2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 25-29, 2020, Las Vegas, NV, USA (Virtual)

Supportive Actions for Manipulation in Human-Robot Coworker Teams

Shray Bansal[1], Rhys Newbury[2], Wesley Chan[2], Akansel Cosgun[2], Aimee Allen[2],
Dana Kulić[2], Tom Drummond[2] and Charles Isbell[1]

Abstract— **The increasing presence of robots alongside humans, such as in human-robot teams in manufacturing, gives rise to research questions about the kind of behaviors people prefer in their robot counterparts. We term actions that support interaction by reducing future interference with others as *supportive* robot actions and investigate their utility in a co-located manipulation scenario. We compare two robot modes in a shared table pick-and-place task: (1) Task-oriented: the robot only takes actions to further its task objective and (2) *Supportive*: the robot sometimes prefers *supportive* actions to task-oriented ones when they reduce future goal-conflicts. Our experiments in simulation, using a simplified human model, reveal that supportive actions reduce the interference between agents, especially in more difficult tasks, but also cause the robot to take longer to complete the task. We implemented these modes on a physical robot in a user study where a human and a robot perform object placement on a shared table. Our results show that a *supportive* robot was perceived more favorably as a coworker and also reduced interference with the human in one of two scenarios. However, it also took longer to complete the task highlighting an interesting trade-off between task-efficiency and human-preference that needs to be considered before designing robot behavior for close-proximity manipulation scenarios.**

I. Introduction

Despite the continued growth of industrial robot sales [1], many assembly tasks are still performed manually in major industries [2]. A vision for the future of manufacturing involves robots working alongside human coworkers on tasks that exploit the respective strengths of both. Surveys identify interaction with co-workers as one of the most important job criteria for human workers [3]. We introduce interaction-supporting actions that aim to improve the coworker experience in human-robot co-located manipulation. We implement these in a close-proximity manipulation task to understand the impact on task performance and the coworker perception as compared to a robot focused solely on completing its task.

We term actions necessary for an agent to complete their task in the absence of other agents as *task-oriented*. We define *supportive* actions as actions that support the interaction by reducing potential interference with other agents but are not necessary for task completion. *E.g.*, when resetting a chessboard, the black agent performs *task-oriented* actions by moving the black pieces to their positions and *supportive* actions by moving the white pieces towards the white player. Although the *supportive* actions help the other agent, they are not altruistic as the agent hopes to benefit from the reduced interference. Humans also perform *supportive* actions, perhaps due to their modeling of others as intentional agents that plan for mutual benefit [4], [5], or their expectations of reciprocity [6].

[1] School of Interactive Computing, Georgia Institute of Technology, Atlanta, GA, 30308, USA. sbansal34@gatech.edu
[2] Department of Electrical and Computer Systems Engineering, Monash University, Clayton, VIC, Australia.

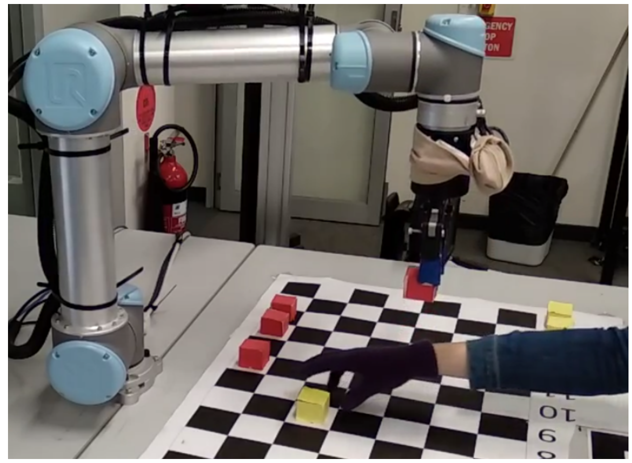


Fig. 1. An example scene from our co-located manipulation scenario. The robot's goal is to place all the red blocks into the row closest to itself, and the human participant's goal is to do the same for the yellow blocks.

Our task is inspired by other close-proximity human-robot interaction (HRI) manipulation studies [7], [8]. It involves two agents, a human and a robot, situated across a table scattered with color-coded blocks, each aiming to bring the blocks of their assigned color quickly back to their side (Fig. 1). The agents have been assigned separate goals without a direct incentive for cooperation and the shared table is expected to induce interference. We focus on high-level decision-making and design *supportive* actions that proactively avoid collision by modifying the goal configuration of the other agent by moving their blocks. In our experiments, the robot operates in one of two modes: (1) *Task-oriented*, where the robot takes only *task-oriented* actions, and (2) *Supportive*, where the robot takes both *supportive* and *task-oriented* actions depending on the situation. We hypothesize that the *supportive* mode would reduce interference and lead to a better human experience. We test this in simulation with a simplified human model and verify it with a user study on a physical robot.

Our main contributions are the introduction of supportive actions in a human-robot collaborative manipulation task, simulation experiments and user study experiments that justify the use of such actions, and the identification of a trade-off between operational and usability metrics when the robot is designed to deliberately take *supportive* actions.

After reviewing the literature in Sec. II, we formulate the problem in Sec. III and describe our methodology in

Sec. IV. We first experiment in simulation in Sec. V to design the *supportive* actions and formulate hypotheses in Sec. VI. We present the implementation and user study details in Sections VIII and VII, respectively. We analyze results in Sec. IX, discuss them in Sec. X and conclude in Sec. XI.

II. Related Work

Human-robot interaction (HRI) includes scenarios where agents work for a shared goal as well as scenarios where agents have separate, sometimes competing, objectives. We are interested in the latter, where humans and robots work alongside each other on separate goals and conflict arises due to shared space.

A focus on anticipating human actions has enabled better robot assistants. For instance, Hawkins *et al.* [9] do this by exploiting known task structure and Nikolaidis *et al.* [10] use online adaptation to user preferences. Similar to us, both of these approaches focus on the high-level decision-making aspect of the task. Cherubini *et al.* [11] plan low-level robot actions and successfully reduce human workload for automotive manufacturing. However, they assume that the role of the robot should be to assist, which simplifies the robot's decision-making. However, the types of roles and interaction modes in mixed human-robot teams are richer, as shown by Gombolay *et al.* [12].

Other work in human-robot co-presence treats the human as an obstacle to be avoided [8], [13]. In this case, the human's goal is either not considered at all, or only used to make predictions to guide a more proactive obstacle avoidance. Our task involves separate goals for the two in a shared space which induces the relatively under-explored concept of Human-Robot goal conflict.

Similar close-proximity manipulation tasks have also been studied by [14], [7]. Bansal *et al.* [14] consider different goals for the human and the robot and develop a game-theoretic model to plan low-level trajectories to minimize conflict between the two goal-driven agents. While Gabler *et al.* [7] consider a collaborative scenario, they design the robot's utility to include both shared and separate goals from the human. They use game-theory to make plans that optimize the order of the robot's task-oriented actions for increased joint task-efficiency. We introduce *Supportive* actions that modify the human's goal configuration by moving their blocks to reduce future conflict.

III. Problem Formulation

We design a pair of pick-and-place tasks on a table shared by a human and a robot and represent it as a two-agent game. The table has two sets of blocks distinguished by color, we assign one set to the robot $\boldsymbol{b_R} = \{b_R^1, ..., b_R^n\}$ and the other to the human $\boldsymbol{b_H} = \{b_H^1, ..., b_H^n\}$. We draw a $2D$ grid on the table and place each block in a single cell. We define this cell as the block's location $l(b^i) = (r, c)$ and a cell near its assigned agent as its destination, $d(b^i) = (r, c)$. A state is a configuration of blocks on the grid, $s = \{b_R^i, b_H^i\} \quad \forall i \quad 1 \leq i \geq n$. Fig.2 shows a grid configuration where $n = 2$ and $\boldsymbol{b_R} = \{\mathbf{1}, \mathbf{3}\}$ and $\boldsymbol{b_H} = \{\mathbf{2}, \mathbf{4}\}$.

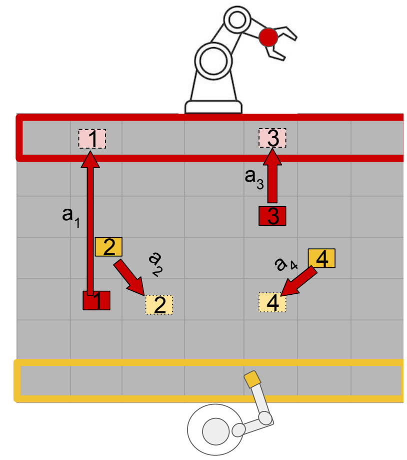


Fig. 2. An example board configuration consisting of blocks $(1 - 4)$ for the robot (red) and the human (yellow). Robot actions (a_{1-4}) are depicted by the arrows. a_1, a_3 are *task-oriented* actions while a_2, a_4 are *supportive* and a_2 is a more useful *supportive* action because it reduces potential interference when reaching for block 1.

In this task, an action a can move at most one block to a different location. For instance, in Fig. 2, $a_1 = (\mathbf{1}, d(\mathbf{1}))$, moves block $\mathbf{1}$ from its location to the goal. We also allow idle actions that do not move any blocks. Both agents are instructed to start performing actions simultaneously, and so, if one agent finishes their action early, they have to wait for the other to complete their action before starting to perform the next one. We assign each agent the goal to take actions that lead to a state, s^G, where each of their blocks is in its destination cell, in minimum time. Their goal state only includes their own blocks and not the others.

IV. Method

We first explain how to construct the sets of *task-oriented* and *supportive* actions and then describe two decision-making strategies used by the robot to perform the task.

A. Action Sets

We define two action sets for the robot to use: task-oriented, $A^{TO}(s)$ and *supportive*, $A^S(s)$. A^{TO} includes actions that each move a robot block to its destination,

$$A^{TO}(s) = \{a = (b_R, d(b_R)) \mid \forall b_R, l(b_R) \neq d(b_R)\}. \quad (1)$$

A^S includes the *supportive* actions. We define a *supportive* action, $a = (b_H, d)$, for one of the human's blocks, b_H, that is closer to the robot than the human. We set the destination,d, of this action as the closest empty cell that is closer to the human. This way, we balance the cost of the additional action with reducing the potential for interference while favoring the human's preference of retrieving objects near them. *E.g.*, in Fig. 2, $A^{TO} = \{a_1, a_3\}$ and $A^S = \{a_2, a_4\}$.

B. Task-Oriented Robot

The *task-oriented* baseline randomly samples an action from the *task-oriented* set at a given state, $a_R \sim A_R^{TO}(s)$. The goal is to complete the task with the fewest actions.

It chooses randomly because all *task-oriented* actions are necessary for reaching the goal state.

C. Supportive Robot

The *Supportive* robot chooses actions using a policy, π containing actions from *task-oriented* and *supportive* sets. This policy is an ordered set of actions ranked by their priority and is defined by the user before starting the task. We take a heuristic approach to create π for the task to reflect the utility of *supportive* actions.

We initialize π as an empty list and populate it by iterating over the following rules until no new action is generated. We also initialize $\mathtt{B}$ to a list of all the blocks in the grid.

1) Return empty if $\mathtt{B}$ is empty.
2) If a block $b_R^i \in \mathtt{B}$ exists such that b_R^i has no human block that might cause a conflict when reaching for it, then pop b_R^i and return a task-oriented action for it.
3) Else, find a *supportive* action from $\mathtt{B}$ that has conflict with the most robot objects in $\mathtt{B}$.

This approach is designed to produce actions that reduce the probability of collision between the human and the robot while trying the minimize the task completion time. It applies to any block configuration.

Given a predefined policy, π, the robot checks the list in order and executes the first action that is feasible in the current state s. If no feasible action is found, it defaults to sampling available *task-oriented* actions until the goal is reached. We assign a fixed π in our task to ensure that the participants observe similar behavior from the robot in every trial.

Fig.2 depicts an example task with four blocks, *task-oriented* actions, $A^{TO} = \{a_0, a_2\}$, and *supportive* actions, $A^S = \{a_1, a_3\}$. The policy, π, for this scenario is $\pi = (a_2, a_1, a_0)$. Here, a *task-oriented* action, a_2, is included first because of the lack of potential goal conflict of block **3**; then the robot takes a *supportive* action, a_1, to reduce the potential interference of block **2**; and finally, it completes the task with the last *task-oriented* action a_0. The planner ignores *supportive* action, a_3 because block **4** causes no potential interference with the robot's blocks.

V. Simulated Experiment

We simulate a scenario with two 2-link robot arms performing pick-and-place actions in 2D (Fig. 3). Our goal is to observe the effect of *supportive* actions in an idealized setting, without the variance introduced by the participants, or errors in sensing and actuation.

We develop an OpenRAVE [15] environment with blocks of two colors scattered on a table. We assign each arm six blocks of the same color. The goal for each arm was to bring blocks of their assigned color to the destination area near the arm, highlighted in Fig. 3. We define a 7×15 grid on the table and place the blocks into these cells according to two configurations, *easy* and *hard*, as shown in Fig. 4. We consider one arm as the robot and the other one as a simulated human. The simulated human chooses *task-oriented* actions while prioritizing closer blocks. We experiment with the robot following both task-oriented and *supportive* algorithms from Sec. IV. The RRT* [16] implementation in OMPL [17] is used to plan joint-space trajectories.

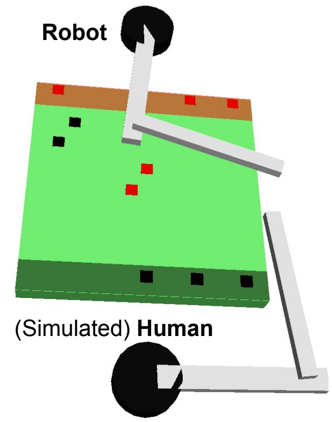


Fig. 3. The simulated 2D environment with two arms, one is a simulation of the human and the other is controlled by the robot policy.

TABLE I
Simulation Results

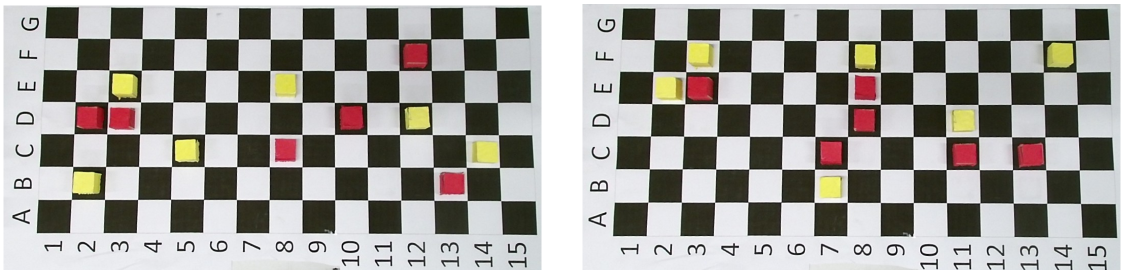
Scenario	Robot Mode	Task Time (s)	Safety Stops
Easy	*Task-Oriented*	15.46 ± 0.3	4.6 ± 0.7
	Supportive	17.75 ± 0.2	3.0 ± 0.7
Hard	*Task-Oriented*	15.9 ± 0.9	7.3 ± 2.2
	Supportive	18.56 ± 0.5	2.4 ± 1.2

Results. The two scenarios and two robot modes make four experimental conditions. Since we use a sampling technique to generate plans, we run each condition 10 times and present the mean and standard deviations in Tab. I. The time taken to complete the task by the slowest agent is termed Task Time. We also record the times the simulated robot was stopped during the interaction to prevent a collision and term these safety stops. The robot stops and waits for the simulated human to move a threshold distance away when this happens while the human is free to move. We find task completion time to be higher for the *supportive* robot but the safety stops are lower in Tab. I. A larger effect due to *supportive* actions is observed for both metrics in the *hard* scenario. The *supportive* robot is always slower than the human and although the additional actions cause a longer task time they also reduced goal conflict leading to less than 50% safety stops in the *hard* scenario.

VI. Hypotheses

Following simulation results, we anticipate the robot's behavior and the initial block configuration to affect collaborative performance. We formulate the following hypotheses to test on a user study with a physical robot.

H1 *Supportive actions will reduce the interference between the agents.* In particular, we expect the *supportive* actions to reduce the safety stops occurring in the interaction, especially for difficult scenarios.



(a) Easy (b) Hard

Fig. 4. Layout of the *easy* (left) and *hard* (right) block configurations, viewed so the human is seated below row A. The human places yellow blocks on the numbers below row A, whereas the robot is across the table and placing red blocks in Row G. The difficulty is due to the conflict caused by the robot and human reaching for the same space. This conflict exists more in (b) since most of the yellow blocks are in front of the robot's.

H2 *Supportive actions will reduce the human's time to complete the task.* We expect people to complete the task faster when interacting with the *supportive* robot leading to more idle time, especially for difficult scenarios.

H3 *Supportive actions will have a positive effect on the subjective measures of task performance.* We expect that participants will prefer the *supportive* robot as a coworker, especially for difficult scenarios.

H4 *Changing the initial block configuration would affect both the subjective and objective measures.* In particular, we expect that participants will find the task more difficult to perform if the initial block configuration includes more goal conflicts. We also expect the effect of *supportive* actions to be more prevalent in difficult scenarios, in general.

VII. User Study Design

We conduct a user study to test the effect of the *supportive* actions. The study was approved by Monash University's Ethics Review Board, Project ID 21010.

A. Independent Variables

We manipulate two independent variables.

- **Robot mode**: $\{$*Task-Oriented*, *Supportive*$\}$ robots as described in Section IV.
- **Scenario**: $\{$*Easy*, *Hard*$\}$ block configurations. (Fig. 4)

The block configuration in the *easy* and *hard* scenarios are designed to cause different levels of goal conflict. While both of them include six blocks, the robot's blocks in the *hard* scenario were arranged to be directly in front of the human's. We expect this would increase task difficulty by causing more interference since both agents need to reach into the same space.

B. Participant Allocation

We recruited 18 subjects aged $20 - 31$ ($M = 22.4$, $SD = 3.1$, 11 male, 7 female) for a within-subject study. To reduce order effects, we counterbalanced the order of the robot mode. We kept the scenario order the same, where *hard* always followed *easy*. The participants were not informed about the kind of robot they would be interacting with or how many types there were.

C. Procedure

The experiment took place in a university lab under experimenter supervision. We seated participants in front of the robot as depicted in Fig.5. After reading the explanatory statement and signing a consent form, they listened to a scripted explanation of the experiment.

The participants were assigned yellow blocks and their goal was to move these blocks to their destinations accurately while minimizing task time. The start of a turn was signaled on the scanning display in Fig. 5 and both agents performed reaching actions simultaneously, continuing until all their blocks were in their respective destinations. This concluded one trial and each participant performed four. Participants were also given three types of surveys, a demographic one at the start of the experiment, one after every trial, and one at the end to record their overall experience. A complete experiment took between 30 and 45 minutes.

D. Dependent Variables

We record both objective and subjective metrics.

Objective measures. We study the effect of *supportive* actions on task completion time for each agent, the total number of safety stops, as well as human's idle time ratio. The task completion time is the time an agent takes to complete a trial and is easily measured for the robot since we programmatically record the time when the robot starts and finishes an action. For the human, we manually annotate this using a video recording of the experiments. We also annotate the time the human waits for the robot after completing an action and compute the ratio of the accumulated wait time over a trial to their total execution time as the human-idle ratio. We also count the times the robot has to stop due to proximity to the human as safety stops.

Subjective measures. Participants answered ten 5-point questions after each trial. Five of these are collected in a Likert-scale that measures robot proficiency as a coworker and include statements about the robot's helpfulness, action-selection, intention-prediction, disruption, *etc*. The rest of the questions are treated as individual differential scale items. We reversed some of the scale items so that 5 is the most positive response to each and 1 the least. We adapt this survey from collaborative HRI studies like [18]. The Likert-scale

TABLE II

LIKERT-SCALE COMPOSED OF INDIVIDUAL SURVEY ITEMS WITH CRONBACH'S α. (R) INDICATES A REVERSE SCALE.

Robot coworker proficiency ($\alpha = 0.807$)
I believe the robot accurately perceived my goals.
The robot was helpful and/or cooperative.
The robot seemed to select the correct object to pick up most of the time.
The robot disrupted me in efficiently performing the task. **(R)**
I felt uncomfortable with the robot. **(R)**

TABLE III

INDIVIDUAL SCALE ITEMS FROM SURVEY.

Individual Measures
I1 How successful were you in achieving your task?
I2 How hard did you have to work to accomplish your level of performance? **(R)**
I3 How much attention did you pay to the robot and its performance during the task?
I4 I felt unsafe with the robot.**(R)**
I5 How would you grade the robot as a coworker, overall?

(Cronbach's $\alpha = 0.807$) is listed in Tab. II and the individual items are listed in Tab. III.

VIII. IMPLEMENTATION DETAILS

Our user study setup is depicted in Fig. 5 and includes the robot and the human around a table with a checkerboard grid on which we place the blocks. We mount an RGB-D sensor overhead to detect the blocks and the person's arm. These detections guide the robot's action-selection and trajectory planning, which are implemented on the Universal Robot 5 (UR5) using the Robot Operating System (ROS) [19]. We also include a scanning area that instructs the participant about the destinations for their blocks. Our experiment is fully-autonomous and does not require human intervention.

A. Sensing

The location of the grid is calibrated in the camera frame ahead of time using OpenCV [20] and we apply a simple color blob detection technique to the RGB image in real-time to localize the blocks.

We instructed participants to wear a colored glove covering their arm to allow for its easy detection. We ensure safety by stopping the robot arm if the user's hand comes within a fixed distance threshold.

B. Robot Control

We implement both *task-oriented* and *supportive* robot policies for action-selection. For a given goal grid location, we generate waypoints for the robot end-effector to it at a fixed vertical offset from the grid and use the MoveIt framework [21] to generate a Cartesian path. This path is followed by the robot controller after which it attempts a vertical move down to either grab or drop the block and then moves back up. Robot joint speed is limited to $(0.314 rad/s)$ to ensure user safety and comfort.

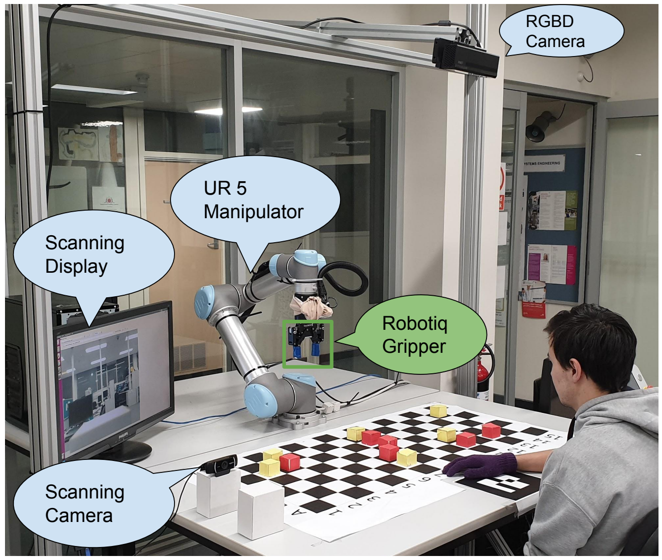


Fig. 5. The experimental setup

We also included a camera station where participants scanned blocks and were informed of their destinations after a short delay. We use this delay to account for the human's higher relative speed to synchronize human-robot actions.

IX. USER STUDY RESULTS

We compare the independent variables through the objective task performance metrics first and then by participant responses to the survey. We had to remove the data for two participants, one due to a robot failure, and the other because the participant did not follow experimental directions. Thus, in total we analyze $(N = 16) \times 4 = 64$ trials.

A. Objective Measures

We analyze some of the objective metrics in Fig. 6.

Safety Stops. We count the times when the robot has to stop due to its proximity to the human's arm. We compare robot types through a Wilcoxon signed-rank test on each scenario because the data was not normally distributed. We find a significant effect due to the *supportive* robot in the *hard* scenario ($w = 79.5, p < 0.05$). Fig. 6a shows that the *supportive* robot had fewer stops in *hard* affirming **H2**.

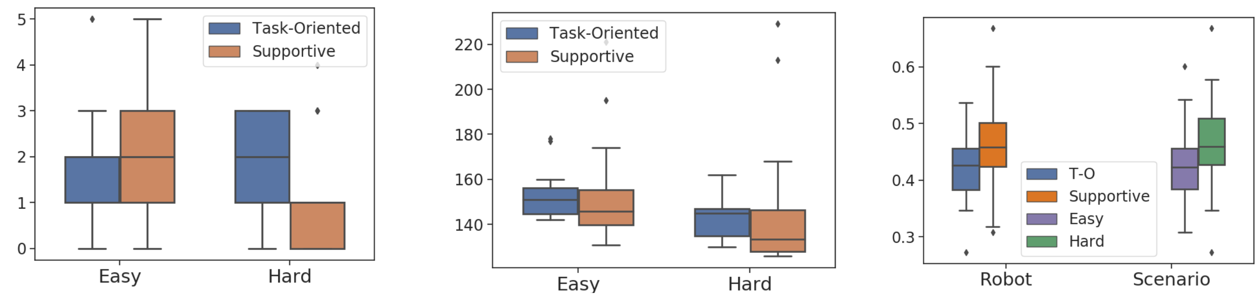
Robot Task Time. We use a repeated-measure two-way ANOVA to compare the robot's task completion time. We find a significant effect due to the *supportive* robot ($F(3, 60) = 74.0, p < 0.01$) and no interaction. Table IV shows that the addition of *supportive* actions led to a longer robot task time.

TABLE IV

TASK COMPLETION TIME OF THE ROBOT.

Robot	Robot Task Time (s)
Baseline	162.9 ± 3.9
Supportive	208.9 ± 3.7

Human Task Time. We use a Wilcoxon signed-rank test to compare the human's task completion time due to the non-normality of this data. We find no significant effect due



(a) Safety Stops

(b) Human Task Time

(c) Human Idle Time

Fig. 6. Objective Measures. Box-and-whisker plots of the (a) number of safety stops; (b) time taken by the human to complete the task; and (c) the proportion of idle time spent by the human. Note, T-O refers to the *task-oriented* robot.

to *supportive* actions for either scenario. Fig. 6b shows the human interacting with the *supportive* robot is faster but with high variance, partly denying **H3**.

Human-Idle Time. We use a repeated-measures two-way ANOVA to analyze the human's idle time ratio as a measure of task fluency. We compute this ratio by accumulating the time the human waited for the robot to complete an action before they could start the next one and dividing it by the human's task time. We find significant effects due to both robot ($F(3, 60) = 7.3, p < 0.05$) and scenario ($F(3, 60) = 5.95, p < 0.05$) types. Fig. 6c shows that *supportive* robot and *hard* scenario each led to higher idle time partially affirming **H3**. This measure was adapted from [18] where it was found to be correlated with higher human preference.

***Supportive* actions.** The robot took on average fewer *supportive* actions in the *easy* (1.9) scenario than the the *hard* (2.6) due to fewer goal conflicts. The participants took only 5 *supportive* actions overall and all of them took place in the *supportive* robot condition.

Summary. The *supportive* robot confirms **H1** in the *hard* scenario by reducing interference; it partly confirms **H3** since human's idle time is increased, however, the human's task completion time is not significantly reduced. Also, the *supportive* robot takes longer to complete this task.

B. Subjective Measures

We analyze some of the survey responses in Fig. 7. We used recommendations from [22] for the following analysis.

Robot coworker proficiency. We perform a two-way repeated-measure ANOVA on the Likert-scale from Table II and find significant interaction ($F(3, 60) = 13.9$, $p < 0.01$). The normalized responses in Fig. 7a show that participants prefer the *supportive* robot in the *hard* scenario but have no preference in the *easy* one affirming **H1** for it. They also show that people prefer *supportive* robot more when the task difficulty increases but preference for the *task-oriented* robot remains similar regardless of task difficulty.

Scenario Effect. We use a Wilcoxon signed-rank test to compare individual scale responses from Table III. We find significant scenario effect for both **I2** ($w = 0.0$, $p < 0.01$) and **I3** ($w = 34.0$, $p < 0.05$). Fig. 7b indicates that participants find the *hard* scenario more difficult to perform, since **I2** is reversed, affirming **H4**. It also leads to the observation that people are more observant of the robot's actions in the *hard* scenario. A caveat is that the perceived difficulty might also be an effect of the order of the two scenarios, since they were kept constant in our study .

Safety Perception. We used a Wilcoxon signed-rank test to compare the **I4** scale item and do not find any significant effect due to *supportive* actions. Fig. 7c shows that participants felt very safe for both robot types in our experiment.

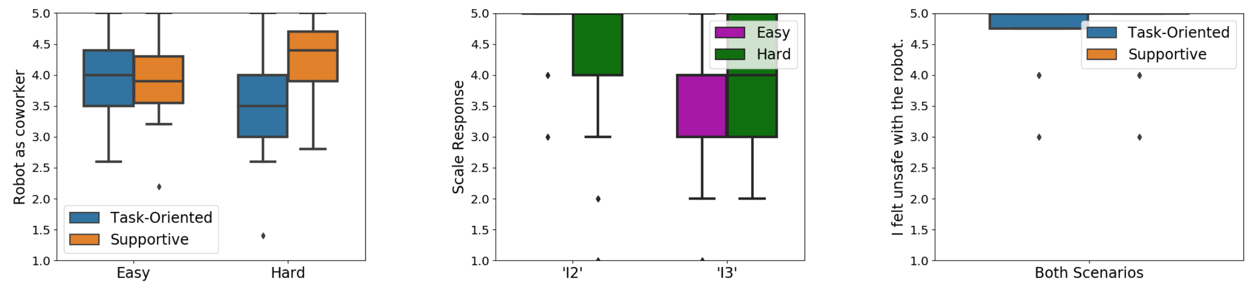
Summary. We find that participants prefer the *supportive* robot as their coworker in the *hard* scenario affirming **H3**; also, participants find the *hard* scenario more difficult and pay more attention to the robot in it, supporting **H4**.

X. Discussion

A surprising finding of our analysis was that *supportive* actions do not reduce safety stops in the *easy* scenario. Safety stops can be caused by factors like an unavoidable conflict between agent goals, uncertainty about each other's goal, and sensor error. We label a configuration as *hard* due to the presence of more goal conflicts; this label does not allude to the other factors. *supportive* actions in our work were designed to reduce goal conflicts, they lead to fewer stops on the *hard* task, but will need to be adapted for other sources of conflict to be effective in different scenarios.

One might think that moving the human blocks close to them would cause people to perceive the robot as helpful and inflate *supportive* robot's proficiency. However, our results, which show that the *supportive* robot is only preferred in the *hard* scenario, provide evidence for the human's preference relying on the situation-dependent suitability of the robot's action-selection.

Hoffman [18] found that collaborative fluency does not track task-efficiency in team tasks. Ours is not a team task, however, our results also show coworker acceptance to be separate from either agent's task-efficiency. We find *supportive* actions to increase coworker acceptance but reduce robot efficiency. They present a trade-off that needs to be considered for designing robot behaviors. *E.g.*, if a robot is introduced into a manual process to reduce repetitive tasks for humans and increase their job satisfaction, then its acceptance might play a bigger role than its efficiency. Our methodology helps highlight this trade-off by combining the subjective and objective impact of supportive robot



(a) Likert Scale

(b) Individual Measures

(c) Safety Perception

Fig. 7. Subjective Measures. Box-and-whisker plots of the (a) normalized survey response to Likert-scale items for the different robot type separated by scenario; (b) response to measures of subjective task difficulty and attention to the robot for the two scenarios; and (c) safety perception for the robot types. Note that the leftmost box in (b) and rightmost box in (c) have no height and so appear as a line at 5.0.

behaviors and applies to other shared-workspace human-robot environments. We consider this methodology as one of the contributions of our work.

XI. Conclusion and Future Work

We introduce interaction-supporting actions and design robot behavior that selects between these and task-oriented actions by considering the human's and its own goals. We implement it on an autonomous robot and evaluate it in a shared-workspace user study. The results show that this robot increases human coworker preference in a scenario with more goal conflicts but decreases efficiency as compared to a robot that only takes task-oriented actions.

Our study illustrates taking actions to support interaction while trading off on efficiency in an assembly task. Although, the rationale from Sec. IV can help guide adaptation to new domains, however, the actions apply only to similar scenarios. In future work, we plan to develop a framework for *supportive* behavior that can perform this reasoning based on task-specific cost functions.

Participants took very few *supportive* actions towards the robot. We believe their unfamiliarity with the task caused uncertainty about allowed actions. An interesting extension would be to apply this to an actual manufacturing task with subjects who are familiar with it to test the generalizability of our findings. We can also improve task naturalness by increasing robot speed by employing better sensors and models for human motion prediction.

References

[1] International Federation of Robotics (IFR), "Ifr press release," 2019, https://ifr.org/ifr-press-releases/news/robot-investment-reaches-record-16.5-billion-usd, Accessed on 2019-10-01.

[2] V. V. Unhelkar, H. C. Siu, and J. A. Shah, "Comparative performance of human and mobile robotic assistants in collaborative fetch-and-deliver tasks," in *2014 9th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. IEEE, 2014, pp. 82–89.

[3] K. S. Welfare, M. R. Hallowell, J. A. Shah, and L. D. Riek, "Consider the human work experience when integrating robotics in the workplace," in *2019 14th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. IEEE, 2019, pp. 75–84.

[4] R. J. Stout, J. A. Cannon-Bowers, E. Salas, and D. M. Milanovich, "Planning, shared mental models, and coordinated performance: An empirical link is established," *Human Factors*, vol. 41, no. 1, pp. 61–71, 1999.

[5] G. Hoffman, "Ensemble: fluency and embodiment for robots acting with humans," Ph.D. dissertation, Massachusetts Institute of Technology, 2007.

[6] E. Fehr and K. M. Schmidt, "Theories of fairness and reciprocity-evidence and economic applications," 2001.

[7] V. Gabler, T. Stahl, G. Huber, O. Oguz, and D. Wollherr, "A game-theoretic approach for adaptive action selection in close proximity human-robot-collaboration," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, 2017, pp. 2897–2903.

[8] J. Mainprice, R. Hayne, and D. Berenson, "Goal set inverse optimal control and iterative replanning for predicting human reaching motions in shared workspaces," *IEEE Transactions on Robotics*, vol. 32, no. 4, pp. 897–908, 2016.

[9] K. P. Hawkins, S. Bansal, N. N. Vo, and A. F. Bobick, "Anticipating human actions for collaboration in the presence of task and sensor uncertainty," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, 2014, pp. 2215–2222.

[10] S. Nikolaidis, R. Ramakrishnan, K. Gu, and J. Shah, "Efficient model learning from joint-action demonstrations for human-robot collaborative tasks," in *2015 10th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, 2015, pp. 189–196.

[11] A. Cherubini, R. Passama, A. Crosnier, A. Lasnier, and P. Fraisse, "Collaborative manufacturing with physical human–robot interaction," *Robotics and Computer-Integrated Manufacturing*, vol. 40, pp. 1–13, 2016.

[12] M. C. Gombolay, R. A. Gutierrez, S. G. Clarke, G. F. Sturla, and J. A. Shah, "Decision-making authority, team efficiency and human worker satisfaction in mixed human–robot teams," *Autonomous Robots*, vol. 39, no. 3, pp. 293–312, 2015.

[13] S. Li and J. A. Shah, "Safe and efficient high dimensional motion planning in space-time with time parameterized prediction," in *2019 International Conference on Robotics and Automation (ICRA)*, 2019, pp. 5012–5018.

[14] S. Bansal, J. Xu, A. Howard, and C. Isbell, "A Bayesian Framework for Nash Equilibrium Inference in Human-Robot Parallel Play," in *Proceedings of Robotics: Science and Systems*, Corvalis, Oregon, USA, July 2020.

[15] R. Diankov, "Automated construction of robotic manipulation programs," Ph.D. dissertation, Carnegie Mellon University, Robotics Institute, August 2010.

[16] S. M. LaValle, "Rapidly-exploring random trees: A new tool for path planning," 1998.

[17] I. A. Şucan, M. Moll, and L. E. Kavraki, "The Open Motion Planning Library," *IEEE Robotics & Automation Magazine*, 2012.

[18] G. Hoffman, "Evaluating fluency in human–robot collaboration," *IEEE Transactions on Human-Machine Systems*, vol. 49, no. 3, pp. 209–218, 2019.

[19] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Ng, "Ros: an open-source robot operating system," 2009.

[20] G. Bradski and A. Kaehler, *Learning OpenCV: Computer vision with the OpenCV library*, 2008.

[21] I. Sucan and S. Chitta, "Moveit motion planning framework," 2019. [Online]. Available: http://moveit.ros.org

[22] M. L. Schrum, M. Johnson, M. Ghuy, and M. C. Gombolay, "Four years in review: Statistical practices of likert scales in human-robot interaction studies," in *Companion of the 2020 ACM/IEEE International Conference on Human-Robot Interaction*, ser. HRI '20. New York, NY, USA: Association for Computing Machinery, 2020, p. 43–52. [Online]. Available: https://doi.org/10.1145/3371382.3380739