Collaborative Assembly in Hybrid Manufacturing Cells: An Integrated Framework for Human–Robot Interaction

Behzad Sadrfaridpour, *Student Member, IEEE*, and Yue Wang, *Senior Member, IEEE*

***Abstract*— Recent emergence of safe, lightweight, and flexible robots has opened a new realm for human–robot collaboration in manufacturing. Utilizing such robots with the new human–robot interaction (HRI) functionality to interact closely and effectively with a human co-worker, we propose a novel framework for integrating HRI factors (both physical and social interactions) into the robot motion controller for human–robot collaborative assembly tasks in a manufacturing hybrid cell. To meet human physical demands in such assembly tasks, an optimal control problem is formulated for physical HRI (pHRI)-based robot motion control to keep pace with human motion progress. We further augment social HRI (sHRI) into the framework by considering a computational model of the human worker's trust in his/her robot partner as well as robot facial expressions. The human worker's trust in robot is computed and used as a metric for path selection as well as a constraint in the optimal control problem. Robot facial expression is displayed for providing additional visual feedbacks to the human worker. We evaluate the proposed framework by designing a robotic experimental testbed and conducting a comprehensive study with a human-in-the-loop. Results of this paper show that compared to the manual adjustments of robot velocity, an autonomous controller based on pHRI, pHRI and sHRI with trust, or pHRI and sHRI with trust, and emotion result in 34%, 39%, and 44% decrease in human workload and 21%, 32%, and 60% increase in robot's usability, respectively. Compared to the manual framework, human trust in robot increases by 38% and 42%, respectively, in the latter two autonomous frameworks. Moreover, the overall efficiency in terms of assembly time remains the same.**

***Note to Practitioners*—Conventionally, industrial robots are used to perform repetitive tasks in human-free cages with minimal HRI for safety concerns. Thanks to the new safety and flexibility functions embedded in human-friendly manufacturing robots, humans and robots can now collaborate closely with each other accomplishing the tasks that were previously done by human workers solely. However, existing criteria for designing robot controllers need to be modified by considering the human workers' demands since the performance of a human worker would vary due to factors such as individual strength, working pattern, and interaction with the robot. To address this problem, we propose a novel framework that integrates HRI factors in controlling the motion of a robot for the collaborative assembly tasks. Within this framework, the speed of robot can be controlled such that its motion progress synchronizes with that of the human during the task to improve pHRI. Moreover, for better sHRI, human trust in robot is calculated and used to select robot path and modify its speed control. Furthermore, we dynamically change the robot facial expression to provide visual feedbacks for performance and safety concerns. The results of the experimental study show that integrating both pHRI and sHRI in the robot controller leads to a significant drop of human perceived workload and considerable increase of robot usability and human trust in robot while the overall efficiency in terms of assembly time remains intact. For practical utilization in assembly plants, sensory devices for tracking the human motion are required. The robot is also required to have built-in safety functions that reduce the impact of possible collisions between the human and the robot. We believe that this paper addresses how the implementation of human-friendly robots in the manufacturing environments can improve HRI and reduce workload.**

***Index Terms*— Assembly in manufacturing, collision avoidance, emotion, human arm movement, human–robot collaboration (HRC), robot motion planning and speed control, trust.**

I. Introduction

DUE to safety concerns, industrial heavy-duty robots are required to be operated in caged environments where only trained operators can interact with them through some external interfaces under running conditions [1]. Recently, the emergence of lightweight, flexible, and human-friendly robots with several built-in safety features promises sound and closes human–robot collaboration (HRC) in manufacturing environments. These developments improve human–robot interaction (HRI) to the extent that robots are perceived as social beings with which humans interact rather than simple tools [2]. An overview of the state of the art in physical HRI (pHRI) is provided in [3]. In collaborative pHRI, usually the labor of the task is divided based on the human and robot abilities such as human dexterity and robot ability in performing repetitive jobs. The existing design criteria of industrial robots focus on safety, cost, productivity, and reliability, among which safety has been the most critical and well-studied topic related to pHRI [4], [5]. The work in [6] summarizes criteria for designing intrinsically safer robots and strategies for ensuring safe execution of tasks in the presence

Manuscript received May 20, 2017; revised August 7, 2017; accepted August 28, 2017. Date of publication September 26, 2017; date of current version July 2, 2018. This paper was recommended for publication by Associate Editor J. Burgner-Kahrs and Editor Y. Sun upon evaluation of the reviewers' comments. This work was supported by the National Science Foundation under Grant CMMI-1454139. *(Corresponding author: Yue Wang.)*

The authors are with the Department of Mechanical Engineering, Clemson University, Clemson, SC 29634 USA (e-mail: bsadrfa@clemson.edu; yue6@clemson.edu).

This paper has supplementary downloadable multimedia material available at http://ieeexplore.ieee.org provided by the authors. The Supplementary Material contains material that is not included within the paper itself. This material is 58.4 MB in size.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TASE.2017.2748386

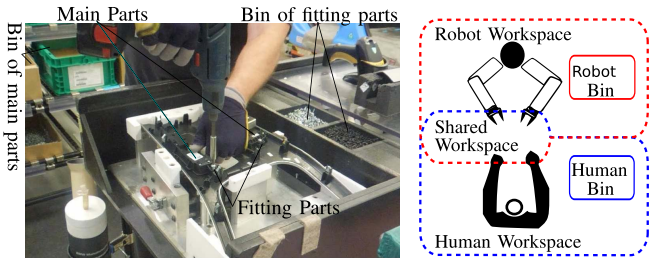


Fig. 1. Conventional manual versus a hybrid HRC manufacturing cell. (a) Conventional manufacturing cell (photo courtesy of BMW Spartanburg, SC, USA). (b) HRC hybrid manufacturing cell.

of humans. Collision avoidance is a fundamental requirement for safe physical interaction and has been studied extensively in robotics [4]. For humans and robots that share the same workspace, real-time collision avoidance can be achieved by detecting and understanding the environment including the human's body and its future motion, running a collision avoidance algorithm, and controlling the robot accordingly.

For a flexible and human-friendly robot that has new HRI functionalities and interacts closely with a human co-worker, considering safety and production efficiency objectives may not suffice [7]. Social HRI (sHRI) has been demonstrated in robots with affective abilities, such as Kismet, iCAT, Flobi, ERWIN, Kobian, NAO, Kamin, Ifbot, WE-3R III, Robokind, Geminoid [8] in education, social services, health care, and entertainment [2]. Nonetheless, sHRI should not be limited only to those areas. Some new collaborative robots designed for manufacturing such as Guizzo [9] have certain features that make the interaction more human-like and appealing. In particular, human trust in robot and robot anthropomorphic features may have high impacts on sHRI. Human-to-robot trust is one of the key factors in sHRI and is a prerequisite for effective HRC [10], [11]. Trust characterizes the reliance and tendency of human in using robots. Factors within a robotic system (e.g., performance, reliability, or attribute), the task, and the surrounding environment can all impact the trust dynamically [10]. Over-reliance or under-reliance might occur due to improper trust, which results in poor team collaboration, hence higher task load and lower overall task performance [10]. Some studies have explored how social behaviors of robots can impact human emotions. This topic is widely studied in the domain of human–computer interaction (HCI) as affective computing [12]. The utilization of embodied conversation agents, human-friendly robots, and facial expression are examples of social capabilities that can be included in robotic systems for a closer human-like interaction [13].

This paper investigates the impacts of augmenting the combined pHRI and sHRI factors into robot controller on the joint performance of a human–robot team performing an assembly task. The task chosen in this paper is usually accomplished by human workers and includes some repetitive physical movements for pick-and-place. Fig. 1(a) shows an example of a conventional manufacturing cell in which the human worker brings the required parts from the bin of main parts and puts them together, grabs some fitting parts such as screws or bolts from the bin of fitting parts, and finally assembles the product. A flexible robot can assist the human worker in doing such a task by bringing the required parts to the human worker [14]. The goal of the human–robot team is to assemble a product in a hybrid manufacturing cell. This cell is equipped with sensory devices for safety reasons and enables the human and the robot to perform tasks collaboratively while some parts of their workspaces are shared with each other [15]. Fig. 1(b) shows an example of a typical hybrid manufacturing cell, where the bins of different assembly parts are assigned to the robot and human worker, respectively. The human workspace includes the shared workspace, the designated bin for the human, and the area covered by paths between them. The robot workspace is defined similarly but contains the bin of the robot. The robot picks the main parts from the robot bin and places them in the shared workspace with the human worker. The human picks the fitting parts from the human bin to the shared workspace and assembles the final product. Our proposed framework allows the robot arm to select paths between the robot bin and the shared human–robot workspace based on trust evaluation and then move along the selected path while its translational velocity along the path is adjustable. More specifically, a set of preplanned paths is stored and the appropriate path is selected based on HRI criteria. This will be discussed in Section III-A. The pHRI-based control condition will involve prediction of human motion and synchronization of the robot motion progress with that of the human. This will be discussed in Section IV-A. Human trust in robot and robot emotional expressions will be considered as two main aspects of sHRI and devised in two integrated control conditions. These will be discussed in Sections IV-C and IV-D, respectively. In Section V, four control conditions are considered, i.e., the pure manual condition, the pHRI-based condition considering motion progress synchronization (Section IV-A), the integrated pHRI and sHRI condition considering motion progress synchronization and trust (Section IV-C), and the more comprehensive integrated condition considering motion synchronization, trust, and emotion displays (Section IV-D). In the manual condition, minimal interaction is considered in the design of the robot controller. The human worker manually adjusts the robot work pattern (i.e., path and speed).

In [16], we investigate how human trust in robot can be measured during the HRC manufacturing and construct a trust model for assembly tasks. In [8], we demonstrate how the integration of dynamic emotion in an HCI can benefit sHRI in assembly tasks in manufacturing. In [17], we explore the robot redundancy and alter its configuration in handover operations during assembly based on artificial robot-to-human trust. More recently, we propose in [14] a framework for augmenting both the pHRI and sHRI factors into the robot controller in which a trust model was considered for sHRI. This paper is extended based on the HRI framework in [14]. The main contributions of this paper beyond our previously works [8], [14], [16], [17] are summarized as follows.

1) Thorough statistical analysis for a set of robotic experiments with a human-in-the-loop is performed. The impacts of different control conditions on some of the well-known HRI criteria including human perceived

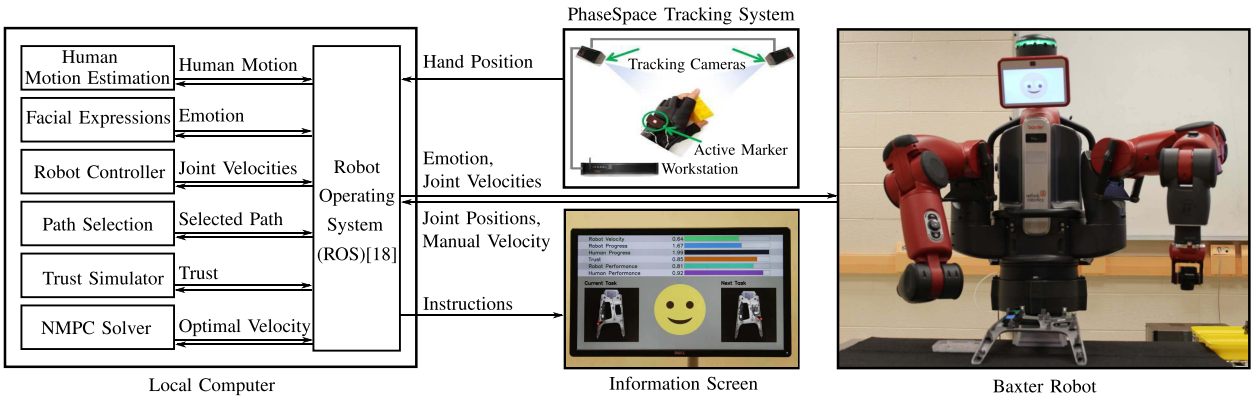


Fig. 2. Architecture of the integrated framework and illustration of the experiment setup.

workload, human trust in robot, robot usability, as well as objective measures in terms of robot average velocity and assembly time are evaluated and compared. More details are provided in Sections V and VI.

2) Both human-to-robot trust and dynamic robot emotions are integrated into the framework for a more comprehensive consideration of sHRI factors. More details are in Sections IV-C and IV-D.
3) A trust-based robot path selection strategy is devised such that if human trust in robot drops below some threshold value, the robot chooses a safer path with less chance of collision. More details are in Section III-A.
4) To increase the transparency of HRI, an HCI is designed to show the HRC system variables and robot emotion. More details are in Section IV-F.

Fig. 2 shows our proposed framework and the experimental setup. Section II summarizes the related works. Section III explains the detailed derivations of the robot motion controller (see the robot controller and path selection blocks in Fig. 2). The HRC system (the human motion estimation, trust simulator, and facial expressions blocks) and control laws [the nonlinear model predictive control (NMPC) solver block] are presented in Section IV. The experimental study (with PhaseSpace for tracking human motion and Baxter robot shown in Fig. 2) conducted for the evaluation of our proposed framework is presented in Section V. Thorough statistical analysis of the results of the experiments is presented in Section VI. This paper is concluded in Section VII.

II. Related Work

HRC in assembly lines can be viewed as a twofold problem: the task scheduling problem and the task execution problem. The solution to the scheduling problem identifies when different tasks should be assigned to the human and the robot, while the solution to the execution problem is robot controllers and motion planners. Both problems aim to improve safety, efficiency, cost, and productivity. In [19], a pHRI framework is proposed for task allocation and planning in HRC assembly. It consists of a two-layer planner for the high-level abstraction and an atomic level of allocation. The higher layer planner generates a coordinated skill sequence for the human–robot team. On the atomic layer, different hierarchical and concurrent state machines describe the skills of the robot. A summary of the EU project ROBO-PARTNER for the integration of automation and human capabilities in assembly operations is presented in [20], where efficient methods for task planning and execution are developed.

In [21], the human and robot agents coexist in the same cell and share tasks such as pick, place, release, and move. The task assignment is sequential and based on capability, availability, and operation time needed by either agent to perform a job. In the framework, the human and robot do not work simultaneously, and their interaction is restricted by the safety consideration. A contact-based pHRI framework for assembly tasks is proposed in [22]. The task is to assemble a car joint that includes insertion of six balls in a joint's case. The robot behaves actively to reduce the load on the human and passively to comply to his/her demands. Through both risk analysis and experiment validations, it is shown that the framework is compatible with safety standards and reduces human workload. Another framework for pHRI is proposed in [23], where the robot controller adapts behavior according to the human fatigue level during the task. The task is comanipulation, and the initial interaction is a leader–follower relationship, in which human is the leader and robot learns skills by feedbacks from the human. When the human fatigue reaches a predetermined level, the interaction alters from collaboration to supervision. The robot takes over the task to reduce the human load and the human controls the high-level interaction behavior. A pHRI framework for hybrid manufacturing cell of cable harness assembly is proposed in [24] for ensuring safety in task execution. Task planning is performed based on a hierarchical task decomposition approach adopted from ergonomics. Different hardware and control strategies such as designated and safe workspaces for the human and robot, safe design of the robot, and human monitoring are proposed for safety implementation. The impacts of robot motion speed and distance from the human are evaluated on system performance and human mental workload. Another HRC assembly cell for task execution is presented in [25]. The system has three key elements: a mobile robot with two manipulators for feeding the parts, production process information interface for the human, and safety management for HRC.

In sum, planning and execution are the two major problems in HRC assembly in manufacturing. Some of the related works address both problems, while others only consider

one aspect. In this paper, we also assume that the solution to the planning problem is given and focus on the execution problem. Moreover, most existing works consider pHRI and more specifically safety. Their ultimate goal is to find safe and efficient control policies for robots to accomplish the required task (assigned by the planner) in the presence of a human worker without collision. The main contribution of our work is to integrate both pHRI and sHRI into the robot path planning and speed control for safety, efficiency, as well as balanced human experience.

III. Robot Motion Controllers

In this section, the design of robot motion controller including path planning and the calculation of the robot joint velocity based on the optimal velocity of the robot end-effector along the planned path will be introduced in sequence.

A. Robot Path Planning

Robot motion planning is an active research topic, and a considerable amount of literature is dedicated to this field. Motion planning includes: 1) path planning for searching a (possibly optimized) collision-free path in the configuration space (i.e., the set of all robot configurations) regardless of the dynamics of the robot and 2) trajectory planning that considers the time evolution of robot dynamics (sometimes along a planned path) for satisfying certain optimization requirements as well as differential constraints. Sampling-based planners are widely employed to construct a data structure (roadmap or tree) for representing collision-free paths [26]. Other approaches for path planning include potential-field-based techniques and combinatorial methods that also make roadmaps, such as cell decompositions [26]. Trajectory planning addresses optimization criteria such as time, energy, force, effort, or the jerk minimization and differential constraints, i.e., limits on velocities, and possibly accelerations due to kinematic and dynamic considerations of the robot. Trajectory planning can be addressed either by direct methods that implement sampling-based algorithms through considering differential constraints or by decoupled approaches that first plan a path and then compute a timing function along the path [26]. In this paper, our focus is on constructing a motion trajectory to improve HRI rather than other well-known optimization criteria. Hence, the decoupled approach is adopted, and we formulate the problem as obtaining the motion velocity along a predefined path based on the consideration of HRI factors. The Baxter robot we use has some safety constraints that prevent solving the problem in the acceleration level. Moreover, identification of robot dynamics can be a challenging task [27]. To make the framework independent of knowledge of the robot dynamics model, this problem is solved in the velocity level.

Path planning can be done either online as the robot is executing the task or offline before the task starts [26]. Online path planning is more desirable if there are uncertainties; however; it requires more computational resources. In the manufacturing setting, since the initial and final positions of the robot end-effector, as well as the shared human–robot workspace, are given (see Fig. 3), the uncertainty is negligible. Thus, the problem is simplified by considering offline obstacle avoidance and polynomial curves defined in the task-space. However, the proposed framework can be extended to consider more advanced motion planning techniques in the future work.

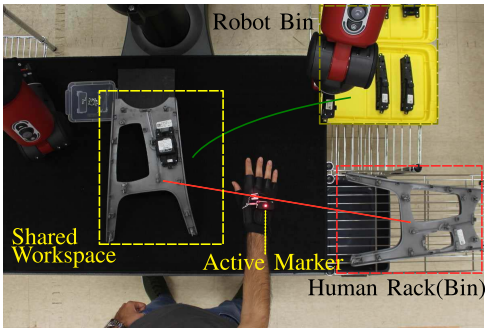


Fig. 3. Robot (green) and human (red) sample paths.

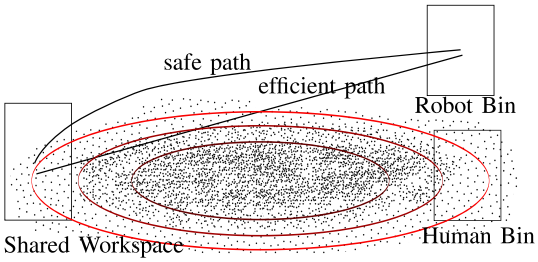


Fig. 4. Illustration of sample data of human movements in the workspace: an efficient robot path versus a safe robot path.

For accomplishing the pick-and-place task, the robot end-effector is desired to pass through a set of points of interest (POIs) defined in the task-space. This set includes the fixed initial and final POIs, and the intermediate POIs that are chosen via the high-level path planing. Thus, given a set of n_p data points $\{\boldsymbol{d_i} = [x_i,\ y_i,\ z_i]^T \in \mathbb{R}^3,\ 1 \leq i \leq n_p\}$, we choose a path in the three-dimensional Euclidean space, $\mathbb{R}^3$, that goes through these points, i.e., $\boldsymbol{p} : s \in [0, l] \rightarrow \mathbb{R}^3$, where the path parameter s is the arc length of the distance traveled along the path. Each point on this path is given by $\boldsymbol{p}(s) = [x(s),\ y(s),\ z(s)]^T,\ s \in [0, l]$. We use a simple cubic polynomial to construct $\boldsymbol{p}(s)$. Moreover, based on the HRI criteria, a high-level path planning approach can be adopted for choosing a candidate path from a set of predefined paths. Fig. 4 shows two of the candidate paths. The dots in the figure represent some sample data of a human worker's hand position while performing a task. By fitting appropriate probability distributions to these data points, regions with different safety levels can be identified. The red contours represent three examples of data points falling within the same density distributions. We utilize the trust of the human to robot as an HRI factor in the path selection strategy. A similar approach is used in [17] by developing an artificial robot-to-human trust model and implementing trust-based arm configuration and motion planning of a collaborative robot in handover tasks. Experiment results with a human-in-the-loop in [17] show that trust-based handover strategy statistically

outperforms nontrust-based strategy in both pHRI and sHRI criteria. In this paper, we adopt a similar concept for selecting the robot path with differences in the detailed trust model and path planning strategy. More specifically, in [17], the robot transitional path in the task-space is fixed, and it performs the motion as planned unless the trust of robot to human drops below some threshold value. In that case, the robot alters its end-effector orientation to minimize the impact force between the human and the robot. In this paper, the robot end-effector orientation is fixed, but we change the robot transitional path in the task-space based on the dynamic evaluation of human-to-robot trust. In the following, we justify the choice between a conservative (safe) and an aggressive (efficient) robot path based on the analysis of human-to-robot trust. Both human and robot can cause a collision. Most works in the literature deal with collision avoidance from the robot perspective. For example, in [28], the collision probability is defined as a function of the measurement error and relative velocity between the human and the robot. As the uncertainty in robot measurement data increases, the likelihood of the collision increases. Thus, low performance of the robot increases the probability of the collision. On the other hand, with the increase of physical workload and hence performance decline, the situational awareness of human worker decreases [29] and the probability of interfering and colliding with the robot increases. Here, we consider both cases and assume that either low performance of the robot or the human increases the likelihood of the collision. According to performance-centered metrics [11], low performance of robot leads to low human trust in robot. Note that due to the close interaction of the human and the robot, the changes in the performance of the robot impact the human's performance as well. In Section IV-C, we will define a computational model of human's trust in the robot (20) that will be considered in our HRI-based motion planning. According to this model, human trust in robot depends on prior trust, robot performance, and human performance. That is, poor performances of the human and robot result in a low trust value. In turn, the computational trust level can be used to indicate both the human and robot performances and, hence, is a criterion for choosing between safe and efficient paths. If only considering efficiency, a path with shorter length would be chosen. However, if the trust level is low, choosing a short path might result in higher probability of collision. Hence, for safety consideration, a more conservative but longer path should be chosen. If trust is high, selecting a short path might still be safe. Therefore, there are tradeoffs between efficiency and safety for choosing the path. Since different levels of human trust in the robot reflect the variation of both the human and robot performances, we can adopt a trust-based path planning method. We use a lookup table to specify a corresponding preplanned path for every range of trust values. In general, low values of trust are associated with low performances of human or the robot, which suggests a more conservative path, while a more efficient path can be chosen for high values of trust.

Remark 1: The computational model of trust in this paper reflects the dynamic and temporary trust of the human worker in the robot during the interaction in terms of flexibility and efficiency as well as human's own performance. This is more specific than the general notation of trust in robot that determines the human's acceptance and, hence, utilization of the robot (automation) [10], [11]. •

The human arm movement data were collected in the pilot study and used for the construction of paths with lower probability of collision of the human and the robot.

TABLE I
DESCRIPTION OF THE PARAMETERS IN MOTION CONTROLLER

Parameter	Description
$\boldsymbol{\phi}$	End-effector orientation in task-space
$\boldsymbol{p}$	End-effector position in task-space
s	Arc length of the distance traveled along a path
$\dot{s}$	Derivative of arc length (s) with respect to time
$\boldsymbol{x}$	End-effector pose including position and orientation
$\boldsymbol{q}$	Joint positions of the robotic arm
$\boldsymbol{f_t}$	Direct kinematics function
$\boldsymbol{J}$	Task Jacobian matrix
$\dot{\boldsymbol{q}}_0$	An arbitrary joint-space initial velocity
$\boldsymbol{t_p}(s)$	Tangent to the path $\boldsymbol{p}(s)$ at s
$\boldsymbol{t}(s)$	Change of end-effector desired pose $\boldsymbol{x}$ with respect to s
$\boldsymbol{\tau_p}(s)$	Vector from the actual pose of the end-effector at s to its desired pose along the path
$\boldsymbol{\tau}(s)$	Change of end-effector actual pose $\boldsymbol{x}$ with respect to s
v	Translational velocity along the path

B. Robot Joint Velocity Control

The problem of controlling a robot manipulator along a predefined path is widely studied in the literature. It mainly focuses on determining the robot joint velocities for moving along a path in a task-space such that it satisfies certain constraints. Most efforts made in this area solve this problem in the acceleration level [30]. However, due to the safety constraints of Baxter and the lack of knowledge of robot dynamics, we solve this problem in the velocity level. The details of the robot joint controller for the proposed HRI framework are discussed in [14]. In the following, we briefly restate the solution, describe an adjustment we made for increasing the control accuracy, and finally present Algorithm 1 for the robot controller. Table I summarizes the parameters used in this section. It is desired to manipulate the robot end-effector along a given path such that its speed along the path can be dynamically adjusted. Let the configuration of the end-effector of the robot arm in the task-space be denoted by the reference point position, $\boldsymbol{p} \in \mathbb{R}^3$, and the orientation vector, $\boldsymbol{\phi} \in \mathbb{R}^3$ (such as Euler angles or the roll-pitch-yaw representation). More specifically, we define $\boldsymbol{x} = [\boldsymbol{p}, \boldsymbol{\phi}]^T \in \mathbb{R}^6$ as the vector of the pose (position and orientation) of the end-effector, where ^{T} denotes transpose. We define $\boldsymbol{q} \in \mathbb{R}^n$ as a vector of the joint positions of the robotic arm in the joint-space. The relation between the task-space and the joint-space is expressed by direct kinematics equation

$$\boldsymbol{x} = \boldsymbol{f_t}(\boldsymbol{q}), \quad m < n \tag{1}$$

where m and n are degrees of freedom (DoFs) in the task-space and joint-space, respectively. In our problem,

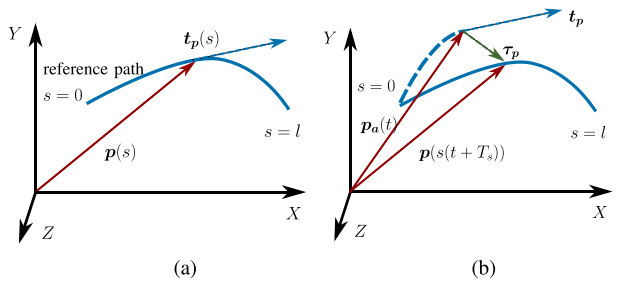


Fig. 5. Reference versus the compensated trajectories. $\boldsymbol{p_a}(t)$ is the actual position of the end-effector at current time and $\boldsymbol{p}(s(t+T_s))$ is the reference position at time $t+T_s$. (a) Reference trajectory. (b) Compenstated trajectory.

$m=6$ and $n=7$. The function $\boldsymbol{f_t}$ represents direct (forward) kinematics. The first-order differential kinematics are

$$\dot{\boldsymbol{x}} = \boldsymbol{J}(\boldsymbol{q})\dot{\boldsymbol{q}} \tag{2}$$

where $\boldsymbol{J}(\boldsymbol{q}) = \partial \boldsymbol{f_t}/\partial \boldsymbol{q}$ is the $m \times n$ task Jacobian matrix. The general solution to (2) is

$$\dot{\boldsymbol{q}} = \boldsymbol{J}^{\dagger}(\boldsymbol{q})\dot{\boldsymbol{x}} + (\boldsymbol{I} - \boldsymbol{J}^{\dagger}(\boldsymbol{q})\boldsymbol{J}(\boldsymbol{q}))\dot{\boldsymbol{q}}_{\boldsymbol{0}} \tag{3}$$

where $\boldsymbol{J}^{\dagger}(\boldsymbol{q})$ is the Moore–Penrose pseudoinverse of $\boldsymbol{J}(\boldsymbol{q})$, $\dot{\boldsymbol{q}}_{\boldsymbol{0}}$ is an arbitrary joint-space initial velocity, and $\boldsymbol{I}$ is an identity matrix.

Remark 2: We assume that in addition to the Cartesian position coordinates and Euler angles representations introduced above, the desired positions and orientations of the end-effector can also be given for all admissible values on the curve parameterized by $s \in [0, l]$, i.e., $\boldsymbol{p}(s)$ and $\boldsymbol{\phi}(s)$. As the robot end-effector follows the given path, since $s(t)$ is a function of time, $\boldsymbol{p}$ and $\boldsymbol{\phi}$ become implicit functions of time and the pose of the robot at the time t would be $\boldsymbol{x}(s(t))$. However, note that $\boldsymbol{x}(s(t)) \neq \boldsymbol{x}(t)$. If the end-effector deviates from the reference path, the actual pose is not on the reference path and it cannot be presented using s anymore. We first solve the problem for the ideal case where the robot exactly follows the given path and then provide a solution for the deviated case.

Now, we show the relation between the end-effector velocity in the task-space $\dot{\boldsymbol{x}}$ and the robot velocity along the path $\dot{s}$. Let us denote the following operators for derivatives with respect to s and t, respectively, as $(\,)' = d/ds$ and $\dot{(\,)} = d/dt$. A tangent to the curve can be calculated as the vector $\boldsymbol{t_p}(s) = \boldsymbol{p}'(s)$. Define the unit vector of this tangent as $\hat{\boldsymbol{t}}_{\boldsymbol{p}}(s) = (1/||\boldsymbol{t_p}(s)||)\boldsymbol{t_p}(s)$, where $||\boldsymbol{t_p}(s)||$ is the 2-norm of $\boldsymbol{t_p}(s)$. Fig. 5(a) shows a demonstration of these definitions. If the robot arm follows $\boldsymbol{p}(s)$, its translational velocity can be written as $\dot{\boldsymbol{p}}(s(t)) = \dot{s}(t)\boldsymbol{p}'(s) = \dot{s}(t)\ \boldsymbol{t_p}(s)$. Since $\dot{s}$ is a scalar variable, this can be rewritten as $\dot{\boldsymbol{p}} = \upsilon\ \hat{\boldsymbol{t}}_{\boldsymbol{p}}(s)$, where $\upsilon = \dot{s}||\boldsymbol{t_p}(s)||$ is a scalar number representing the velocity at which robot is moving on $\boldsymbol{p}(s)$. We call υ as the path velocity along $\boldsymbol{p}(s)$. Similar to $\boldsymbol{p}(s)$, we can define the minimal description of the end-effector orientation as $\boldsymbol{\phi}(s) = [\alpha(s),\ \beta(s),\ \gamma(s)]^T$. Since in pick-and-place applications, the robot orientation along the path can remain unchanged, and the pose direction vector $\boldsymbol{t}(s)$ can be written as

$$\boldsymbol{t}(s) = [\hat{\boldsymbol{t}}_{\boldsymbol{p}}(s),\quad \boldsymbol{0}]^T. \tag{4}$$

Note that the vector $\boldsymbol{t}(s)$ represents the change of end-effector pose with respect to s. Therefore, due to the fixed-orientation of the end-effector, the change of orientation vector is set to $\boldsymbol{0}$, i.e., the given orientations for all values of s are equal. Since $\boldsymbol{x} = [\boldsymbol{p},\ \boldsymbol{\phi}]^T$, the robot end-effector velocity in task-space can be written as $\dot{\boldsymbol{x}}(s) = \upsilon \boldsymbol{t}(s)$. Together with (3), the joint velocities for the robot end-effector to move along $\boldsymbol{p}(s)$ can be computed as

$$\dot{\boldsymbol{q}} = \upsilon \boldsymbol{J}^{\dagger}(\boldsymbol{q})\boldsymbol{t}(s(t)) + (\boldsymbol{I} - \boldsymbol{J}^{\dagger}(\boldsymbol{q})\boldsymbol{J}(\boldsymbol{q}))\dot{\boldsymbol{q}}_{\boldsymbol{0}}. \tag{5}$$

To account for the deviations of the robot end-effector from the reference path, we define $\boldsymbol{x_a}(t) = [\boldsymbol{p_a}(t),\ \boldsymbol{\phi_a}(t)]^T$ as the actual pose vector of the end-effector. Let $\boldsymbol{\tau_p}(t)$ be a vector connecting the actual position of the end-effector $\boldsymbol{p_a}(t)$ at time t to the next point on the reference path $\boldsymbol{p}(s(t+T_s))$, where T_s is the controller sampling time [see Fig. 5(b)]. Define the drifted trajectory direction as

$$\boldsymbol{\tau}(t) = [\hat{\boldsymbol{\tau}}_{\boldsymbol{p}}(t),\quad \boldsymbol{0}]^T \tag{6}$$

where $\hat{\boldsymbol{\tau}}_{\boldsymbol{p}}(t)$ is the unit vector of $\boldsymbol{\tau_p}(t)$. Now, the compensated task velocity can be defined as $\dot{\boldsymbol{x}}_{\boldsymbol{c}}(t) = \upsilon\ (\rho_1 \boldsymbol{t}(s(t))) + \rho_2 \boldsymbol{\tau}(t))$, where ρ_1 and ρ_2 are the weighting factors with $\rho_1 + \rho_2 = 1$, $\rho_i \in [0, 1]$, $i = 1, 2$. Here, choices of the values of ρ_1 and ρ_2 are tradeoffs between the accuracy and speed of robot end-effector, respectively. For the closest distance between the robot end-effector and reference trajectory, we can set $\rho_1 = 0$ and $\rho_2 = 1$. For the fastest forwarding speed in the direction of the reference, we can set $\rho_1 = 1$ and $\rho_2 = 0$. In this paper, since both accuracy in reaching the goal and forwarding speed are important, we choose $\rho_1 = \rho_2 = 0.5$. By replacing $\dot{\boldsymbol{x}}$ with $\dot{\boldsymbol{x}}_{\boldsymbol{c}}$ in (5), the compensated equation for joint velocity vector can be computed as $\dot{\boldsymbol{q}} = \upsilon \boldsymbol{J}^{\dagger}(\boldsymbol{q})(\rho_1 \boldsymbol{t}(s(t)) + \rho_2 \boldsymbol{\tau}(t)) + (\boldsymbol{I} - \boldsymbol{J}^{\dagger}(\boldsymbol{q})\boldsymbol{J}(\boldsymbol{q}))\dot{\boldsymbol{q}}_{\boldsymbol{0}}$. Furthermore, to avoid singularities, we use the damped least-squares technique [31] by letting

$$\boldsymbol{J}^T(\boldsymbol{q})\dot{\boldsymbol{x}} = (\boldsymbol{J}^T(\boldsymbol{q})\boldsymbol{J}(\boldsymbol{q}) + \lambda^2 \boldsymbol{I})\dot{\boldsymbol{q}} \tag{7}$$

instead of (2), where $\lambda \in \mathbb{R}$ is the damping factor. Assuming zero initial joint velocities, i.e., $\dot{\boldsymbol{q}}_{\boldsymbol{0}} = \boldsymbol{0}$, the solution of (7) is

$$\dot{\boldsymbol{q}} = \boldsymbol{J}^*(\boldsymbol{q})\dot{\boldsymbol{x}} \tag{8}$$

$$\boldsymbol{J}^*(\boldsymbol{q}) = (\boldsymbol{J}(\boldsymbol{q})\boldsymbol{J}^T(\boldsymbol{q}) + \lambda^2 \boldsymbol{I})^{-1}\boldsymbol{J}^T(\boldsymbol{q}). \tag{9}$$

The details of this approach are described in [31]. The solution is a tradeoff between accuracy and feasibility for choosing the joint-space velocity needed to achieve $\dot{\boldsymbol{x}}$. Considering the compensated task velocity $\dot{\boldsymbol{x}}_{\boldsymbol{c}}$, (8) can be written as

$$\dot{\boldsymbol{q}} = \upsilon \boldsymbol{J}^*(\boldsymbol{q})(\rho_1 \boldsymbol{t}(s(t)) + \rho_2 \boldsymbol{\tau}(t)). \tag{10}$$

Equation (10) is the equivalent of (11) in [14]. To further increase the accuracy of the robot end-effector motion along the path, we implement the closed-loop inverse kinematics approach [32]. Thus, (10) can be written as

$$\dot{\boldsymbol{q}} = \boldsymbol{J}^*(\boldsymbol{q})\{\upsilon[\rho_1 \boldsymbol{t}(s(t)) + \rho_2 \boldsymbol{\tau}(t)] + \boldsymbol{K}[\boldsymbol{x}(t) - \boldsymbol{f_t}(\boldsymbol{q})]\} \tag{11}$$

where $\boldsymbol{K}$ is a constant positive-definite gain matrix. In our experiment (Section V), we choose $\boldsymbol{K} = 0.01\boldsymbol{I}_{n\times n}$. Algorithm 1 provides a brief summary of the motion control procedure of the robot.

Algorithm 1: Moving Robot Along a Given Path

procedure MOVE_ROBOT($\boldsymbol{p}$) ▷ Moves along $\boldsymbol{p}$
  $s \leftarrow 0$
  Subscribe(υ) ▷ Update path velocity from solver
  Read($\boldsymbol{q}$) ▷ Get joint positions
  while $s \leq l$ **do** ▷ Check if the robot reaches the end
    Calculate($\boldsymbol{t}(s)$) ▷ Eqn. (4)
    Calculate($\boldsymbol{\tau}(s)$) ▷ Eqn. (6)
    Calculate($\boldsymbol{J}^*$) ▷ Eqn. (9)
    Calculate($\boldsymbol{f}_t(\boldsymbol{q})$)
    Calculate($\dot{\boldsymbol{q}}$) ▷ Eqn. (11)
    Publish($\dot{\boldsymbol{q}}$) ▷ Send joint velocity commands
    Read($\boldsymbol{q}$) ▷ Get joint positions
    Calculate($\boldsymbol{x}(t)$) ▷ Eqn. (1)
    Calculate(s)
  end while
end procedure

C. Robot Path Velocity Controller

So far, we discuss the control of the robot end-effector along a reference path with a given translational velocity. Next, the controller for adjusting the transitional velocity along the path will be discussed. In HRC assembly scenarios, this velocity can be either fixed or varying depending on criteria such as HRI and productivity. On one hand, faster movement of the robot results in higher efficiency. On the other hand, the robot should keep pace with human for better HRI. The robot path velocity can be set manually by the human worker based on his/her preference or automatically based on objective performance measures. The manual adjustment of the robot speed can be realized through some human–machine interface. Baxter has a wheel button on both of its wrists and can be used for manual adjustment of the robot velocity. For automatic speed control, we model HRC systems based on pHRI and the integration of pHRI and sHRI, respectively. For each HRC system, we utilize the NMPC approach to solve for the optimal path velocity υ. We incorporate the NMPC toolbox [33] into robotic operating system (ROS) for this purpose. The details of the HRC systems and the optimal control are described in the next section.

IV. Human–Robot Collaboration System

In this section, we develop robot velocity control along the path based on the pHRI and sHRI factors for the HRC system. For the pHRI system, it is desired to control robot motion so that its motion progress can follow that of the human. For the sHRI system, the robot motion may be altered such that human trust in robot during the interaction is always higher than a threshold for effective HRC. Furthermore, robot emotion displays will be augmented in the framework for providing visual feedbacks regarding safety and performance.

A. Physical Human–Robot Interaction (pHRI) System

For assembling a product, the robot and human are required to bring r and h parts, respectively, to the shared workspace and human assembles them together. In the real factory environment, this process is continuously repeated in the assembly line. Here, in the laboratory setting, we assume that the total

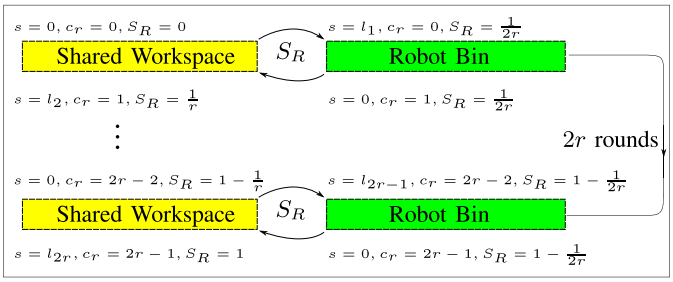


Fig. 6. Demonstration of robot progress S_R for assembling the first product. This process is repeated N_p times.

number of finished products to be assembled is N_p (each with r and h parts from the robot and human, respectively). For each part, the robot moves along a path between the bin of parts and the shared workspace back and forth (Fig. 3). The sequence of the reference paths, $\{\boldsymbol{p}_i\}$, $i = 1, 2, \ldots,$ is determined by the high-level trust-based path planning discussed in Section III-A. We denote the robot's path progress, $S_R \in \mathbb{R}^+$, as follows:

$$S_R = \frac{s}{2rl_i} + \frac{c_r}{2r} \tag{12}$$

where s is the arc length of distance traveled along the chosen reference path $\boldsymbol{p}_i$ with length l_i and c_r is the number of times that the robot completely traveled a path from the shared workspace and the robot bin or vice versa (see the illustration in Fig. 6). The term $(s/2l_i)$ gives the ratio of the distance traveled by the robot end-effector along a reference path with respect to the total round trip length of the chosen reference path for picking one part. The robot starts the task of moving along a path with length l_1 from the shared workspace, where initially both c_r and S_R are 0. Just before the robot reaches the bin of the required parts for the first time, we have $s = l_1$, $c_r = 0$, and $S_R = (1/2r) + (0/2r) = (1/2r)$. After the robot reaches the bin of required parts, it picks up a part and moves back toward the shared workspace on a path with length l_2. Note that at the start of this motion, the value of S_R is still the same since $s = 0$ and $c_r = 1$. When the end-effector reaches the shared workspace, $s = l_2$, $c_r = 1$, and $S_R = (1/2r) + (1/2r) = (1/r)$. The value of S_R increases by $(1/r)$ each time the robot returns to the shared workspace and increases by one unit every time the robot finishes bringing all r parts to the shared workspace.

In Section III-B, from $\upsilon = \dot{s}||\boldsymbol{t}_{\boldsymbol{p}}(s)||$, it follows that $\dot{s}(t) = (\upsilon(t)/\|\boldsymbol{t}_{\boldsymbol{p}}(s(t))\|)$, $0 \leq \upsilon(t) \leq \overline{\upsilon_R}$, where $\upsilon(t)$ and $\overline{\upsilon_R}$ are the control input and its maximal value, respectively. Without loss of generality, we set $\overline{\upsilon_R} = 1$ representing the highest speed achievable by the robot. The kinematics of robot path progress, S_R, is then given by

$$\dot{S}_R(t) = \frac{\dot{s}(t)}{2rl_i} = \frac{\upsilon(t)}{2rl_i\|\boldsymbol{t}_{\boldsymbol{p}}(s(t))\|}, \quad 0 \leq \upsilon(t) \leq \overline{\upsilon_R}.$$

Since digital controllers and sensors work in the discrete-time settings, we next consider and implement our system kinematics in discrete time

$$S_R(k+1) = \frac{\upsilon(k)T_s}{2rl_i\|\boldsymbol{t}_{\boldsymbol{p}}(s(k))\|} + S_R(k), \quad 0 \leq \upsilon(k) \leq \overline{\upsilon_R} \tag{13}$$

where T_s and k are the sampling time and time step, respectively. Here, $S_R((k+1)T_s)$ and $S_R(kT_s)$ are written as $S_R(k+1)$ and $S_R(k)$, respectively, for the sake of simplicity.

Similarly, the path progress made by human, denoted as S_H, can be defined according to (13) but with a slight modification. Human hand motion has uncertainties and does not follow a specific path in general. However, the start and end points of the motion are fixed and located at the shared workspace and the human bin. We denote the length between these fixed points as l_h and consider the line that connects these two points as the human reference line. The human hand position can be measured by the PhaseSpace motion capture system as shown in Fig. 2. Let us denote the position of the human hand as $\boldsymbol{p_h}(k) = [x_h(k),\ y_h(k),\ z_h(k)]^T$, the position of the start point of the reference line as $\boldsymbol{p_0}(k) = [x_o(k),\ y_o(k),\ z_o(k)]^T$, and the position of the endpoint as $\boldsymbol{p_f}(k) = [x_f(k),\ y_f(k),\ z_f(k)]^T$. Denote the vector that connects the start and end points of the reference line and its unit vector as $\boldsymbol{i_h}(k) = \boldsymbol{p_f}(k) - \boldsymbol{p_0}(k)$ and $\hat{\boldsymbol{i}}_{\boldsymbol{h}}(k)$, respectively. The position of the human hand along the reference line at time step k can then be computed as $\hat{\boldsymbol{i}}_{\boldsymbol{h}}(k)(\boldsymbol{p_h}(k) - \boldsymbol{p_0}(k))$. Human path progress, $S_H \in \mathbb{R}^+$, is defined as

$$S_H(k+1) = \frac{\hat{\boldsymbol{i}}_{\boldsymbol{h}}(k)(\boldsymbol{p_h}(k) - \boldsymbol{p_0}(k))}{2hl_h} + \frac{c_h}{2h} \tag{14}$$

where c_h is the number of previously traveled paths by human between the shared workspace and the human bin. The first term in (14) is the ratio that the human hand has moved along the current reference line so far with respect to the total length that his/her hand is required to travel for bringing h parts to the shared workspace. Similar to c_r, c_h increases by 1 after each time the human reaches the shared workspace or the human bin. The details for estimating $\boldsymbol{p_h}$ are explained in the next section.

In the pHRI system, it is desired that the robot path progress follows the human path progress efficiently. This can be formulated as the following NMPC problem:

$$\min_{v(0),\ldots,v(N-1)} \sum_{k=1}^{N} \{||S_R(k) - S_H(k)||_Q + ||v(k) - \overline{v_R}||_R\} \tag{15}$$

subject to (13) and (14), where N is the prediction horizon and $||\cdot||_Q$ ($||\cdot||_R$) represents the weighted norm with respect to the positive number Q (R). The first term in (15) addresses the human–robot synchronization of motion progress, i.e., it is desired that $S_R = S_H$ during the pick-and-place operations so that the robot brings the required assembly parts to the human in time. The second term seeks to maximize robot efficiency. Minimizing the cost has the effect of pushing S_R to S_H and v to $\overline{v_R}$. Since $v(k) \geq 0$, S_R is nondecreasing according to (13). If the value of Q is considerably higher than R, then the robot velocity will decrease and eventually stop when the robot progress gets ahead of human progress, i.e., $S_R(k) > S_H(k)$. These stops make the robot motion nonsmooth and annoying for the participant, and thus will impact the experiments in a negative way. If the value of Q is considerably lower than R, then the robot velocity is always close to $\overline{v_R}$. This also impacts the experiments in a negative way since the robot does not adapt to the human. Hence, equal weighting of Q and R should be chosen to ensure that the robot adaptability with human motion progress and efficiency imposes equal effects on the cost evaluation.

Remark 3: Here, the assembly of car center console is considered as a task that human and robot accomplish together. Nonetheless, this framework can be applied to other human–robot collaborative tasks such as hose assembly and car door assembly where the progress synchronization for pick-and-place is important. •

B. Human Kinematics Learning

We now provide the details of the estimation of human hand kinematics. It should be noted that the human kinematics are required in order to predict the motion of human worker over a specified time horizon that will be used to design robot motion controller accordingly. Human hand position at the next time step can be predicted using its current velocity and position. Thus, to estimate the future values of human hand position over the horizon N in (15) and (23), the value of $\boldsymbol{v_h}$ is required. In this paper, a recursive least-square (RLS)-based black-box approach is exerted in order to obviate the problem of considering the human kinematics directly and through a model-based perspective. As reported in [34], there exist many advantages for this learning algorithm including low computational burden, fast convergence to the solution, and unbiasedness when it is subject to the white noise. Our goal is to estimate the value of the human worker's hand velocity, $\boldsymbol{v_h}$ based on

$$\boldsymbol{v_h}(k+1) = \boldsymbol{\theta}^T \boldsymbol{\Phi}(k) \tag{16}$$

where $\boldsymbol{\theta}$ is the vector with real coefficients and $\boldsymbol{\Phi}(k)$ is the matrix with the input and past output values of the system [34]. At each time step, we consider $\boldsymbol{\Phi}(k)$ as

$$\boldsymbol{\Phi}(k) = \begin{bmatrix} [-\boldsymbol{v_h}(k) & \ldots & -\boldsymbol{v_h}(k+1-i)]^T \\ [\boldsymbol{p_h}(k) & \ldots & \boldsymbol{p_h}(k-j)]^T \end{bmatrix} \tag{17}$$

where $i \geq 1$ and $j \geq 1$ are arbitrary numbers, k indicates the present time step, and $k-o$ represents o time steps before the present. The input and output data are the position and velocity of the human hand, respectively. Obviously, the elements of $\boldsymbol{\Phi}$ consist of the previous outputs and inputs data as well as the present input value. The estimated value of human hand velocity, $\hat{\boldsymbol{v}}_{\boldsymbol{h}}(k+1)$ can be found by $\hat{\boldsymbol{v}}_{\boldsymbol{h}}(k+1) = \hat{\boldsymbol{\theta}}^T(k)\boldsymbol{\Phi}(k)$, where $\hat{\boldsymbol{\theta}}(k)$ is the estimated value of the filter at time step k. Based on the least square method, the following cost function should be minimized for estimating $\boldsymbol{v_h}(k+1)$: $J(k) = \sum_{i=1}^{k} [\boldsymbol{v_h}(i) - \hat{\boldsymbol{\theta}}^T(k)\boldsymbol{\Phi}(i-1)]^2$. Instead of solving this equation, to reduce the computational burden, the RLS method can be used to find the filter recursively through the following equations [34]:

$$\begin{aligned} \hat{\boldsymbol{\theta}}(k+1) &= \hat{\boldsymbol{\theta}}(k) + K(k)(\boldsymbol{v_h}(k+1) - \hat{\boldsymbol{\theta}}^T(k)\boldsymbol{\Phi}(k)) \\ K(k) &= \boldsymbol{F}(k)\boldsymbol{\Phi}(k)[1 + \boldsymbol{\Phi}^T(k)\boldsymbol{F}(k)\boldsymbol{\Phi}(k)]^{-1} \\ \boldsymbol{F}(k+1) &= (\boldsymbol{I} - K(k)\boldsymbol{\Phi}^T(k))\boldsymbol{F}(k) \end{aligned} \tag{18}$$

where $\boldsymbol{I}$ is the identity matrix. The above process is repeated until the termination condition $\|\hat{\boldsymbol{\theta}}(k) - \hat{\boldsymbol{\theta}}(k-1)\| \leq \epsilon$ is

satisfied, where ϵ is a sufficiently small positive value. For estimating the value of $\boldsymbol{p_h}$ at the future time steps which is needed for solving (15) over the horizon N, the following equations are used to recursively update the output values over the look-ahead horizon for time step $k+m,\ m=1,2,\ldots,N$:

$$\begin{aligned}\hat{\boldsymbol{p}}_{\boldsymbol{h}}(k+m) &= \hat{\boldsymbol{v}}_{\boldsymbol{h}}(k+m-1)T_s + \hat{\boldsymbol{p}}_{\boldsymbol{h}}(k+m-1)\\ \hat{\boldsymbol{v}}_{\boldsymbol{h}}(k+m) &= \hat{\boldsymbol{\theta}}^T(k)\hat{\boldsymbol{\Phi}}(k+m-1)\\ \hat{\boldsymbol{\Phi}}(k+m) &= \begin{bmatrix}[-\hat{\boldsymbol{v}}_{\boldsymbol{h}}(k+m) & \ldots & -\hat{\boldsymbol{v}}_{\boldsymbol{h}}(k+m+1-i)]^T\\ [\hat{\boldsymbol{p}}_{\boldsymbol{h}}(k+m) & \ldots & \hat{\boldsymbol{p}}_{\boldsymbol{h}}(k+m-j)]^T\end{bmatrix}.\end{aligned} \tag{19}$$

C. Integrated Human–Robot Collaboration System

We now extend the optimal control formulation (15) by further considering human trust in robot to assure smooth and effective HRC while the assembly is efficient. Based on previous studies in human factors [10], [35], human trust in robot depends on prior trust, robot performance, human performance, and fault occurrences. In this paper, we utilize our previous results [16] of a time-series dynamic model of human-to-robot trust in HRC manufacturing. Note that in manufacturing environments, the required tasks of the robot and the environment itself are fixed, and thus it is reasonable to assume that the robot performs the tasks as planned and the main performance metric is the robot flexibility. With that mindset, we use the following computational model for human-to-robot trust, denoted as $T(k)$:

$$T(k) = aT(k-1) + bP_R(k-1) + cP_H(k-1) \tag{20}$$

where P_R and P_H represent robot and human performance, respectively. The coefficients a, b, and c are constants to be determined for a specific application and individual. As we described in [16], a common method for determining these parameters is to use the autoregressive–moving-average (ARMA) model for the data collected during the training of the experiments. First, human and robot performances and human trust in robot are collected. The performances can be measured objectively but the human trust can only be realized subjectively. Since it is difficult to ask for subjective human trust to the robot with a high sampling frequency, this value is asked from the human worker after he/she finishes each task trial, i.e., assembles one product. Next, the trust value at each trial and the average value of performances during that trial are stored. Finally, using ARMA in MATLAB System Identification Toolbox, the coefficients a, b, and c are fit to (20) using the stored data for each individual participant in the given task. In this paper, we consider human working speed and his/her coordination with the robot co-worker as the two main measures of performance P_H. We asked two human workers who are experts in performing assembly tasks to perform the same set of operations and collected their data as the reference human working speed. This ideal speed is a function of the path progress and we denote it as $v_{\text{ref}}(S) \in [0,1]$, where $S \in \mathbb{R}^+$ is the path progress (of the human, S_H, or the robot, S_R). Any difference between the human working speed and this reference value along the path indicates low human performance. Moreover, if the human path progress, S_H, is less than that of robot, S_R, the human is considered not to do a good job compared to his/her robot co-worker. We define P_H as

$$\begin{aligned}P_H(k+1) = \overline{P_H} &- w_1(S_R(k) - S_H(k))H(S_R(k) - S_H(k))\\ &- w_2\left|\frac{\hat{\boldsymbol{i}}_{\boldsymbol{h}}\boldsymbol{v_h}(k)}{\overline{v_H}} - v_{\text{ref}}(S_H(k))\right|\end{aligned} \tag{21}$$

where $\overline{P_H} = 1$ and $\overline{v_H}$ are the maximal values of human performance and human hand velocity, with $w_1 + w_2 = 1$, $w_i \in [0,1], i = 1,2$, and $H(.)$ is the Heaviside step function. The robot is desired to follow the human progress during the interaction. Thus, we define P_H such that it does not decrease if the human progress leads, i.e., if $S_H > S_R$. The robot performance is defined in a fashion similar to the human performance by including both robot working speed and its flexibility in keeping up with the human co-worker. Here, any difference between S_R and S_H results in low P_R since the robot is required to follow the human worker's progress. The definition of P_R is hence

$$\begin{aligned}P_R(k+1) = \overline{P_R} &- w_3|S_R(k) - S_H(k)|\\ &- w_4\left|\frac{v(k)}{\overline{v_R}} - v_{\text{ref}}(S_R(k))\right|\end{aligned} \tag{22}$$

where $\overline{P_R} = 1$ is the maximal value of robot performance with $w_3 + w_4 = 1,\ w_i \in [0,1], i = 3,4$.

Remark 4: In both human and robot performance models, the first measure depicts the quality harmony between the two agents. Lack of coordination of the progress of the agents results in frustration, higher error failure rate, and a decline of overall performance of the human–robot team [36]. However, the coordination between the agents is not sufficient. If both agents perform the task with harmony but slowly, the overall progress would be slow. Hence, the second measure depicts each agent's individual progress rate toward finishing the task. In summary, the agent performance is high only if it performs the task with a fast pace and in accordance with the other agent. •

In the integrated HRC system, the robot is required to meet the human expectations and preferences. Thus, it is desired to control the robot speed such that it follows the human path progress while human trust in robot is higher than a threshold value. This can be formulated as the following NMPC problem:

$$\begin{aligned}\min_{v(0),\ldots,v(N-1)} \sum_{i=1}^{N}\{&||S_R(i) - S_H(i)||_Q\\ &+ ||v(i) - \overline{v}||_R + ||T(i) - \overline{T}||_W\}\end{aligned} \tag{23}$$

subject to (13), (14), (20)–(22), and $T(i) > \underline{T}$, where $\underline{T}$ and $\overline{T}$ are the minimal threshold and maximal values of trust determined based on task specifications and individual preferences [16]. Here, minimizing cost has similar effects as of (15) and also pushes T to $\overline{T}$. As explained in Section III-A, a high-level path planning approach is developed based on the trust of human in robot. We define a lookup table for selecting the robot path based on the average trust of human in robot, T_{avg}. This table specifies a path corresponding to

Fig. 7. Robot emotions (facial expressions). (a) Happy face. (b) Worried face. (c) Bored face.

each range of trust values. At the end of each travel (i.e., at the initial and final positions), the high-level path planner matches T_{avg} with the ranges in the table and selects the corresponding path.

D. Integrated Human–Robot Collaboration System With Emotional Expressions

Emotions help people to interact with each other more naturally and intuitively. Many studies showed that cognition and emotion play interrelated roles in intelligent decision-making, planning, communication, and social interaction [37]. For example, emotion helps to prioritize different concerns by guiding the attention toward important matters and away from distractions [12]. The possible benefits of integrating emotion into the robots resulted in the design of emotion-inspired mechanisms such as Kimset, ERWIN, Kobian, NAO, Flobi, iCAT, Robokinds, and Geminoids [8], [13]. While these robots are mostly utilized for social services [38], their applications can be effectively extended to other HRI scenarios. Thus, we borrow this idea to design an emotion-inspired robot in a hybrid manufacturing cell. In this setting, the robot emotion plays a role of nonverbal communication, informing the human about a possible safety or efficiency concern, thus increases safety and performance.

The robot emotion is added to the integrated HRC system to make the interaction more intuitive and to alert safety or efficiency concerns to the human worker by displaying a facial expression (see Figs. 7 and 9) both on the robot head screen and a computer information screen. To make the interaction more human-like, we also add eye motion to the robot facial expression and let it follow the human hand all the time. Under the nominal condition when there is no chance of immediate collision and the human performance is relatively coordinated with the robot, the robot expresses a happy face. However, the robot end-effector might collide with the human hand if the distance between them is small. We call this distance the safety index and define it as $I_S = |\boldsymbol{x} - \boldsymbol{p_h}|$, where $\boldsymbol{x}$ and $\boldsymbol{p_h}$ are the position of the robot's end-effector and human worker's hand, respectively. We define L_S as the threshold value of I_S for the safe interaction of the human and the robot. When $I_S > L_S$, the robot and information screens display a happy or bored face depending on the human and robot progress. If there is a possible collision between the human and the robot, i.e., $I_S \leq L_S$, the robot stops working and a worried face will be displayed on the robot and information screens. As soon as the human worker moves his/her hand away from the robot and $I_S > L_S$, the robot facial expression changes from worried to either happy or bored emotion and the robot will continue to move on the planned path. We also denote the difference between the robot and human progress as the efficiency index $I_E = S_R - S_H$. Let $L_E = 0.5$ be the threshold value of I_E for efficient interaction of the human and the robot. If $I_E > L_E$ and the robot's relative distance to human is in the safe region, the robot progress is remarkably higher than human and Baxter displays a bored face to the human expressing that his/her progress is too slow. This emotion contributes to efficiency by encouraging the participant to keep pace with the robot.

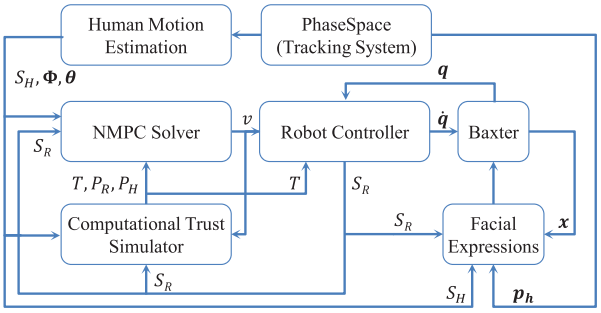


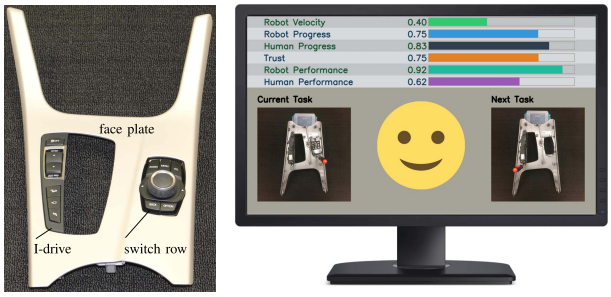
Fig. 8. Block diagram of the HRC control framework.

E. Control Framework Diagram

Fig. 8 shows the block diagram connecting different HRC system components. All of the programs communicate through ROS. The robot controller is coded in Python. It receives the robot joint positions $\boldsymbol{q}$ and path position velocity v from Baxter and the NMPC solver, respectively. It calculates the robot path progress S_R and the corresponding joint velocities $\dot{\boldsymbol{q}}$ using v. These joint velocity commands are sent to Baxter. The human kinematics estimator is coded in C++ and receives the time and 3-D position data from the PhaseSpace tracking system workstation. It calculates human path progress S_H, the regressand vector $\boldsymbol{\Phi}$, and estimates the filter vector $\boldsymbol{\theta}$. Computational trust simulator is used for the integrated experiment. The input is the human progress data S_H from the estimator, robot translational velocity v from the NMPC solver, and robot progress S_R from the robot controller. The trust simulator uses the input data and returns the human performance P_H, robot performance P_R, and estimated computational trust T as the output. We add ROS interface to the NMPC toolbox written in C++ [33]. It receives all of the data and calculates the next control input v. The facial expressions block uses values of robot end-effector position $\boldsymbol{x}$, human's hand position $\boldsymbol{p_h}$, robot progress S_R, and human progress S_H, for calculating the safety and efficiency indexes.

F. Transparency

To keep the human awareness about the robot actions and reduce confusion, we improve transparency by showing different states of the HRC systems (Section IV) used for controlling the robot, speed of the robot v, and the computational trust of human to the robot T, through an information screen. If the speed of the robot is adjusted manually, only the robot velocity v, human progress S_H, and robot progress S_R will



(a) (b)

Fig. 9. Completed assembly and information screen. (a) Assembly parts. (b) Information screen.

be shown on the screen. Fig. 9(b) shows this interface for the integrated HRC system with emotional expressions.

V. Experiments With a Human-in-the-Loop

A. Experimental Design and Participants

We evaluated the effects of implementation of the proposed control conditions through an experimental case study. A within-subject test with Latin square design test order was performed under four different control conditions: 1) manual control condition (**C1**); 2) pHRI-based control condition (**C2**); 3) integrated pHRI- and trust-based control condition (**C3** or integrated in short); and 4) integrated pHRI- and sHRI-based control condition considering both trust and emotion display (**C4** or emotion-integrated in short). Twenty participants (6 females and 14 males) with an age ranging from 25 to 36 (average 29.9) years, participated in the experiments. Similar sample size has been used in prior works on experimental studies of HRI [39]–[41]. Half of the participants had no experience in working with a robot before. Based on the preliminary results of our previous work [14] and the discussed literature in Sections I and II for the collaboration of a human and a robot under these control conditions, we hypothesize the following.

- *H1:* As we move from **C1** to **C4**, the human perceived workload decreases.
- *H2:* As we move from **C1** to **C4**, the human perceives higher trust toward the robot.
- *H3:* As we move from **C1** to **C4**, the human perceives a higher usability of the robot.
- *H4:* The robot average velocity and the assembly time do not change significantly in all conditions.

B. Measurements and Scales

The independent variable of this paper is the control condition (**C1**-**C4**) utilized based on the level of the interaction. The dependent variables are the following subjective and objective measures.

1) *Workload:* At the end of each experiment, the subjective overall workload was measured using the NASA task load index (TLX) method [42]. This measure can vary from 5 to 100.
2) *Subjective Trust:* At the end of each experiment, the subjective trust of the participant in robot was measured using a human trust in automation questionnaire [43]. The questionnaire was adjusted to be suitable for assessments of robots.
3) *User Satisfaction:* At the end of each experiment, the subjective satisfaction with the usability of the robot was measured using the IBM usability satisfaction questionnaire [44].
4) *Robot Velocity:* The average robot velocity (v_{avg}) is a measure of the efficiency of the human–robot team in accomplishing the task and was calculated after the experiment was finished.
5) *Assembly Time:* This is the time spent to assemble $N_p = 3$ products, used as another measure of efficiency.

C. Apparatus

Fig. 2 shows the equipment and framework architecture used for the experiment. We used a humanoid manufacturing research robot, Baxter, made by Rethink Robotics for our experiment. The robot is suitable for lightweight material handling and intelligent assembly, especially for small batch productions. It has a removable base and two redundant arms with seven DoFs on each arm. There is a rotary screen attached to the top of the robot as its head. We use the PhaseSpace motion tracking system to capture the human hand motion. The tracking system includes a set of cameras, a set of active markers, and a workstation for tracking rigid bodies in a 3-D environment. The other nodes of the framework were set up on a local computer as described in Section IV-E.

D. Task Scenario

The selected task in the experiment is from the automotive assembly industry, and the goal is to assemble three parts together to form a BMW center console. This task is similar to the assembly task that a human worker performs in the manufacturing assembly lines and requires a high level of HRC in which a human and a robot work together in a hybrid-cell [1] as shown in Fig. 1(b). The assembly parts include face plate, I-drive, and switch row [see Fig. 9(a)]. Each participant was asked to continuously assemble these parts together. The current and next assembly parts required to be assembled, together with some other task information, are shown to the participant via the information screen [see Fig. 9(b)]. In the current settings, it is difficult for the robot to grab the face plate and bring it to the human due to its geometry. Therefore, the participant was required to fetch the face plate from the human bin and place it in the shared workspace. At the same time, the robot fetched the I-drive and switch row from the robot bin to the shared workspace. The participant assembled these two parts on the face plate using a screwdriver to form the final car center console. This process was repeated thrice ($N_p = 3, h = 1$, and $r = 2$) for each of the four control conditions of the experiment. Fig. 10 shows the collaboration of a participant and the robot during an experiment under the emotion-integrated control condition (**C4**). A brief summary of the task procedure for the integrated control condition is provided in Algorithm 2.

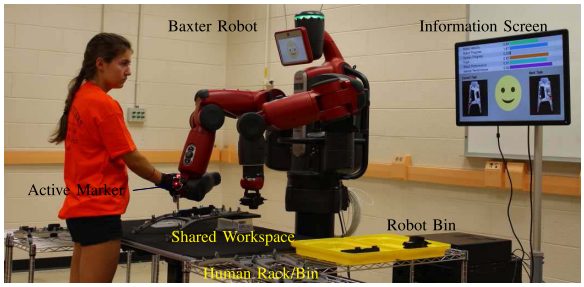


Fig. 10. Experiment scenario.

Algorithm 2: Implementation of the Integrated Control Condition in the Hybrid Cell

1: Move to the shared workspace ▷ Initial point
2: $c_r \leftarrow 0$ ▷ Reset the number of traveled paths
3: Subscribe(T) ▷ Subscribe to the trust simulator
4: $T_{\text{avg}} \leftarrow T$ ▷ Initialize the average trust value
5: $\boldsymbol{p} \leftarrow choose\_path\_fwd(T_{\text{avg}})$ ▷ Select forward path
6: **repeat**
7: $Move\_Robot(\boldsymbol{p})$ ▷ Move to the robot bin
8: $Pick\ up$ ▷ Pick up a part from the robot bin
9: $c_r \leftarrow c_r + 1$ ▷ Increase number of traveled paths
10: $Update(T_{\text{avg}})$
11: $\boldsymbol{p} \leftarrow choose\_path\_bwd(T_{\text{avg}})$ ▷ Select backward path
12: $Move\_Robot(\boldsymbol{p})$ ▷ Move to the shared workspace
13: $Place$ ▷ Place the part near the worker
14: $Update(T_{\text{avg}})$
15: $\boldsymbol{p} \leftarrow choose\_path\_fwd(T_{\text{avg}})$ ▷ Select forward path
16: $c_r \leftarrow c_r + 1$ ▷ Increase number of traveled paths
17: **until** $c_r == 2r \times N_p$

E. Experiment Procedure

Each participant was asked to fill out the consent form and demographic questionnaire. The information regarding the task scenario and the roles of the participant and the robot were explained in detail to the participant. For training purpose, the participant was asked to collaborate with the robot until he/she feels comfortable and familiar with the task and the robot. After the training, the participant performed all four conditions of the experiment based on Latin square order. Upon completion of each condition, the participant was asked to fill out NASA-TLX, trust, and IBM usability questionnaires.

VI. Results, Analysis, and Discussion

A. Sample HRC System Evolution

Fig. 11 shows a sample HRC system evolution for participant number 15 under emotion-integrated condition (**C4**). The robot performance P_R and robot path progress S_R do not change at the pick-and-place locations where the robot end-effector only moves vertically. At the start of the first task (Region A), both human and robot performances are high. The human moves toward the human bin to bring the face plate ($S_H = 0.5$) while the robot moves toward the robot bin to bring the next part ($S_R = 0.25$) to the shared workspace. As the human progress reaches $S_H = 0.5$, the difference in human and robot motion progress results in decrease of the robot performance P_R, which results in lower trust value

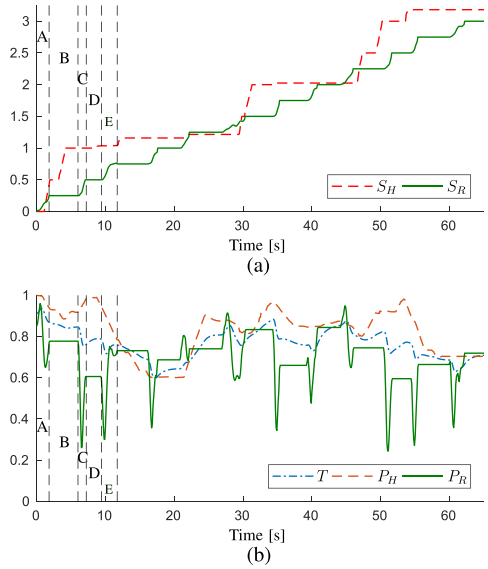


Fig. 11. Sample HRC system. (a) Human and robot progress. (b) Trust, human, and robot performances.

as well. When the robot reaches the robot bin (Region B), it picks up a part (an I-drive). At the same time, human reaches the shared workspace and since S_H is greater than S_R, P_H does not change significantly. However, as the human velocity deviates from the reference velocity, P_H decreases slightly. The trust of human in robot has the same trend. Next, the robot starts to move back to the shared workspace (Region C). At this point, the robot performance gets updated. Since there is a major difference between human and robot progress, the robot performance P_R drops significantly at first and then recovers when the robot reaches the shared workspace where $S_R = 0.5$. As the robot places the object (Region D), human path progress is greater than robot progress. Moreover, human velocity matches the reference velocity, and hence, the human performance P_H is high and trust increases. When the robot starts to moves back for the next part (a switch row) (Region E), the discrepancy between the human and robot path progress is high at first but decreases and robot performance recovers. Similar trends repeat until the task is accomplished completely.

B. Statistical Analysis

For each dependent variable explained in Section V-B, a one-way repeated measure analysis of variance (ANOVA) was conducted to determine whether there were statistically significant differences in dependent variables over the interaction type. Mauchly's test of sphericity shows that the assumption of sphericity was violated for human trust in robot ($\chi^2(2) = 26.626$ and $p < 0.001$), robot usability ($\chi^2(2) = 16.317$ and $p = 0.006$), and perceived workload index ($\chi^2(2) = 31.523$ and $p < 0.001$), but it is valid for robot average velocity ($\chi^2(2) = 6.744$ and $p = 0.241$), and assembly time ($\chi^2(2) = 10.846$), $p = 0.055$. Therefore, Greenhouse–Geisser correction was applied ($\epsilon = 0.567$, $\epsilon = 0.634$, and $\epsilon = 0.509$ for human trust-in robot,

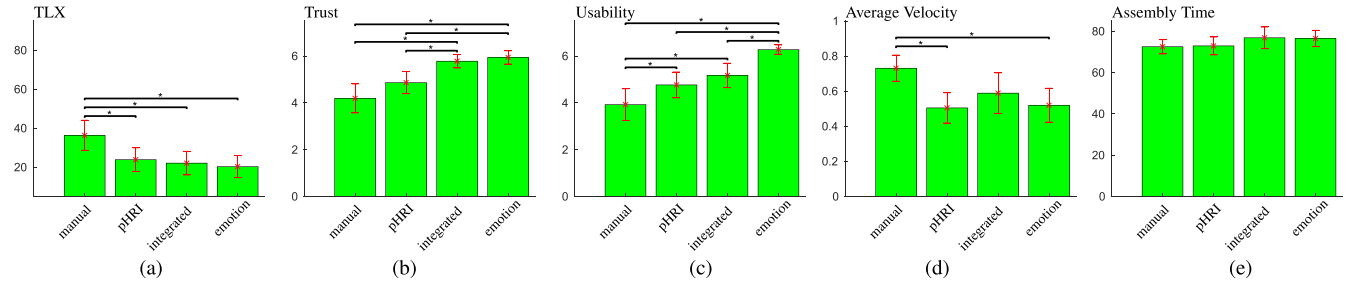


Fig. 12. Results. The horizontal bars with the asterisk on top show the statistically significant changes of the interaction levels. (a) Human workload. (b) Human trust in robot. (c) Robot usability. (d) Robot average velocity. (e) Assembly time.

robot usability, and perceived workload index, respectively). The results with upper and lower bounds for the 95% confidence interval are presented in Fig. 12 and described as follows.

1) Perceived Workload: The manipulation of the interaction type elicited statistically significant changes in workload between the different conditions, $F(1.526, 29.002) = 8.930$, $p < 0.005$, and $\eta_p^2 = 0.320$. Post-hoc analysis with a Bonferreoni adjustment showed that the workload under the manual condition ($M = 36.367$ and $SD = 3.686$) is significantly higher compared to the other conditions [pHRI ($M = 23.917$ and $SD = 2.915$), integrated ($M = 22.117$ and $SD = 2.879$), and emotion-integrated ($M = 20.383$ and $SD = 2.700$)].

2) Human Trust in Robot: The manipulation of the interaction type elicited statistically significant changes in trust, $F(1.702, 32.338) = 17.839$, $p < 0.001$, $\eta_p^2 = 0.484$. Post-hoc analysis with a Bonferreoni adjustment showed that trust both under the integrated ($M = 5.783$ and $SD = 0.137$) and emotion-integrated ($M = 5.946$ and $SD = 0.136$) conditions are significantly higher compared to the trust under the manual ($M = 4.196$ and $SD = 0.296$) and pHRI ($M = 4.874$ and $SD = 0.225$) conditions.

3) Usability: The manipulation of the interaction type elicited statistically significant changes in usability, $F(1.901, 36.116) = 28.671$, $p < 0.001$, $\eta_p^2 = 0.601$. Post-hoc analysis with a Bonferreoni adjustment showed that usability under the manual condition ($M = 3.930$ and $SD = 0.325$) is significantly lower compared to the usability under the other conditions including pHRI ($M = 4.770$ and $SD = 0.263$), integrated ($M = 5.175$ and $SD = 0.248$), and emotion-integrated ($M = 6.280$ and $SD = 0.096$). Moreover, it shows that usability under the emotion-integrated conditions is significantly higher compared to all of the other conditions.

4) Robot Average Velocity and Assembly Time: The manipulation of the interaction type elicited statistically significant changes in robot average velocity, $F(3, 57) = 7.299$, $p < 0.001$, $\eta_p^2 = 0.278$. Post-hoc analysis with a Bonferreoni adjustment showed that the robot average velocity under the manual condition ($M = 0.731$ and $SD = 0.036$) is significantly higher compared to the pHRI ($M = 0.505$ and $SD = 0.041$) and integrated-emotion ($M = 0.520$ and $SD = 0.046$) conditions. The interaction type did not elicit statistically significant changes in assembly time, $F(3, 57) = 2.091$, $p > 0.05$, $\eta_p^2 = 0.099$.

C. Discussion

It can be seen from the results that, in general, as we augment physical and social capabilities into the framework and change the control condition from **C1** to **C4**, HRI improves and at least one of the subjective measures (i.e., human perceived workload, human trust in robot, and robot usability) improves. More specifically, although none of the hypotheses are completely true, they are partially correct. In **H1**, the human perceived workload does not constantly drop from **C1** to **C4**. However, compared to the manual adjustment of robot speed (**C1**), using any autonomous controllers (**C2** to **C4**) decreases the human perceived workload significantly. Two main reasons for this could be that under **C1** a participant has to either adapt his/her work pattern to that of the robot or pay more attention to the robot velocity and adjust it accordingly. Both of these result in more perceived workload of the participant.

Regarding **H2**, considering merely pHRI for controlling the robot velocity does not impact the human trust in the robot. However, compared to **C1** and **C2** frameworks, as we augment more social capabilities to **C3** and **C4**, the trust of human in robot significantly improves. The main reason for this significant change might be that the participant becomes more confident in robot since the human and robot performances are shown to them in both **C3** and **C4**. The trust-based controller and path selection improve the trust by clarifying robot's intent and behaving more predictably and human-like.

The impact on robot usability is more obvious relative to the other subjective measures. Compared to **C1**, in any framework with autonomous controller (**C2** to **C4**), the usability increases. This, together with **H1**, suggests that an autonomous controller results in higher usability and lower perceived workload. Moreover, the utilization of emotions in **C4** results in higher robot usability compared to the rest of the conditions (**C1** to **C3**). This is probably because adding emotions makes the interaction more human-like and appealing.

In terms of efficiency, the results indicate that **H4** is not completely true. The robot average velocity is significantly higher in **C1** compared to **C2** and **C4**. We observed two major trends in **C1**: 1) some of the participants felt competitive toward the robot and intentionally tried to challenge themselves and the robot by setting the robot velocity at a high value rather than adjusting it based on their speed and 2) some of them simply were too engaged in the experiment and forgot to adjust the robot speed. These observations also

justify the high value of human workload in **C1**. Note that the rest of the participants paid more attention to the robot velocity to determine the speed adjustments, so they also perceived high workload. The variation of assembly time in different conditions matches **H4** and does not change significantly. This is probably because although participants used higher robot velocities in **C1**, they could not keep up with the robot progress and had to spend some time at the end of the experiment to finish the assembly tasks.

VII. Conclusion

We proposed a novel framework for HRC in manufacturing assembly lines. We described the kinematics of such HRC system and demonstrated how this framework includes both pHRI and sHRI for finding the optimal velocity of the robot. We also demonstrated the problem of moving the robot end-effector with an arbitrary velocity along a given path and implemented the control method for the HRC system. We experimentally evaluated this framework by designing an HRC testbed. The results show that the pHRI- and sHRI-based autonomous controllers can reduce human workload while maintaining the overall performance of the human–robot team compared to the manual adjustments of the robot velocity. Moreover, it is shown in our experiments that human trust in robot can be remarkably increased if sHRI factors are integrated into the pHRI-based framework. Furthermore, the robot usability can be significantly increased if emotion is added to the integrated framework while the objective measures do not show statistical significance among the automated conditions.

Acknowledgment

The authors would like to thank the BMW US Manufacturing Company, Spartanburg, SC, USA, for loaning Baxter to the Interdisciplinary and Intelligence Research (I^2R) Laboratory at Clemson University, Clemson, SC, USA.

References

[1] J. Shi, G. Jimmerson, T. Pearson, and R. Menassa, "Levels of human and robot collaboration for automotive manufacturing," in *Proc. Workshop Perform. Metrics Intell. Syst. (PerMIS)*, New York, NY, USA, 2012, pp. 95–100.

[2] C. Breazeal, "Social interactions in HRI: The robot view," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 34, no. 2, pp. 181–186, May 2004.

[3] S. Haddadin and E. Croft, "Physical human–robot interaction," in *Springer Handbook of Robotics*. Cham, Switzerland: Springer, 2016, pp. 1835–1874.

[4] A. De Santis, B. Siciliano, A. De Luca, and A. Bicchi, "An atlas of physical human—Robot interaction," *Mech. Mach. Theory*, vol. 43, no. 3, pp. 253–270, 2008.

[5] A. M. Zanchettin, N. M. Ceriani, P. Rocco, H. Ding, and B. Matthias, "Safety in human-robot collaborative manufacturing environments: Metrics and control," *IEEE Trans. Autom. Sci. Eng.*, vol. 13, no. 2, pp. 882–893, Apr. 2016.

[6] A. Bicchi, M. Peshkin, and J. Colgate, "Safety for physical human–robot interaction," in *Springer Handbook of Robotics*. Berlin, Germany: Springer, 2008, pp. 1335–1348.

[7] A. Steinfeld *et al.*, "Common metrics for human-robot interaction," in *Proc. 1st ACM SIGCHI/SIGART Conf. Hum.-Robot Interact.*, 2006, pp. 33–40.

[8] S. M. M. Rahman and Y. Wang, "Dynamic affection-based motion control of a humanoid robot to collaborate with human in flexible assembly in manufacturing," in *Proc. ASME Dyn. Syst. Controls Conf.*, 2015, p. V003T40A005.

[9] E. Guizzo, "Sawyer: Rethink robotics unveils new robot," *IEEE Spectr.*, Mar. 2015.

[10] P. A. Hancock, D. R. Billings, K. E. Schaefer, J. Y. Chen, E. J. De Visser, and R. Parasuraman, "A meta-analysis of factors affecting trust in human-robot interaction," *Hum. Factors, J. Hum. Factors Ergonom. Soc.*, vol. 53, no. 5, pp. 517–527, 2011.

[11] J. D. Lee and K. A. See, "Trust in automation: Designing for appropriate reliance," *Hum. Factors, J. Hum. Factors Ergonom. Soc.*, vol. 46, no. 1, pp. 50–80, 2004.

[12] R. W. Picard and R. Picard, *Affective Computing*, vol. 252. Cambridge, MA, USA: MIT Press, 1997.

[13] C. Breazeal, "Emotion and sociable humanoid robots," *Int. J. Hum.-Comput. Stud.*, vol. 59, nos. 1–2, pp. 119–155, 2003.

[14] B. Sadrfaridpour, H. Saeidi, and Y. Wang, "An integrated framework for human-robot collaborative assembly in hybrid manufacturing cells," in *Proc. IEEE Int. Conf. Autom. Sci. Eng. (CASE)*, Aug. 2016, pp. 462–467.

[15] F. Chen, K. Sekiyama, F. Cannella, and T. Fukuda, "Optimal subtask allocation for human and robot collaboration within hybrid assembly system," *IEEE Trans. Autom. Sci. Eng.*, vol. 11, no. 4, pp. 1065–1075, Oct. 2014.

[16] B. Sadrfaridpour, H. Saeidi, J. Burke, K. Madathil, and Y. Wang, "Modeling and control of trust in human-robot collaborative manufacturing," in *Robust Intelligence and Trust in Autonomous Systems*. New York, NY, USA: Springer, 2016, ch. 7, pp. 115–141.

[17] S. M. M. Rahman, Y. Wang, I. D. Walker, L. Mears, R. Pak, and S. Remy, "Trust-based compliant robot-human handovers of payloads in collaborative assembly in flexible manufacturing," in *Proc. IEEE Int. Conf. Autom. Sci. Eng. (CASE)*, Aug. 2016, pp. 355–360.

[18] M. Quigley *et al.*, "ROS: An open-source robot operating system," in *Proc. ICRA Workshop Open Source Softw.*, Kobe, Japan, 2009, vol. 3. no. 3.2, p. 5.

[19] L. Johannsmeier and S. Haddadin, "A hierarchical human-robot interaction-planning framework for task allocation in collaborative industrial assembly processes," *IEEE Robot. Autom. Lett.*, vol. 2, no. 1, pp. 41–48, Jan. 2017.

[20] G. Michalos, S. Makris, J. Spiliotopoulos, I. Misios, P. Tsarouchi, and G. Chryssolouris, "Robo-partner: Seamless human-robot cooperation for intelligent, flexible and safe operations in the assembly factories of the future," *Procedia CIRP*, vol. 23, pp. 71–76, Dec. 2014.

[21] P. Tsarouchi, A.-S. Matthaiakis, S. Makris, and G. Chryssolouris, "On a human-robot collaboration in an assembly cell," *Int. J. Comput. Integr. Manuf.*, vol. 30, no. 6, pp. 580–589, 2016.

[22] A. Cherubini, R. Passama, A. Crosnier, A. Lasnier, and P. Fraisse, "Collaborative manufacturing with physical human–robot interaction," *Robot. Comput.-Integr. Manuf.*, vol. 40, pp. 1–13, Aug. 2016.

[23] L. Peternel, N. Tsagarakis, D. Caldwell, and A. Ajoudani, "Adaptation of robot physical behaviour to human fatigue in human-robot co-manipulation," in *Proc. IEEE-RAS 16th Int. Conf. Humanoid Robots (Humanoids)*, Nov. 2016, pp. 489–494.

[24] J. T. C. Tan, F. Duan, Y. Zhang, K. Watanabe, R. Kato, and T. Arai, "Human-robot collaboration in cellular manufacturing: Design and development," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2009, pp. 29–34.

[25] M. Morioka and S. Sakakibara, "A new cell production assembly system with human–robot cooperation," *CIRP Ann.-Manuf. Technol.*, vol. 59, no. 1, pp. 9–12, 2010.

[26] S. M. LaValle, *Planning Algorithms*. Cambridge, U.K.: Cambridge Univ. Press, 2006.

[27] A. D. Wilson, J. A. Schultz, A. R. Ansari, and T. D. Murphey, "Dynamic task execution using active parameter identification with the baxter research robot," *IEEE Trans. Autom. Sci. Eng.*, vol. 14, no. 1, pp. 391–397, Jan. 2017.

[28] J. A. Marvel, "Performance metrics of speed and separation monitoring in shared workspaces," *IEEE Trans. Autom. Sci. Eng.*, vol. 10, no. 2, pp. 405–414, Apr. 2013.

[29] C. M. Perry, M. A. Sheik-Nainar, N. Segall, R. Ma, and D. B. Kaber, "Effects of physical workload on cognitive task performance and situation awareness," *Theor. Issues Ergonom. Sci.*, vol. 9, no. 2, pp. 95–113, 2008.

[30] F. Pfeiffer and R. Johanni, "A concept for manipulator trajectory planning," *IEEE J. Robot. Autom.*, vol. RA-3, no. 2, pp. 115–123, Apr. 1987.

[31] A. S. Deo and I. D. Walker, "Overview of damped least-squares methods for inverse kinematics of robot manipulators," *J. Intell. Robot. Syst.*, vol. 14, no. 1, pp. 43–68, 1995.

[32] P. Chiacchio, S. Chiaverini, L. Sciavicco, and B. Siciliano, "Closed-loop inverse kinematics schemes for constrained redundant manipulators with task space augmentation and task priority strategy," *Int. J. Robot. Res.*, vol. 10, no. 4, pp. 410–425, 1991.
[33] L. Grüne and J. Pannek, *Nonlinear Model Predictive Control: Theory and Algorithms*. London, U.K.: Springer-Verlag, 2011.
[34] A. K. Tangirala, *Principles of System Identification: Theory and Practice*. Boca Raton, FL, USA: CRC Press, 2014.
[35] J. Lee and N. Moray, "Trust, control strategies and allocation of function in human-machine systems," *Ergonomics*, vol. 35, no. 10, pp. 1243–1270, 1992.
[36] D. J. Bruemmer, D. I. Gertman, C. W. Nielsen, D. A. Few, and W. D. Smart, "Supporting complex robot behaviors with simple interaction tools," in *Human Robot Interaction*. Rijeka, Croatia: InTech, 2007.
[37] C. Breazeal and R. Brooks. "Robot emotion: A functional perspective" in *Who Needs Emotions: The Brain Meets the Robot*, J.-M. Fellous and M. Arbib, Eds. Oxford, U.K.: Oxford Univ. Press, pp. 271–310, 2005.
[38] T. Fong, I. Nourbakhsh, and K. Dautenhahn, "A survey of socially interactive robots," *Robot. Auto. Syst.*, vol. 42, nos. 3–4, pp. 143–166, 2003.
[39] M. Swangnetr and D. B. Kaber, "Emotional state classification in patient–robot interaction using wavelet analysis and statistics-based feature selection," *IEEE Trans. Human–Mach. Syst.*, vol. 43, no. 1, pp. 63–75, Jan. 2013.
[40] Y. Huang, Y. S. Yong, R. Chiba, T. Arai, T. Ueyama, and J. Ota, "Kinematic control with singularity avoidance for teaching-playback robot manipulator system," *IEEE Trans. Autom. Sci. Eng.*, vol. 13, no. 2, pp. 729–742, Apr. 2016.
[41] J. Mainprice, R. Hayne, and D. Berenson. (Jun. 2016). "Goal set inverse optimal control and iterative re-planning for predicting human reaching motions in shared workspaces." [Online]. Available: https://arxiv.org/abs/1606.02111
[42] S. G. Hart and L. E. Staveland, "Development of NASA-TLX (task load index): Results of empirical and theoretical research," *Adv. Psychol.*, vol. 52, pp. 139–183, Dec. 1988.
[43] J.-Y. Jian, A. M. Bisantz, and C. G. Drury, "Foundations for an empirically determined scale of trust in automated systems," *Int. J. Cognit. Ergonom.*, vol. 4, no. 1, pp. 53–71, 2000.
[44] J. R. Lewis, "IBM computer usability satisfaction questionnaires: Psychometric evaluation and instructions for use," *Int. J. Hum.-Comput. Interact.*, vol. 7, no. 1, pp. 57–78, 1995.

Behzad Sadrfaridpour (S'16) received the B.S. and M.S. degrees in mechanical engineering from Tabriz University, Tabriz, Iran, and the University of Tarbiat Modares, Tehran, Iran, in 2008 and 2011, respectively. He is currently pursuing the Ph.D. degree in mechanical engineering with Clemson University, Clemson, SC, USA.

His current research interests include human–robot interaction, manufacturing automation, and motion planning.

Mr. Sadrfaridpour was a recipient of the Best Student Conference Paper Award at the 12th IEEE Conference on Automation Science and Engineering (2016).

Yue Wang (S'06–M'11–SM'16) received the B.S. degree from Shanghai University, Shanghai, China, in 2005, and the M.S. and Ph.D. degrees from the Worcester Polytechnic Institute, Worcester, MA, USA, in 2008 and 2011, respectively.

She was a Post-Doctoral Researcher with the Electrical Engineering Department, University of Notre Dame, Notre Dame, IN, USA. She is currently the Warren H. Owen–Duke Energy Assistant Professor of Engineering with Clemson University, Clemson, SC, USA. Her current research interests include cooperative control and decision-making for human–robot collaboration systems, cyber-physical systems, and multiagent systems.

Prof. Wang is the Chair of the IEEE CSS TC on Manufacturing Automation and Robotic Control and a member of the IEEE RAS TC on Multi-Robot Systems. She was a recipient of the Air Force Young Investigator Award in 2016 and the NSF CAREER Award in 2015.