Human-Aware Robotic Assistant for Collaborative Assembly: Integrating Human Motion Prediction With Planning in Time

Citation: Unhelkar, Vaibhav V. et al. "Human-Aware Robotic Assistant for Collaborative Assembly: Integrating Human Motion Prediction With Planning in Time." IEEE Robotics and Automation Letters 3, 3 (July 2018): 2394-2401 © 2016 IEEE

As Published: http://dx.doi.org/10.1109/lra.2018.2812906

Publisher: Institute of Electrical and Electronics Engineers (IEEE)

Persistent URL: https://hdl.handle.net/1721.1/124626

Version: Author's final manuscript: final author's manuscript post peer review, without publisher's formatting or copy editing

Human-Aware Robotic Assistant for Collaborative Assembly: Integrating Human Motion Prediction with Planning in Time

Vaibhav V. Unhelkar*,[1], Przemyslaw A. Lasota*,[1], Quirin Tyroller[2,3],
Rares-Darius Buhai[1], Laurie Marceau[3], Barbara Deml[2] and Julie A. Shah[1]

***Abstract*—Introducing mobile robots into the collaborative assembly process poses unique challenges for ensuring efficient and safe human-robot interaction. Current human-robot work cells require the robot to cease operating completely whenever a human enters a shared region of the given cell, and the robots do not explicitly model or adapt to the behavior of the human. In this work, we present a human-aware robotic system with single-axis mobility that incorporates both predictions of human motion and planning in time to execute efficient and safe motions during automotive final assembly. We evaluate our system in simulation against three alternative methods, including a baseline approach emulating the behavior of standard safety systems in factories today. We also assess the system within a factory test environment. Through both live demonstration and results from simulated experiments, we show that our approach produces statistically significant improvements in quantitative measures of safety and fluency of interaction.**

***Index Terms*—Physical Human-Robot Interaction, Collaborative Robots, Assembly**

I. Introduction

ROBOTS that work in proximity to or collaboratively with people have been a primary focus for robotics and automation in recent years [1]. Indeed, several robots that can safely operate alongside human collaborators have been recently developed and fielded for assembly applications [2]. However, despite this promising trend, the majority of collaborative robots within the manufacturing domain have a small operating region (e.g., the operating range for the UR10 is 1300 mm) and remain stationary. These limitations adversely impact the overall equipment effectiveness [3].

Mobile robots have a larger operating region than stationary robots, allowing for higher effectiveness and greater flexibility in the design of manufacturing processes. Here, we focus on the application of collaborative robots for delivering parts to human associates during automotive engine assembly. Such a system would need to fetch parts from depots and efficiently deliver them to humans at their workstations, while ensuring safe interactions (see ISO 10218/2 [4]).

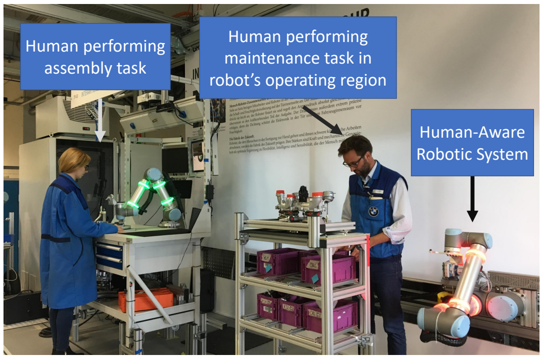

Human performing assembly task
Human performing maintenance task in robot's operating region
Human-Aware Robotic System

Fig. 1: The human-aware robotic system delivering parts to a human associate during an engine assembly task, while another human performing an unrelated task intercepts its path between the depot and the workstation.

We utilize a human-safe robot arm mounted on a linear axis unit; both the arm and unit are certified for and used in industrial applications. Prior work indicates the theoretical potential for a human-aware motion planner to yield improvements to both the safety and efficiency of human-robot interaction (HRI) [5]. This paper reports on CobotSAM, a human-aware robotic system that realizes this benefit. The system employs human motion prediction in conjunction with a complete, time-optimal path planner to execute motions in the shared environment (Fig. 1). The integrated system was successfully fielded in a BMW test environment that involved live interactions with human associates. The system was evaluated against three alternative methods, including a baseline approach that emulates the behavior of standard robot safety systems utilized in factories today. Through results from both live demonstrations and simulated experiments, we show that our integrated prediction and planning approach results in fewer safety-related stops, shorter task completion times, and improved measures of fluency of interaction.

For human motion prediction, collaboration with a mobile robot in the manufacturing domain requires accurate predictions over both short and long time horizons. For example, the robot must know where a person will be in the short term

Manuscript received: September, 10, 2017; Revised January, 3, 2018; Accepted February, 12, 2018.

This paper was recommended for publication by Editor Yasuyoshi Yokokohji upon evaluation of the Associate Editor and Reviewers' comments. This work was partially funded by BMW and supported by a NASA Space Technology Research Fellowship.

*These authors contributed equally to this work. Corresponding authors: plasota@mit.edu, unhelkar@csail.mit.edu.

[1]P. A. Lasota, V. V. Unhelkar, R.-D. Buhai and J. A. Shah are with the Massachusetts Institute of Technology, Cambridge, Massachusetts,

[2]Q. Tyroller and B. Deml are with the ifab, Karlsruher Institut für Technologie, Karlsruhe, Germany,

[3]Q. Tyroller and L. Marceau are with the Innovation Production Division, BMW Group, Munich, Germany.

Digital Object Identifier (DOI): see top of this page.

in order to maintain effective collision avoidance, but also must know the human's long-term predicted path in order to plan efficient motion toward its own goal. To accommodate this requirement, we employ the Multiple-Predictor System (MPS) [6], a data-driven approach that synthesizes a high-performance predictor using a library of component prediction methods, each with a unique performance profile that varies as a function of look-ahead time. The MPS enables automatic selection of the most accurate prediction approaches over both short and long time horizons.

The robot also requires a method for adapting its own behavior based on the knowledge of human behavior provided by the MPS – specifically, a planner that can leverage these predictions to generate motion for CobotSAM's linear axis unit. For example, if the robot receives predictions indicating that a human will cross in front of it, the robot can plan to either yield the way to the human or continue moving depending upon when the cross is predicted to occur. However, the system has limited freedom to perform such adaptations due to its single-axis mobility. This necessitates an approach that can generate plans quickly while reasoning about time and predictions. Schedule considerations drive the production environment and it is also crucial that the online planning system incorporates an explicit representation of time (i.e. performs planning in time). Thus, we use the Safe-Interval Path Planner (SIPP), a time-optimal search algorithm for planning in time, to plan robot trajectories [7].

SIPP generates plans under the assumption that the available predictions are fixed and accurate; however, in practice, predictions evolve as available information changes during task execution. The physical position of the robot will also change during the time-critical planning process. Hence, along with SIPP, we incorporate an algorithm to interleave prediction and planning with the execution of robot motion.

The key contributions of this paper are:

1) The first robot system to employ complete, time-optimal path planning in time in conjunction with a multiple predictor system for human motion. The integrated system interleaves prediction, planning, and execution to produce anticipatory robot behaviors that are derived automatically as the robot interacts with a live human.
2) The first physical demonstration of such a prediction, planning, and execution system using an arm and linear axis unit, both certified for and used in industrial settings.
3) Evaluation in simulation to assess improvements to safety and efficiency, compared to state-of-the art approaches applicable to factory environments. Results demonstrated reductions in safety-related stops, decreases in task times, and improvements in measures of fluency of interaction.

The application considered in this paper involves navigation along a linear axis within a factory environment; however, the algorithms provided herein are also applicable to general human-robot co-navigation.

II. Task Description

During the final assembly of automotive engines, human workers must move within their environment to fetch the parts required for the assembly process. The use of mobile collaborative robots to perform this task offers an alternative that allows human workers to focus on the dexterous, value-added work of engine assembly, yielding significant time and cost savings [8]. Mobility achieved through translation on a linear axis [9] provides a near term opportunity for new forms of human-robot collaboration using existing hardware and control solutions, which come with safety guarantees that make them readily suitable for a production environment.

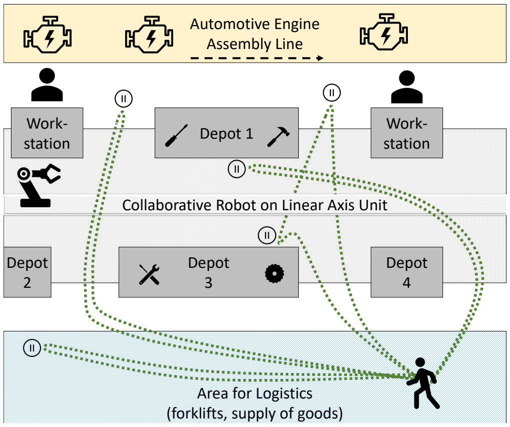


Fig. 2: Schematic of a prototypical factory environment for deploying our robotic system. The collaborative robot moves along the linear axis unit to assist humans at their workstations. The robot's operating region (orange) is shared with human workers. The set of potential human motions is represented by green dotted lines.

Delivery of assembly parts: Figure 2 presents a schematic of a prototypical factory environment in which the robot is deployed. The primary task of the system is to transit between workstations and part depots to deliver parts to human associates. These humans remain at their workstations throughout the task and use the parts delivered by the robot to assemble the engine. As engine assembly is a repetitive task, the parts and the order in which they are to be delivered remains fixed – i.e., the robot's task plan (or, equivalently, the sequence of goal locations) is pre-specified.

Sharing environment with humans: The set of possible human motions within the human-robot shared region is depicted in Fig. 2. The humans moving within this region differ from those at the workstations, and include workers responsible for stocking part depots and cleaning. While the sequence of robot goal locations is predetermined, it requires algorithms to plan and execute its trajectory to each goal. Humans within the shared environment may exhibit arbitrary motions (from the known set of possible motions) at any time during the robot's operation. Thus, we provide a system that employs algorithms for human motion prediction and trajectory planning in time for CobotSAM's base to achieve safe and efficient execution of the part delivery task.

III. Related Work

A. Collaborative Robots in Automotive Manufacturing

In recent years, multiple robotic systems have been researched and developed for operation among humans in automotive final assembly [10], [11]. These include stationary robots designed to provide ergonomic benefits to human associates during assembly tasks, such as cockpit installation [12], rear axle assembly [13], or door sealant application [2]. A few robots with varied degree of mobility have also been researched and developed [14]–[17], including “Robot Workmate” a system with single-axis mobility for inspection tasks [18]. Similarly, our system achieves the desired degree of mobility using a linear axis unit. However, in contrast to existing mobile collaborative robots, ours incorporates human motion prediction and path planning in time, enabling safe and anticipatory behavior within a shared environment.

B. Human Motion Prediction

Tasks executed within the manufacturing context have an inherent structure, with people moving between key locations within their environment to achieve specific goals. One common approach to predicting human motion in such structured environments centers around inferring the actions or goals of humans in order to predict their future motion. This body of work can be split into two main groups, based on the length of the prediction time horizon.

The first subset focuses on prediction over short time horizons, often in the context of reaching motions. For instance, researchers have developed techniques utilizing inverse optimal control [19] and Bayesian classification [20] to generate motion predictors using labeled data of arm motion. To avoid the need for manually labeled motions, Luo et al. presented an unsupervised approach to predicting reaching motions using Gaussian mixture models of the palm and arm [21]. Besides considering only the human motion, other works also reason about environmental constraints and object affordances to predict human motion [22].

The second subset of work related to goal-based human motion prediction deals with predictions over longer time horizons, and is typically focused on ambulatory motion (e.g., [23]). One such approach leverages the assumption that people move in an efficient manner while navigating an environment in order to model human motions using maximum entropy inverse optimal control [24]. Environmental constraints and biases can also be leveraged to predict walking motion, as shown in work by Karasev et al. [25]. Finally, recent work by Chen et al. improved the prediction of pedestrian motion by addressing issues arising from the modeling of human motions with Gaussian process regression by developing a framework called the Dirichlet process active region [26].

In the context of HRI within the manufacturing domain, there is a need for accurate motion predictions over both short and long time horizons. While the above methods work well in their respective domains, they are not necessarily suitable for prediction across the entire time horizon spectrum. Results from prior work indicate that utilizing a set of complimentary prediction approaches for different time horizons results in improved prediction accuracy [6].

C. Path Planning between Humans

For collaborative robots, the need for human-aware task and motion planning has become evident [27]. Sisbot et al. provided one of the first human-aware motion planners for robot navigation by explicitly accounting for human preferences [28]; however, their planner does not utilize any predictive information regarding humans. Recently, several factors have been incorporated for human-aware planning, including gaze [29], legibility [30], and proxemics [31], [32].

Predictions of human motion have also been previously used for robot planning. Ziebart et al. shaped a robot’s navigation cost according to predictions in order to mitigate potential collision points, and used time-independent path planners to compute collision-free paths [24]. In contrast, we explore planners that utilize explicit representation of time.

While our robotic system is not expected to operate in crowded regions, techniques of note for navigation within crowds have been developed [33], [34]. By leveraging the cooperative effect of robot motion on human motion, Trautman et al. avoid highly conservative robot behavior caused due to prediction uncertainty in dense environments [35].

Timing is a critical component for HRI [36], [37], and is of particular importance for planning and executing the motion of our system, due to its single-axis mobility. However, the computational burden of path planning with explicit modeling of time becomes prohibitively large [24] – even more so for long time horizons encountered while planning paths in factory environments. Recently, two approaches for efficient path planning with explicit modeling of time have been explored. In their work, Khambhaita et al. first generated the global plan of the robot’s motion without considering human motion, and then modified the execution of this path using timed elastic bands [38]. However, this approach does not provide any guarantees with regard to path optimality.

An alternate approach that does provide such guarantees is to pose planning in time as a graph search problem [39]. In their work, Phillips et al. [7] provided SIPP, a computationally attractive approach for this problem while maintaining the optimality of the resulting plan. Evaluation of SIPP assumed perfect prediction of the environment, and thus involved execution of a single plan generated prior to motion execution. However, in practice, predictions change during execution, requiring approaches for both online replanning and the interleaving of execution with planning. As discussed in Section VI, we use and develop upon SIPP to provide these elements for our robotic system.

IV. System Overview

In this section, we briefly summarize the various components of our human-aware robotic system (see Figure 3).

1) Physical Robot: A UR10 collaborative robot serves as the robotic arm for performing manipulation tasks. Desired system mobility is achieved by mounting the arm on a linear axis unit. Arm joint angles are controlled using Universal Robot on-board controllers, and the arm is held in a fixed configuration while in motion due to the linear axis unit.

2) Safety System: A 2-D laser scanner is mounted on the robot; the scanner triggers a safety stop when any human (or object) is within the safety radius of the robot. A threaded implementation is used for the safety system. Once the stop

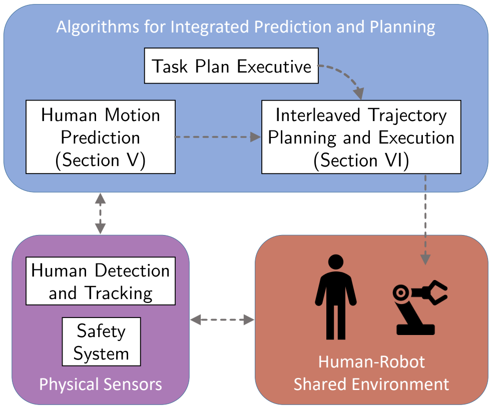

Algorithms for Integrated Prediction and Planning
Task Plan Executive
Human Motion Prediction (Section V)
Interleaved Trajectory Planning and Execution (Section VI)
Human Detection and Tracking
Safety System
Physical Sensors
Human-Robot Shared Environment

Fig. 3: An overview of the human-aware robotic system.

is triggered, the entire physical robotic system is rendered immobile until the human leaves the safety radius.

3) Human Motion Prediction: Humans within the shared region are tracked using a Kinect sensor and the OpenNI tracker within Robot Operating System (ROS) [40]. The prediction sub-system uses human detection and the algorithms described in Section V to provide predictions of human motion, updated at a frequency of 5 Hz. The 2D coordinates of the human's head serve as features for the predictors.

4) Trajectory Planning and Execution: Once a goal location is issued, the system described in Section VI generates a plan for CobotSAM's base and executes the robot motion. The system checks for changes in predictions and replans at a frequency of 10 Hz.

5) Communication between Sub-Systems: Communication between the physical robot components (UR10, linear axis unit and the sensors) and the software is implemented using a programmable logic controller, TCP/IP sockets and ROS. A task plan executive issues a predetermined sequence of goals, based on the task plan, to initiate the robot task.

V. Human Motion Prediction

1) Challenges: One of the main challenges of predicting human motion in the context of collaboration with a mobile robot within a factory setting is that accurate predictions are necessary over both short and long time horizons. Short-term predictions are critical for effective collision avoidance, as the robot must have accurate knowledge of where a person will be in the immediate future in order to stop or execute an evasive motion. While a safety system can serve as a fallback to prevent collisions, it is not a sufficient solution for ensuring that interaction feels safe and comfortable for human workers [5] – an essential requirement for human-robot interaction [41]–[43]. As efficiency is also imperative within the manufacturing domain, there is also a need for accurate long-term predictions. These predictions, when combined with planning in time, allow the robot to make intelligent decisions about how to move toward its own goal in a manner that minimizes interference with humans.

In the scenario considered, the humans whose motions the system is trying to predict are associates passing through the shared space who are not actively involved in the robot's task. As such, from the robot's perspective, the motions executed in the work cell occur in a random order, meaning that human task sequence information cannot be leveraged for prediction.

2) Multiple-Predictor System: In order to accommodate the need for accurate human motion prediction in both the short and long term, we utilize the Multiple-Predictor System (MPS) [6]. This method uses given motion data to learn how to best combine a set of complementary prediction methods based on their relative performances at various time horizons of interest. One benefit of this data-driven approach is that it is designed to generalize to different types of tasks and motions. While in [6] the MPS was evaluated on short reaching motions (1.88±0.48 s) with a prediction time horizon range of 0.05-0.5 s, in this work we demonstrate and evaluate its use for long, ambulatory motions (16.05±2.41 s) with a prediction time horizon ranging from 0.1-6.0 s.

The implemented version of the MPS is composed of three prediction methods. The first method, velocity-based position projection (VBPP), estimates future locations by projecting the human's current position through an estimate of his or her velocity as computed via the Savitzky-Golay Filter [44]. Once the velocity is estimated, the VBPP assumes that the person will continue moving at that speed for the duration of the time horizon of interest. The second predictor is the time series classification (TSC) method, which builds upon the goal-based time series classification approach presented by Pérez-D'Arpino and Shah [20]. The final predictor in the MPS is a sequence prediction (SP) method that reasons on observed sequences of actions instead of the motion itself. The sequence prediction method uses previously observed action sequences to learn which sets of actions occur before others [45]. As described in [6], for both the TSC and SP methods, the original approaches were extended to predict a human's future positions by identifying a point on the mean trajectory of the predicted action that corresponds with the current location of the human, and advancing forward by the queried time horizon.

The MPS is trained via a two-stage process. First, a subset of the data is used to train the parameters of the individual prediction methods as a function of a discrete set of prediction time horizons. Next, a second subset of the data is utilized in a predictor fusion technique based on the Polynomial Weights algorithm [46], where the loss function for each predictor i is defined as the magnitude of the prediction error normalized by the sum of the mean and standard deviation of the prediction errors encountered during training, with an upper limit of 1: $L_t^i = min\left\{\frac{\|\widehat{\boldsymbol{x}}^i - \boldsymbol{x}\|}{\mu+\sigma}, 1\right\}$.

VI. Trajectory Planning and Execution

1) Challenges: Given a robot goal, one approach for robot motion is to execute pre-programmed paths and maintain safety through reactive systems [47]; this is the method used for repetitive robot motion within many factories today. However, as shown in Sec. VII, such a system yields poor task efficiency and fluency of interaction in the absence of a means to anticipate and adapt to human behavior.

Our system, thus, leverages human motion predictions to generate robot motions. Planning with predictions, however,

Algorithm 1 Interleaving Planning and Execution

```
1: procedure MAIN($R_{\text{goal}}$)
2:     $H_{\text{predictions}} \leftarrow \text{MPS}()$                                   ▷ Get predictions
3:     $R_{\text{start}} \leftarrow R_{\text{position}}$                                    ▷ Get robot start state
4:     $R_{\text{trajectory}} \leftarrow \text{SIPP}(R_{\text{start}}, R_{\text{goal}}, H_{\text{predictions}})$   ▷ Plan robot trajectory
5:     while Goal not reached do                            ▷ Executed at replanning rate
6:         if Safety stop triggered then
7:             return False
8:         Update robot command ($R_{\text{command}}$) using $R_{\text{trajectory}}$
9:         Issue $R_{\text{command}}$ to hardware controller
10:        $H_{\text{predictions}} \leftarrow \text{MPS}()$                               ▷ Update predictions
11:        replan ← Predictions changed or $R_{\text{trajectory}}$ is empty
12:        if replan then
13:            $R_{\text{start}} \leftarrow R_{\text{command}}$                             ▷ Update start state
14:            $R_{\text{new-trajectory}} \leftarrow \text{SIPP}(R_{\text{start}}, R_{\text{goal}}, H_{\text{predictions}})$   ▷ Replan
15:            if PLANCHANGED ($R_{\text{trajectory}}, R_{\text{new-trajectory}}$) then
16:                $R_{\text{trajectory}} \leftarrow R_{\text{new-trajectory}}$
17:                Issue stop $R_{\text{command}}$ to hardware controller
18:    return True
```

can be computationally expensive. State-of-the-art human-aware collaborative manipulation systems typically utilize short-term predictions [48]; however, predictions involving significantly longer time horizons may be useful for planning long robot paths ($\approx$ 10m) in a factory setting. Further, due to its single-axis mobility, our robotic system has limited freedom to adapt to the behavior of nearby humans. If the planning time is lengthy and the robot reacts to human too late, it might be unable to exhibit anticipatory behavior despite the existence of predictions that include the desired anticipatory information. This emphasizes the need for computationally tractable approaches that account for the critical effect of timing for path planning, and allow for interleaving online planning with execution.

2) Safe-Interval Path Planning: To account for the time-critical nature of the robot planning problem, we use a representation that explicitly models time. We incorporate SIPP as the underlying planner for our system [7]. By using safe time-intervals, SIPP significantly reduces the cardinality of the state space and provides a computationally efficient approach for planning despite explicit modeling of time. Using time-indexed predictions of human motion, SIPP provides a feasible robot trajectory if one exists, and returns a failure otherwise. Given accurate predictions, SIPP is both complete and time-optimal. We implement the planner as an extension to the Search-Based Planning Library [49] and ROS.

3) Interleaving Planning and Execution: Due to changing human motion predictions, SIPP alone is not sufficient for executing robot motion. SIPP assumes the predictions to be accurate, and thus does not include a mechanism for updating the robot plan online. To incorporate the latest predictions, our robotic system additionally requires online replanning, during which the robot itself may be in motion. It is undesirable to stop the robot during replanning; therefore, an approach to interleave replanning and execution is required.

We provide Algorithm 1 in order to achieve interleaved planning and execution. Upon receiving the goal location as input, Algorithm 1 uses the current predictions from the MPS, and the start state of the robot, to compute the robot's trajectory via SIPP (lines 1-4). To interleave trajectory execution and planning, motion commands are executed using the planned trajectory, which is updated at the replanning rate (lines 5-17). The algorithm returns a failure if a safety stop is triggered during execution (lines 6-7); to complete the goal once the stop is deactivated, the goal is reissued in order to reinitiate Algorithm 1.

As long as the goal is not reached and a safety stop is not triggered, motion commands are sent to the hardware controller according to the latest planned trajectory (lines 8-9). If SIPP could not identify a feasible solution a stop command is issued to the robot. The robot then updates its safe time-interval representation according to the latest available predictions (line 10). If the predictions change, or if the previous planner call returned a failure, replanning is performed (line 12-17). To account for change to the robot state (due to motion) between when the planner is called and the resulting plan is executed, the latest commanded pose of the robot is used as the start state for replanning with SIPP.

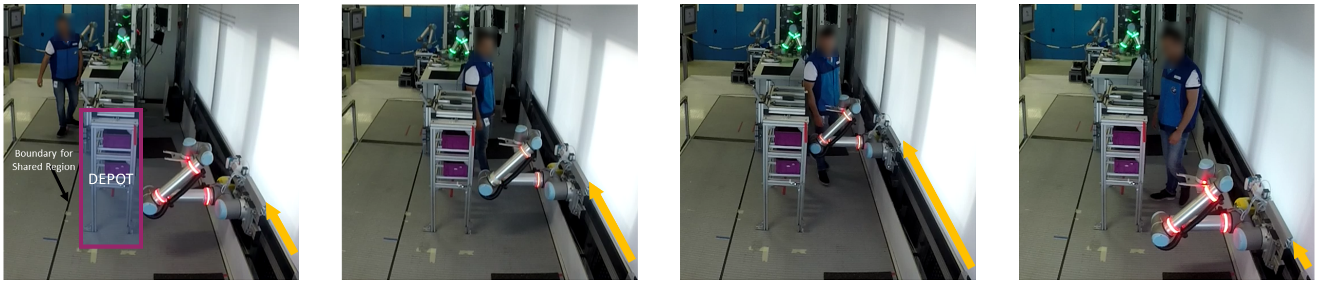
Human motion predictions continue to evolve over the course of task execution; however, not all changes to predictions result in plan changes. The new plan, generated in line 14, is thus compared against the plan currently being executed via the PLANCHANGED() method. The method returns success if either the trajectory lengths differ by a time threshold (1s) or the L^{∞}-norm of the difference in the two trajectories exceeds a distance threshold (0.5m), and prevents fluctuations in robot trajectory due to minor updates to predictions. If PLANCHANGED() returns success, the robot is stopped and the updated trajectory is used for execution from the following timestep. The algorithm returns success once the goal has been reached (line 17). By interleaving planning in time and motion execution, our system can adapt quickly and efficiently to the behavior of nearby humans.

VII. SYSTEM EVALUATION

A. Physical System Demonstrations

We demonstrated our system using the physical robots described in Section IV within the environment depicted in Fig. 1 in a BMW test environment. In our demonstrations, the robot operated in one of three modes. In all modes, the current task and goal of the human was unknown to the robot. In the "Baseline" mode, no information about the human's current or predicted position is given to the planner, and the robot simply pauses its motion whenever a human enters the shared area of the work cell. This mode emulates the behavior of state-of-the-art reactive safety systems designed for factory environments, such as SafetyEye [47]. In the "Planning with Detection" mode, the planner incorporates the human's current position by assuming that the human will remain in that location until a new position is received. Finally, in our approach (the "Planning with Prediction" mode), the robot uses both the currently detected human location and a set of position predictions obtained from the MPS. These predictions are made at a discrete set of time horizons ranging from 0.1-3 s in increments of 0.1 s, and are recomputed at a rate of 5 Hz. Video attachments representing the operation of these three modes are included with the paper and also available at http://tiny.cc/cobotSAM (see Fig. 4 for illustrative snapshots).

Through the results from our integrated system demonstrations, we observed that our system can anticipate and



(a) Start configuration. (b) Baseline mode. (c) Planning with detection. (d) Planning with prediction.

Fig. 4: Stills of the three modes (b-d) from the factory test environment demonstration. The robot's task is to navigate to the other side of the linear axis, while a human attempts to go to the depot. The black line parallel to the rail denotes the boundary of the shared region. The difference in the robot's position on the linear axis (yellow arrow) at the start (a) and between the three modes (b-d) when the human arrives at the depot illustrates the anticipatory behavior of our approach (d).

adapt to the behavior of nearby humans when utilizing both prediction and planning techniques. In one example of an observed adaptive behavior, the robot paused its motion and even moved backwards (Fig. 4(d)) to allow a human to reach the depot, then automatically resumed its task after the person moved back toward the workbench. Importantly, this behavior was automatically derived during execution, without the need for preprogramming. In contrast, when planning without anticipation of human motion, the robot was unable to effectively adapt its motion to unexpected human behavior, resulting in less fluent interaction, with the robot blocking the human's path and triggering safety stop more frequently (Fig. 4(c)). Lastly, while in the Baseline mode, the system yielded safe but inefficient robot motion (Fig. 4(b)), as the robot often stopped unnecessarily, sometimes in a position that interfered with the human's path.

B. *Evaluation in Simulation*

While the physical demonstrations provide intuitive, qualitative examples of the benefits of our system, the context for these demonstrations consisted of one robot assisting a single worker in a small work cell, which is not representative of how a robotic assistant would be deployed in a real factory. Therefore, we also performed a more thorough, quantitative analysis of the benefits of our system in simulation, with a much larger work cell in an analogue factory environment, as depicted in Fig. 2.

In the simulation, the length of the rail was 10 m and the shared workspace was 10 m by 3.1 m. The robot performed a pre-set sequence of tasks, simulating the pickup and delivery of components between depots and workbenches. The robot utilized the planning approach described in Section VI to plan its motions toward the task plan's goal locations. We used a grid size of 10 cm and a replanning rate of 10 Hz for the planner, and set the maximum speed of the robot to 1 m/s. We simulated the laser scanner safety system by stopping the robot whenever a human came within a safety radius of 0.75 m. The simulated robot operated in the same three modes as those used during the physical demonstrations: Baseline, Planning with Detection, and Planning with Prediction. Due to the larger workspace and longer trajectories, the discrete set of time horizons at which the MPS made predictions was extended to a range of 0.1-6 s in increments of 0.1 s, with the same prediction recomputation rate as before (5 Hz).

In order to make the simulated human motions and predictions more realistic, we collected human walking trajectories via a motion capture system. We defined a set of four possible human actions corresponding to the four trajectories depicted in Fig. 2. The "pause" symbols in the figure represent places where the human would pause for ≈ 3 seconds before continuing along their trajectory, which is intended to simulate a worker stopping in order to perform a task (e.g., picking up a tool or reading a value from a monitor). Two participants performed each of the four motions 10 times each, while the 2D position and orientation of their head was recorded at a rate of 120 Hz. The trajectories were then downsampled to 10 Hz and used for training and evaluation of the MPS, with 50% of the trajectories serving to train the individual predictor parameters, 20% for the predictor fusion, and 30% for use in the simulation.

We ran a total of 30 trials in each of the three robot modes, with a different sequence of simulated human actions occurring in each trial. Each sequence consisted of a random permutation of eight actions, with each of the four actions occurring twice. The motion of the human while performing each action in the sequence was simulated by playing back a sample trajectory of that action chosen at random from the holdout set. To further improve the realism of our simulation and model variability in human motion, we also incorporated a waiting behavior for the simulated human whenever the robot was in its path. Specifically, when the human was within the robot's safety radius and the human's approach angle towards the robot was less than 30°, the simulated human would pause for a period between 4 and 5 seconds sampled from a uniform distribution, and then resume motion at 50% of the original speed until clear of the robot. This slower resumption of motion is intended to simulate a human carefully moving past the robot after the initial stop.

C. *Simulation Results and Discussion*

1) Multiple-Predictor System: We assessed the performance of the individual predictors and the MPS with a leave-one-out cross-validation. For each iteration of the cross-validation, we held out one set of demonstrations for testing (one example of each action), 13 for training, and 6 for

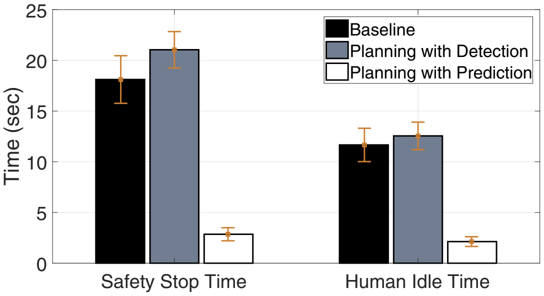


Fig. 5: Simulation outcomes (mean and std. error) for human idle times and safety stop times (the time during which the robot was idle due to a safety stop) across the three modes.

model selection. The overall mean prediction errors across all time horizons and iterations for the velocity-based position projection (VBPP), time series classification (TSC), sequence prediction (SP), and MPS were 181.9 cm, 43.0 cm, 174.3 cm, and 41.2 cm, respectively. We applied the Friedman test to verify that the prediction method had a significant effect on these prediction errors ($p < 0.001$, $\chi^2 = 50.22$) and then used the Wilcoxon signed rank test to perform pairwise comparisons to the MPS. Given the above averages, the MPS outperformed the VBPP and SP methods by a large margin, and exhibited a small improvement over the TSC method (VBPP and SP: $p < 0.001$; TSC: $p = 0.015$). As expected, due to the randomness of the human motion sequences, the sequence prediction method performed poorly across all time horizons. Consequently, the MPS was composed of only the other two methods, with the intuitive assignment of the VBPP method for short time horizons (0.1-0.6 s) and TSC method for the remaining time horizons (0.7-6 s).

2) Safety, Efficiency, and Fluency: Using the simulation, we examined various objective measures of human safety, task efficiency, and fluency of interaction. To assess the statistical significance of our results, we applied the Friedman test to determine the effect of the given robot mode (Baseline, Planning with Detection, or Planning with Prediction) on these measures, and the Wilcoxon signed rank test to assess pairwise comparisons. Results from these evaluations are summarized in Fig. 5 and Table I.

One key measure of human safety in our evaluation is the number of times the robot's safety stop was triggered – i.e., the inability of the robotic system to anticipate and avoid the movement of nearby humans. We observed that as compared to both the Baseline and Planning with Detection modes, which resulted in an average of 3.5 and 5.1 safety stop triggers, respectively, our system resulted in fewer safety stops (mean 0.5). This effect is statistically significant, and is also evident through the correlated measure of safety stop time depicted in Fig. 5. When compared with the Baseline mode, our system also statistically significantly shortened both idle and task times for both the human and the robot, with human and robot idle times reduced by 81.8% and 44.1%, respectively, and task times reduced by 5.8% and 16.9%, respectively. This demonstrates that not only did the incorporation of prediction and planning result in fewer safety stop triggers, but also improved task efficiency.

TABLE I: Simulation Results[a]

Dependent Variable	Baseline	Planning + Detection	Planning + Prediction	Friedman Test
Safety Stop Triggers (#)	3.5 $p < 0.001$	5.1 $p < 0.001$	0.5	$\chi^2 = 50.04$ $p < 0.001$
Human Idle Time (s)	11.7 $p < 0.001$	12.6 $p < 0.001$	2.13	$\chi^2 = 28.92$ $p < 0.001$
Robot Idle Time (s)	91.7 $p < 0.001$	46.4 $p = 0.086$	51.3	$\chi^2 = 46.07$ $p < 0.001$
Human Task Time (s)	165.2 $p < 0.001$	168.1 $p < 0.001$	155.7	$\chi^2 = 27.27$ $p < 0.001$
Robot Task Time (s)	204.0 $p < 0.001$	159.6 $p = 0.002$	169.5	$\chi^2 = 51.67$ $p < 0.001$
Safety Stop Time (s)	18.1 $p < 0.001$	21.0 $p < 0.001$	2.85	$\chi^2 = 37.49$ $p < 0.001$

[a] Mean values of the dependent variables. The p values in the Baseline and Planning + Detection columns correspond with the pairwise comparisons between these modes and Planning + Prediction mode.

Compared with the Planning with Detection mode, our system also statistically significantly reduced human idle and task times by 83.1% and 7.4%, respectively. Interestingly, the Planning with Detection mode resulted in robot idle and task times comparable to our own approach. The total robot idle time is composed of the sum of time the robot stopped due to the safety system being engaged (specified as "Safety Stop Time" in Table I) and the time during which the planner commanded the robot to pause. On further inspection, we observed that the robot idle time due to safety stops is insignificant for our system as compared with that observed while the robot operated under the Planning with Detection mode. Indeed, this unplanned idle time contributed to only 5.56% of total robot idle time when using predictions, whereas its contribution increased to 45.33% when using only detections. This indicates that, although the robot remains idle using our approach for a similar amount of time as that observed in the Planning with Detection mode, this behavior was due to the planner commanding the robot to pause in order to yield to the human, improving safety and reducing human idle time.

3) Interleaving Prediction, Planning and Execution: We conducted an additional 30 simulation trials in which the robot planned its motion using SIPP but without interleaving planning and execution (similar to how SIPP was employed in [7]). The robot created a plan using SIPP and available predictions at the outset of its motion execution and executed it till either the goal was reached or safety stop was triggered. As compared to Planning with Prediction, this "SIPP-baseline" mode resulted in higher numbers of safety stops (5.5, $p < 0.001$) and cumulative duration (19.48 s, $p < 0.001$), higher human idle time (10.8 s, $p < 0.001$) and task time (173.0 s, $p < 0.001$), and a higher ratio of unplanned idle time (42.5%).

D. Future Directions

Our evaluations demonstrated safe and efficient performance of the CobotSAM system, and raise a number of directions for future research. For instance, the prediction sub-system assumed that all human actions come from a known set and was not designed to handle previously unseen motions. While the constrained nature of the factory domain lent itself well to this assumption, anomalous motions can still occur. To address this issue, we plan to incorporate the ability to recognize unmodeled motion and adjust the composition of the MPS online in future work. Further, the robot's trajectory planner replanned from scratch once new predictive information was made available. Hence, we are investigating the development of incremental planners for planning in time that can replan efficiently by reusing their previous planning process.

VIII. Conclusion

We present CobotSAM, a human-aware robotic assistant designed to deliver parts to human associates performing dexterous assembly tasks. The robot is equipped with algorithms for prediction of the motion of nearby humans, along with a planning algorithm that leverages these predictions by planning in time. We demonstrate the efficacy of our system in a BMW test environment. Through the use of these algorithms, our system exhibits anticipatory behavior and results in safe and efficient execution of a part-delivery task while sharing its environment with humans in the factory.

References

[1] "National Robotics Initiative 2.0: Ubiquitous Collaborative Robotics," http://www.nsf.gov/funding/pgm_summ.jsp?pims_id=503641.

[2] "Innovative human-robot cooperation in BMW Group Production," https://www.press.bmwgroup.com/global/article/detail/T0209722EN/.

[3] R. Hansen, *Overall equipment effectiveness*. Industrial Press, 2001.

[4] "ISO 10218-2:2011: Robots and robotic devices – Safety requirements for industrial robots," https://www.iso.org/standard/41571.html.

[5] P. A. Lasota and J. A. Shah, "Analyzing the Effects of Human-Aware Motion Planning on Close-Proximity Human-Robot Collaboration," *Human Factors*, vol. 57, no. 1, pp. 21–33, Jan 2015.

[6] ——, "A multiple-predictor approach to human motion prediction," in *ICRA*. IEEE, 2017.

[7] M. Phillips and M. Likhachev, "SIPP: Safe interval path planning for dynamic environments," in *ICRA*. IEEE, 2011.

[8] Harvard Business Review, "Smarter smaller safer robots," http://hbr.org/2015/11/smarter-smaller-safer-robots, [Online].

[9] "KUKA linear units," https://www.kuka.com/en-us/products/robotics-systems/robot-periphery/linear-units, [Online].

[10] "Audi Smart Factory," https://www.audi-mediacenter.com/de/amtv/videos/2175, [Online].

[11] S. Sheikholeslami, A. Moon, and E. A. Croft, "Cooperative gestures for industry: Exploring the efficacy of robot hand configurations in expression of instructional gestures for human–robot interaction," *IJRR*, vol. 36, pp. 699–720, May 2017.

[12] P. Akella, M. Peshkin, E. Colgate, W. Wannasuphoprasit, N. Nagesh, J. Wells, S. Holland, T. Pearson, and B. Peacock, "Cobots for the automobile assembly line," in *ICRA*. IEEE, 1999.

[13] R. D. Schraft, C. Meyer, C. Parlitz, and E. Helms, "Powermate-a safe and intuitive robot assistant for handling and assembly tasks," in *ICRA*. IEEE, 2005.

[14] "Plug and Navigate robots for smart factories," http://www.pan-robots.eu/, [Online; last accessed 1-Sept-2017].

[15] B. Hamner, S. Koterba, J. Shi, R. Simmons, and S. Singh, "An autonomous mobile manipulator for assembly tasks," *Autonomous Robots*, vol. 28, no. 1, pp. 131–149, 2010.

[16] V. V. Unhelkar, J. Perez, J. C. Boerkoel Jr, J. Bix, S. Bartscher, and J. A. Shah, "Towards control and sensing for an autonomous mobile robotic assistant navigating assembly lines," in *ICRA*. IEEE, 2014.

[17] F. Rovida and V. Krüger, "Design and development of a software architecture for autonomous mobile manipulators in industrial environments," in *Intl. Conf. on Industrial Technology*. IEEE, 2015.

[18] R. Müller, M. Vette, and M. Scholer, "Robot workmate: A trustworthy coworker for the continuous automotive assembly line and its implementation," *Procedia CIRP*, vol. 44, pp. 263–268, 2016.

[19] J. Mainprice, R. Hayne, and D. Berenson, "Predicting human reaching motion in collaborative tasks using Inverse Optimal Control and iterative re-planning," in *ICRA*. IEEE, 2015.

[20] C. Pérez-D'Arpino and J. A. Shah, "Fast Target Prediction of Human Reaching Motion for Cooperative Human-Robot Manipulation Tasks using Time Series Classification," in *ICRA*. IEEE, 2015.

[21] R. Luo, R. Hayne, and D. Berenson, "Unsupervised early prediction of human reaching for human-robot collaboration in shared workspaces," *Autonomous Robots*, pp. 1–18, Jul 2017.

[22] H. S. Koppula and A. Saxena, "Anticipating Human Activities using Object Affordances for Reactive Robotic Response," in *R:SS*, 2013.

[23] V. V. Unhelkar, C. Prez-D'Arpino, L. Stirling, and J. A. Shah, "Human-robot co-navigation using anticipatory indicators of human walking motion," in *ICRA*. IEEE, 2015.

[24] B. D. Ziebart, N. Ratliff, G. Gallagher, C. Mertz, K. Peterson, J. A. Bagnell, M. Hebert, A. K. Dey, and S. Srinivasa, "Planning-based prediction for pedestrians," in *IROS*. IEEE, 2009.

[25] V. Karasev, A. Ayvaci, B. Heisele, and S. Soatto, "Intent-aware long-term prediction of pedestrian motion," in *ICRA*. IEEE, 2016.

[26] Y. Chen, M. Liu, S.-Y. Liu, J. Miller, and J. P. How, "Predictive Modeling of Pedestrian Motion Patterns with Bayesian Nonparametrics," in *AIAA Guidance, Navigation, and Control Conference*, 2016.

[27] T. Kruse, A. K. Pandey, R. Alami, and A. Kirsch, "Human-aware robot navigation: A survey," *Robotics and Autonomous Systems*, vol. 61, no. 12, pp. 1726–1743, 2013.

[28] E. A. Sisbot, L. F. Marin-Urias, R. Alami, and T. Simeon, "A human aware mobile robot motion planner," *IEEE Transactions on Robotics*, vol. 23, no. 5, pp. 874–883, 2007.

[29] D. V. Lu and W. D. Smart, "Towards more efficient navigation for robots and humans," in *IROS*. IEEE, 2013.

[30] A. D. Dragan, K. C. Lee, and S. S. Srinivasa, "Legibility and predictability of robot motion," in *HRI*. IEEE, 2013.

[31] C. Dondrup, N. Bellotto, M. Hanheide, K. Eder, and U. Leonards, "A computational model of human-robot spatial interactions based on a qualitative trajectory calculus," *Robotics*, vol. 4, no. 1, 2015.

[32] R. Mead and M. J. Matarić, "Autonomous human–robot proxemics: socially aware navigation based on interaction potential," *Autonomous Robots*, vol. 41, no. 5, pp. 1189–1201, 2017.

[33] M. Kuderer, H. Kretzschmar, C. Sprunk, and W. Burgard, "Feature-based prediction of trajectories for socially compliant navigation," in *R:SS*, 2012.

[34] H. Kretzschmar, M. Spies, C. Sprunk, and W. Burgard, "Socially compliant mobile robot navigation via inverse reinforcement learning," *IJRR*, vol. 35, no. 11, pp. 1289–1307, 2016.

[35] P. Trautman, J. Ma, R. M. Murray, and A. Krause, "Robot navigation in dense human crowds: Statistical models and experimental studies of human–robot cooperation," *IJRR*, vol. 34, no. 3, pp. 335–356, 2015.

[36] G. Hoffman, M. Cakmak, and C. Chao, "Workshop on timing in human-robot interaction," in *HRI*. ACM, 2014.

[37] F. Broz, I. R. Nourbakhsh, and R. G. Simmons, "Planning for Human-Robot Interaction Using Time-State Aggregated POMDPs," in *AAAI*, 2008.

[38] H. Khambhaita and R. Alami, "A human-robot cooperative navigation planner," in *HRI*. ACM, 2017.

[39] M. Kollmitz, K. Hsiao, J. Gaa, and W. Burgard, "Time dependent planning on a layered social cost map for human-aware robot navigation," in *European Conference on Mobile Robots*. IEEE, 2015.

[40] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "ROS: an open-source Robot Operating System," in *ICRA Workshop on Open Source Software*, 2009.

[41] X. Broquere, D. Sidobre, and I. Herrera-Aguilar, "Soft motion trajectory planner for service manipulator robot," in *IROS*. IEEE, 2008.

[42] T. Arai, R. Kato, and M. Fujita, "Assessment of operator stress induced by robot collaboration in assembly," *CIRP Annals - Manufacturing Technology*, vol. 59, no. 1, pp. 5–8, Jan 2010.

[43] E. A. Sisbot, L. F. Marin-Urias, X. Broquère, D. Sidobre, and R. Alami, "Synthesizing Robot Motions Adapted to Human Presence," *International Journal of Social Robotics*, vol. 2, no. 3, Jun 2010.

[44] A. Savitzky and M. J. Golay, "Smoothing and differentiation of data by simplified least squares procedures." *Analytical chemistry*, vol. 36, no. 8, pp. 1627–1639, 1964.

[45] B. Letham, C. Rudin, and D. Madigan, "Sequential event prediction," *Machine Learning*, vol. 93, no. 2-3, pp. 357–380, Nov 2013.

[46] N. Nisan, T. Roughgarden, E. Tardos, and V. V. Vazirani, *Algorithmic game theory*. Cambridge University Press Cambridge, 2007.

[47] "Pilz Safe Camera System SafetyEye," http://www.eltron.pl/uploads/manufacturer_catalogs/16/10338/SafetyEYE_EN.pdf.

[48] J. S. Park, C. Park, and D. Manocha, "Intention-aware motion planning using learning based human motion prediction," in *R:SS*, 2017.

[49] "Search-Based Planning Library (SBPL)," http://www.sbpl.net/Software, [Online].