bias in the results.

2. *A demonstration of assembly pattern:* a demonstration of "Model D" assembly is carried out with assistance of the cobot and the user interface to eliminate any ambiguity or panic, then the participant is asked if he/she has any further questions about the task assembly, the cobot behavior, or the user interface to be clarified.
3. *The assembly tasks:* The user is asked to assemble the other three models (A, B and C) in a specific sequence² one by one with assistance of the cobot, and the experiments are video recorded to be analyzed later. Note that at this phase, the start/finish signals of the tasks were replaced by the messages emitted from the user interface, and not expressed by the participant as in the first phase.
4. *The survey:* Finally, the participant is asked to answer the NASA-TLX test to measure his/her cognitive load.

EXPERIMENTAL RESULTS

The preliminary results (Table 1) indicate a significant decrease in the average number of errors (60%), the average number of gestures (31%), and the average subjective cognitive load (2%). On the contrary, the time for task completion has increased (150%) due to the simplification³ in the framework, hence finer optimization has to be considered in the further studies to remediate this drawback.

	Phase 1			
	Min	Max	SD	Avg
Time (minutes)	1.1	4.4	0.92	2.73
Errors	0	16	4	5
Gestures	41	124	20	74
Cognitive load	15.2%	59.07%	12.91%	37.49%
	Phase 2			
	Min	Max	SD	Avg
Time (minutes)	4	10.68	1.55	6.82
Errors	0	6	1.83	2
Gestures	34	72	10	51
Cognitive load	11.33%	62.53%	15.32%	35.52%

Table 1: Experimental Results. SD - Standard Deviation, Avg - Average value. Phase 1: 33 participants. Phase 2: 23 participants different from Phase 1.

Discussion

In this work, we proposed a modular architecture to adapt cobot behavior to human task variability. This architecture makes use of HTN to generate flexible plans that account for human task variability, allows free-style human-robot interaction, and assists the human partner in correcting their errors without hindering the progress of the assembly process. These preliminary results show that, from an ergonomic perspective, this architecture and the automated planning are adequate to address human-cobot interactions effectively in

³Only left arm of the cobot was used, and the speed was reduced for the sake of comfort human-robot interaction.

industrial workplaces. The intelligent behavior acceptability is enhanced by a user interface that makes explicit the robot behavior to the user. The errors decrease significantly indicating higher quality in performing the task, the number of gestures also decreases, which reflects less physical load, and the cognitive load has significantly decreased.

Task completion time increase highlights the importance of utilizing an enhanced module for the motion and the coordination between the robot and the human partner. We do believe there is room to enhance this architecture's runtime performance and converge towards human-alone effectiveness.

References

- [1] Caroly, S.; Landry, A.; Cholez, C.; Davezies, P.; Bellemare, M.; and Poussin, N. 2012. Innovation in the occupational health physician profession requires the development of a work collective to improve the efficiency of msd prevention. *8th World congress on Ergonomics - Designing a sustainable future* 41:5–13.
- [2] Cheng, Y.; Sun, L.; and Tomizuka, M. 2021. Human-aware robot task planning based on a hierarchical task model. *IEEE RA-L* 6(2):1136–1143.
- [3] Erol, K.; Hendler, J.; and Nau, D. S. 1994. Htn planning: Complexity and expressivity. In *AAAI*, volume 94, 1123–1128.
- [4] Faroni, M.; Beschi, M.; Ghidini, S.; Pedrocchi, N.; Umbrico, A.; Orlandini, A.; and Cesta, A. 2020. A layered control approach to human-aware task and motion planning for human-robot collaboration. In *2020 29th IEEE RO-MAN*, 1204–1210.
- [5] Gabler, V.; Stahl, T.; Huber, G.; Oguz, O.; and Wollherr, D. 2017. A game-theoretic approach for adaptive action selection in close proximity human-robot-collaboration. In *2017 IEEE ICRA*, 2897–2903.
- [6] Grigore, E. C.; Roncone, A.; Mangin, O.; and Scasselati, B. 2018. Preference-based assistance prediction for human-robot collaboration tasks. In *2018 IEEE/RSJ IROS*, 4441–4448.
- [7] Hart, S. G., and Staveland, L. E. 1988. Development of nasa-tlx (task load index): Results of empirical and theoretical research. In Hancock, P. A., and Meshkati, N., eds., *Human Mental Workload*, volume 52 of *Advances in Psychology*. North-Holland. 139–183.
- [8] Hawkins, K. P.; Bansal, S.; Vo, N. N.; and Bobick, A. F. 2014. Anticipating human actions for collaboration in the presence of task and sensor uncertainty. In *2014 IEEE ICRA*, 2215–2222.
- [9] Höller, D.; Behnke, G.; Bercher, P.; Biundo, S.; Fiorino, H.; Pellier, D.; and Alford, R. 2020. HDDL: an extension to PDDL for expressing hierarchical planning problems. In *AAAI*, 9883–9891. AAAI Press.
- [10] Johannsmeier, L., and Haddadin, S. 2017. A hierarchical human-robot interaction-planning framework for task allocation in collaborative industrial assembly processes. *IEEE RA-L* 2(1):41–48.
- [11] Kalakoski, V.; Selinheimo, S.; Valtonen, T.; Turunen, J.; Käpykangas, S.; Ylisassi, H.; Toivio, P.; Järnefelt, H.; Hannonen, H.; and Paajanen, T. 2020. Effects of a cognitive ergonomics workplace intervention (cogerg) on cognitive strain and well-being: a cluster-randomized controlled trial. a study protocol. *BMC Psychology* 8(1).
- [12] Karasek, R. A. 1979. Job demands, job decision latitude, and mental strain: Implications for job redesign. *Administrative Science Quarterly* 24(2):285–308.
- [13] Kuhner, D.; Aldinger, J.; Burget, F.; Göbelbecker, M.; Burgard, W.; and Nebel, B. 2018. Closed-loop robot task planning based on referring expressions. In *2018 IEEE/RSJ IROS*, 876–881.
- [14] Le, A. T.; Kratzer, P.; Hagenmayer, S.; Toussaint, M.; and Mainprice, J. 2021. Hierarchical human-motion prediction and logic-geometric programming for minimal interference human-robot tasks. In *2021 30th IEEE RO-MAN*, 7–14.
- [15] Lee, S.-J.; Park, J.-M.; Kim, D.-H.; and Kim, J.-H. 2018. Adaptive task planner for performing home service tasks in cooperation with a human. In *2018 IEEE/RSJ IROS*, 2857–2864.
- [16] Milliez, G.; Lallement, R.; Fiore, M.; and Alami, R. 2016. Using human knowledge awareness to adapt collaborative plan generation, explanation and monitoring. In *2016 11th ACM/IEEE HRI*, 43–50.
- [17] Pan, T.; Wells, A. M.; Shome, R.; and Kavraki, L. E. 2021. A general task and motion planning framework for multiple manipulators. In *2021 IEEE/RSJ IROS*, 3168–3174.
- [18] Paxton, C.; Jonathan, F.; Hundt, A.; Mutlu, B.; and Hager, G. D. 2018. Evaluating methods for end-user creation of robot task plans. In *2018 IEEE/RSJ IROS*, 6086–6092.
- [19] Pellier, D., and Fiorino, H. 2018. PDDL4J: a planning domain description library for java. *J. Exp. Theor. Artif. Intell.* 30(1):143–176.
- [20] Pellier, D., and Fiorino, H. 2020. Totally and partially ordered hierarchical planners in PDDL4J library. *Proceedings of the International Planning Competition* abs/2011.13297.
- [21] Roncone, A.; Mangin, O.; and Scasselati, B. 2017. Transparent role assignment and task allocation in human robot collaboration. In *2017 IEEE ICRA*, 1014–1021.
- [22] Rovida, F.; Grossmann, B.; and Krüger, V. 2017. Extended behavior trees for quick definition of flexible robotic tasks. In *2017 IEEE/RSJ IROS*, 6793–6800.
- [23] Srinivasan, D., and Mathiassen, S. E. 2012. Motor variability in occupational health and performance. *Clinical Biomechanics* 27(10):979–993.
- [24] Zhu, Y.; Tremblay, J.; Birchfield, S.; and Zhu, Y. 2021. Hierarchical planning for long-horizon manipulation with geometric and symbolic scene graphs. In *2021 IEEE ICRA*, 6541–6548.