# A Dynamic Planner for Safe and Predictable Human-Robot Collaboration

Andrea Pupa, *Member, IEEE*, Marco Minelli, and Cristian Secchi, *Senior Member, IEEE*

***Abstract*—The new face of modern industrial scenarios involves shared workspaces where humans and robots work closely together. To ensure safe human-robot collaboration (HRC), regulations have been updated introducing the ISO/TS15066. However, complying with these regulations often leads to inefficient behavior, such as unnecessarily reducing robot speed or unpredictably changing the robot path, which may negatively affect the operator perception of the robot. In this letter an optimal approach to address together these two issues is proposed. Starting from a desired final configuration, the framework plans a collision-free trajectory for the robot. Subsequently, predictability is taken into account and a set of virtual tubes into which the path of the robot can move is built. Lastly, an optimization problem is solved online to ensure that the robot stays within these tubes and the velocities are compliant with the ISO/TS 15066. The proposed approach has been experimentally validated in two different scenarios: one composed by a mobile manipulator, i.e. a UR10e mounted on a Neobotix MPO-500, and one composed by only a collaborative manipulator, i.e. a UR5e.**

***Index Terms*—Human-robot collaboration, safety in HRI, human-aware motion planning, optimization and optimal control.**

## I. Introduction

INDUSTRIAL applications where collaborative robots are used in close proximity to human operators are growing rapidly. Despite the increase of the flexibility in the production, this new paradigm requires great prudence in guaranteeing the safety of operators. For this reason, the safety standards have been updated accordingly [1]. The ISO 10218-1 and the ISO 10218-2 [2], [3] standards codify four different collaborative modes: *safety-rated monitored stop* (SMS), *hand guiding* (HG), *speed and separation monitoring* (SSM) and *power and force limiting* (PFL). For each collaborative mode, the technical specification ISO/TS 15066 [4] provides guidelines on the risk assessment. Typically, SSM is adopted in industrial application. In this collaborative mode the robot speed must be adapted according to both the human-robot distance and the human velocity. In some cases, this approach may be too conservative and the speed can be unnecessarily limited.

Manuscript received 13 June 2023; accepted 12 November 2023. Date of publication 20 November 2023; date of current version 28 November 2023. This letter was recommended for publication by Associate Editor N. Yamanobe and Editor G. Venture upon evaluation of the reviewers' comments. *(Andrea Pupa and Marco Minelli contributed equally to this work.) (Corresponding author: Andrea Pupa.)*

The authors are with the Department of Sciences and Methods for Engineering, University of Modena and Reggio Emilia, 42122 Reggio Emilia, Italy (e-mail: andrea.pupa@unimore.it; marco.minelli@unimore.it; cristian.secchi@unimore.it).

This letter has supplementary downloadable material available at https://doi.org/10.1109/LRA.2023.3334977, provided by the authors.

Digital Object Identifier 10.1109/LRA.2023.3334977

The problem of ensuring safety and implementing collision avoidance behavior HRC applications has been extensively addressed in the literature. In [5] a solution to estimate the future human occupancy volume based on human kinematic model is presented. This occupancy volume is then exploited to reduce the robot velocity along the path. In [6] the scaling procedure is exploited to build a safety framework for multi-robot collaborative application. The robots speed are reduced in order to prevent that a safety index falls below a certain value. When the scaling procedure is not enough, an emergency stop is applied.

Adapting the robot speed along a pre-planned path is not always the best solution. In some cases it would be better to modify the pre-planned path to increase the performances. In [7] the concept of static and kinetostatic danger field is exploited to avoid collisions with human operators. Similarly, in [8] the authors generates at runtime collision-free trajectories building a potential field around the whole robot body. In [9] a combination of attractive and repulsive potential field, namely virtual fixtures, is used to guarantee safety in a teleoperated environment. In [10] the authors propose an efficient control scheme for safe HRC. The approach exploits the use of attractive and repulsive danger field to generate constraints on the control inputs. Despite the simplicity of implementation and their effectiveness in the case of simple environments, potential fields are subjected to the problem of local minima that may compromise the task execution.

To avoid this, many researchers focused in leveraging on optimization-based algorithms. In this way, it is possible to obtain a minimum deviation from the desired path, while ensuring safety through the constraints in the optimization problem. In [11] the authors propose the use of an optimization problem to force the robot to stay inside a safe set, evaluating the variation of a safety index. In [12], an optimization-based control algorithm that addresses the safety while trying to preserve the desired trajectory is proposed. The strategy exploits the use of control barrier functions [13] around the robot body to maintain a collision-free trajectory while fulfilling the ISO/TS 15066. In [14] the authors propose to integrate an online planning strategy with a safety-aware scaling optimization algorithm. The robot moves along a planned path adapting the speed based on the safety standards constraint. When the reduction of the speed is too high, the current path has become largely inefficient. The planner is then triggered to plan a new collision-free trajectory, allowing the robot to reach the desired goal faster.

In order to improve collaboration, it is essential to analyze how human operators perceive speed and path changes.

In [15] the authors introduce the concept of the Expectable Motion Unit (EMU), which allows the embedding of human factors into the control problem. The EMU ensures the reduction of involuntary motions in human operators by imposing a velocity

constraint, addressing startle or surprise reactions. Furthermore, in [16] the EMU is used inside an MPC control framework to generate deviations for improving the robot performance.

However, such deviations from the expected path may have a negative impact on the human operator. According to [17], [18], human operators feel safer when robots adhere closely to the expected path, resulting in more predictable robot behavior. Consequently, planner strategies that dynamically adjust to the current human intentions, such as [19], might have a negative impact on human operators. On the other hand, it is crucial to adapt robot behavior to the human operator with whom it is collaborating, based on factors such as mutual trust [20]. Indeed, there might be some operators who are *confident* and accept deviations from paths, while others may be *unconfident* users who prefer predictable robot behavior.

In this letter we propose a novel framework for trajectory planning that takes inspiration from these approaches to enhance the HRC while ensuring safety. Given a desired final configuration, the framework firstly plans a collision-free trajectory for the robot and establishes a subspace, i.e. a set of tubes, where it can move freely without collisions. These tubes allow controlled deviations from the nominal path based on the operator preferences, By incorporating these deviations, the framework ensures that the robot behavior remains as predictable as desired by the human operator. Secondly, the framework computes the inputs that enable the robot to follow the desired path while guaranteeing compliance with the SSM limit and remaining within the established subspace. By integrating these elements, the proposed framework promotes efficient HRC by explicitly considering both safety and collaboration preferences.

The main contributions of this letter are:

- A framework for trajectory planning that takes into account the high dynamism of the environment and human preferences to generate collision-free trajectories associated with the desired level of predictability in an HRC scenario.
- A novel optimization problem that explicitly considers the human operators preferences, i.e. the desired predictability level, to improve the collaboration, while ensuring adherence to SSM limit imposed by safety standards.
- An extensive experimental validation, considering different HRC scenarios.

The letter is organized as follows: in Section II the planning problem with attention to the ISO/TS 15066 is detailed. In Section III the overall architecture is presented, while Section III-A and Section III-B details the most important components of the framework, i.e. trajectory planner and optimizer. Finally in Section IV an experimental validation of the proposed approach is presented and Section V addresses the conclusions.

## II. Problem Statement

Consider an industrial application where a human operator and a $n$-DOFs velocity-controlled collaborative robot have to work together to perform a job. The robot can be modeled as:

$$\dot{\mathbf{q}} = \mathbf{u}, \tag{1}$$

where $\dot{\mathbf{q}} \in \mathbb{R}^n$, and $\mathbf{u} \in \mathbb{R}^n$ are the joints velocities and the controller input, respectively.

During the collaboration, the robot has to perform a set of predefined tasks. Each task is associated with a trajectory $\mathbf{q}_{des}(\cdot) \in \mathbb{R}^n$ that allows the robot to reach a desired final configuration $\mathbf{q}_{des}(t_f) = \mathbf{q}_f \in \mathbb{R}^n$ from an initial configuration $\mathbf{q}_{des}(t_i) = \mathbf{q}_i \in \mathbb{R}^n$. The ISO/TS 15066 [4] provides the guidelines for defining when the trajectory is considered safe according to the SSM collaborative mode. In particular, during the collaboration, the robot-human distance $S_{rh}$ must be:

$$S_{rh}(t) \geq S_p(t) = S_h(t) + S_r(t) + S_s(t) + C + Z_d + Z_r, \tag{2}$$

where $S_p(t)$ is the minimum protective separation distance at time $t$, while $t$ is the current time. $S_h(t)$ represents the contribution to the protective separation distance due to the operator's movements, $S_r(t)$ is the one derived from the robot reaction time and $S_s(t)$ is the contribution caused by the robot stopping time. $C$ represents the intrusion distance, i.e. the distance that a part of the body can intrude into the sensing field before it is detected. $Z_d$ and $Z_r$ are the position uncertainties of the human operator inside the workspace and of the robot system respectively.

Under the assumptions of constant robot velocity in the robot reaction phase, constant acceleration in the stopping phase, human velocity constant in these phases, and recalling that the robot stopping time can be expressed as a function of the actual robot velocity, i.e. $T_s = \frac{v_r(t)}{-a_{\max}}$ where $a_{\max} \in \mathbb{R}^-$ is the maximum deceleration, the distance limit can be expressed as a velocity limit [14]:

$$v_{rh}(t) \leq \sqrt{v_h(t)^2 + (a_{\max}T_r)^2 + 2a_{\max}K(t)} + a_{\max}T_r - v_h(t) \tag{3}$$

where $v_{rh}(t) \in \mathbb{R}$ and $v_h(t) \in \mathbb{R}$ are the scalar velocity of the robot towards the human operator and the scalar velocity of the human operator, respectively. $T_r \in \mathbb{R}$ is the robot reaction time, and $K(t) = C + Z_d + Z_r - S_p(t)$.

To check and ensure that the overall trajectory is compliant with the safety standards, the collaborative area is endowed with a monitoring unit that allows to track the human movements. Several algorithms that deal with the human tracking problem are already available in literature, e.g. skeleton tracking with multiple cameras [21], markers on the human body [22], machine learning techniques [23].

However, satisfying (3) may lead to significant and somehow unpredictable deviations from the nominal trajectory. Thus, the safe behavior of the robot may be perceived unsafe by the user, e.g. leveraging a control strategy that optimally computes the inputs [12]. Indeed, as shown in [17], the human operator should be able to predict the robot behavior during the collaboration task.

In this work, we consider the trajectory of the robot predictable when the trajectory of each link is sufficiently close to the desired one, namely:

$$||\mathbf{x}_i - \mathbf{x}_{des,i}|| \leq \lambda_h \quad \forall i \in \{1, \dots, n\}, \tag{4}$$

where $\mathbf{x}_i \in \mathbb{R}^3$ and $\mathbf{x}_{des,i} \in \mathbb{R}^3$ represent the real Cartesian position of the $i$-th link and the desired one, respectively, and the term $\lambda_h$ represents the maximum deviation that the human operator may accept. This parameter embeds the human characteristics, such as trust or confidence [24], and integrating it into the control strategy translates on constraining each link to lie inside a set of tubes, one around each nominal path.

In a first approximation, we consider a constant $\lambda_h$, which can be related, e.g., to the level of experience of the operator (high for confident users and low for the unconfident ones). We assume that the level of confidence does not change during the task and, consequently, that $\lambda_h$ stays constant during the collaboration.

In this work, we aim at designing a safety kinodynamic architecture that:

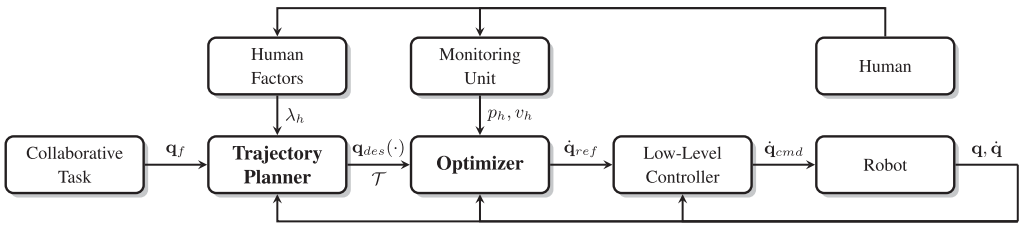


Fig. 1. Proposed overall framework.

- Computes a nominal trajectory that is collision-free, i.e. a trajectory that the robot could ideally execute at maximum speed and that avoids all the surrounding obstacles.
- Starting from the nominal trajectory, it automatically builds a set of safety tubes where the robot could stay without colliding with the environment, while maintaining a predictable behaviour.
- Exploiting the tracking of the human movements, it computes online the optimal safety robot inputs that aim at following the nominal trajectory while ensuring that the robot behaviour is compliant with the safety standards.

## III. Architecture

The proposed safety collaborative strategy is summarized in Fig. 1, where two main components may be distinguished:

- The **Trajectory Planner** block, which is responsible of planning the collision-free trajectory $\mathbf{q}_{des}(\cdot)$ that allows the robot to reach the desired final configuration $\mathbf{q}_f$.
- The **Optimizer** block solves online a Mixed-Integer Linear Programming (MILP) problem to compute the optimal input $\dot{\mathbf{q}}_{ref}$ that allows the robot to follow the planned trajectory while being compliant with the safety standards.

The overall procedure starts with the assignment of the collaborative task to be executed, which can be implemented with several procedures already available in literature [25], [26], [27], [28]. The task is associated with a desired final configuration $\mathbf{q}_f$ that is exploited by the **Trajectory Planner** to compute the collision-free trajectory $\mathbf{q}_{des}(\cdot)$. Moreover, it leverages the maximum admissible deviation $\lambda_h$ to build a set of tubes around each link trajectory $\mathcal{T}$. To this purpose, the **Trajectory Planner** must be aware of all the static obstacles that are inside the collaborative workspace. In this phase, the human operator is ignored since is a very dynamic obstacle.

Then, the **Optimizer** exploits the human monitoring information to compute online the optimal input $\dot{\mathbf{q}}_{ref}$. This optimization problem ensures that the implemented behaviour is compliant with the ISO/TS 15066 and that the resulting Cartesian motion lies inside the tubes $\mathcal{T}$.

Lastly, the input is forwarded to the low-level controller that takes care of implementing the desired velocity.

### A. Trajectory Planner

The role of this component is to find a robot trajectory $\mathbf{q}(\cdot)$ that is collision-free and that the robot could ideally execute at maximum speed. Since the human operator is in a certain sense an unpredictable obstacle, in this first planning phase it is not considered, and the trajectory is calculated only on the static collaborative workspace.

**Algorithm 1:** TrajcetoryPlanner().

1: **Require:** $\mathbf{q}_i, \mathbf{q}_f, \lambda_h$
2: $\bar{\mathbf{q}}_{des}(\cdot) \leftarrow \mathbf{plan}(\mathbf{q}_i, \mathbf{q}_f)$
3: $\mathbf{q}_{des}(\cdot) \leftarrow \mathbf{generateTrajectory}(\bar{\mathbf{q}}_{des}(\cdot))$
4: $\mathcal{T} \leftarrow \mathbf{generateTubes}(\mathbf{q}_{des}(\cdot), \lambda_h)$
5: $\mathbf{send}(\mathbf{q}_{des}(\cdot), \mathcal{T})$

The trajectory planning is implemented according to the pseudo-code reported in Algorithm 1.

The trajectory planner needs as input the initial and the final configuration, respectively $\mathbf{q}_i$ and $\mathbf{q}_f$, and the maximum admissible path deviation $\lambda_h$ (Line 1). It firstly plans the collision-free path $\bar{\mathbf{q}}_{des}(\cdot)$ that the robot should perform (Line 2). The function **plan** can be implemented using different strategies already available for robotic applications, e.g. [29], [30], [31]. Once the path has been computed, the planner defines a trajectory that follow the desired path and which is compliant with the robot limits, i.e. maximum joint velocity, maximum torque. To this purpose it is possible to exploit the strategy presented in [32], which allows to obtain the maximum speed trajectory (Line 3). Subsequently, it computes the set of tubes $\mathcal{T}$ such that: in each tube the $i$-th link can freely move without colliding and whose radius is, at most, equal to $\lambda_h$. Lastly, the trajectory and the set of tubes are forwarded to the optimizer block (Line 5).

### B. Optimizer

Once the nominal trajectory $\mathbf{q}_{des}(\cdot)$ and the set of tubes $\mathcal{T}$ have been computed, the optimizer generates a safe and predictable reference trajectory to be set as a setpoint to track to the robot.

First, a mathematical description of the set of tubes $\mathcal{T}$ needs to be computed in order to synthesize the controller. The idea is to fit the tube with a set of simpler volumes, such as spheres or cubes, and to constrain the Cartesian links positions inside this set of volumes. To keep the computational cost low, we chose to use cubes, as they allow a linear formulation and, therefore, to set up a linear optimization problem

Fig. 2 reports a simplified example of how this step is performed on a 2-joints planar manipulator. Considering Fig. 2 left, the Trajectory planner provides first the desired trajectory (light blue dots) and the set of tubes (green areas). The set of tube is computed as collision free considering the position of the obstacle (dark red area). Then, for each link, a set of cubes is located inside the original tube (see Fig. 2 right), ensuring a minimum of overlap to guarantee communication between the cubes, and contain the initial and final position. This constraint, with (3) and the limits of the robot are finally used to summarize

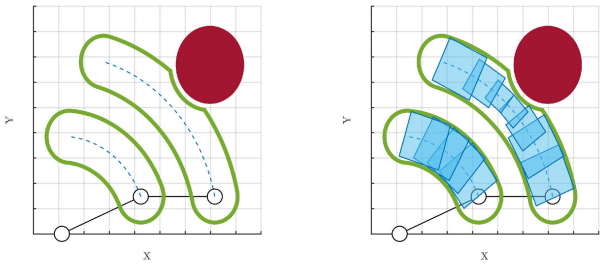

Fig. 2. Visual representation of the proposed predictability constraint in 2D. On the left the constructions of the tubes around the paths. On the right the approximation as a set of squares.

an optimization problem, the solution of which will be used as a reference to control the robot.

Since this component works as a dynamic planner, it considers the robot as perfectly kinematic initializing a state $\mathbf{q}^v$ as the state of the robot $\mathbf{q}$ at the first control cycle, and updating it using the model 1.

The procedure starts by constraining the Cartesian position of the generic $i$-th link to lie inside the $j$-th cube of its trajectory. The choice behind the use of cubes is related to possibility to express these constraints in a linear form. This can be done by constraining the position to lie in the volume of space enclosed within the 6 planes defining the faces of the cube.

Let $c_i^j = [{}^x c_i^j, {}^y c_i^j, {}^z c_i^j]^T \in \mathbb{R}^3$, $l_i^j \in \mathbb{R}$, with $l_i^j \leq \lambda_h$, and $Rc_i^j \in \mathbb{R}^{3\times3}$ be the centre, the edge size, and the rotation matrix of the cube, respectively. The $i$-th link position $x_i^v \in \mathbb{R}^3$ lies inside of that cube if:

$$\bar{A}_i^j x_i^v \leq \bar{b}_i^j, \tag{5}$$

with $\bar{A}_i^j = \text{diag}\{[-1,1]^T, [-1,1]^T, [-1,1]^T\}$ and $\tilde{b}_i^j = [l_i^j/2, l_i^j/2, l_i^j/2, l_i^j/2, l_i^j/2, l_i^j/2]^T$ representing the matrices grouping the coefficients of the 6 planes.

Since the robot is considered as perfectly kinematic, (1) holds and it is possible to assume that the position of the $i$-th link evolves as a single integrator in the discrete time domain as:

$$\mathbf{x}_i^v(k+1) = \mathbf{x}_i^v(k) + BJ_i^p(\mathbf{q}^v)\mathbf{u}^v(k), \tag{6}$$

where $B = \text{diag}\{\Delta t_c\} \in \mathbb{R}^{3\times3}$ is the input matrix, with $\Delta t_c$ the sampling time ($t = k\Delta t_c$, $k \in \mathbb{Z}$), $J_i^p(\mathbf{q}^v) \in \mathbb{R}^{3\times n}$ is the position part of the $i$-th link Jacobian, and $\mathbf{u}^v \in \mathbb{R}^n$ represents the joint velocity input.

Using (6) in (5), the position of the $i$-th link of the robot can be constrained to lie inside a cube by constraining its future position and choosing the input velocities $\mathbf{u}^v$ such that:

$$A_i^j \mathbf{u}^v(k) \leq b_i^j, \tag{7}$$

where:

$$A_i^j = \bar{A}_i^j BJ_i^p(\mathbf{q}^v), \quad b_i^j = \bar{b}_i^j - A_i^j x_i(k). \tag{8}$$

The position of each link must lie inside the set made up by the union of all cubes.

To this aim, it is necessary to expand the constraint (7) to all the cubes of the trajectory. This cannot be done by simply imposing (7) for all cubes. This in fact will turn to an AND type of constraint, requiring that the robot is at the same time in all cubes, which is clearly impossible. The overall constraint needs to be converted into an OR type constraint, i.e. requiring that the robot stays in one cube at a time. This can be done by (see [33] for more details):

1) Designing a constraint like (7) for each cube.
2) Adding for each cube a binary extra variable representing if the $i$-th link belongs to that cube.
3) Adding a constraint on the extra binary variables to force the $i$-th link to belong to a single cube at a time.

Formally, by setting for each link:

$$A_i^j \mathbf{u}^v(k) - M\epsilon_i^j \leq b_i^j \quad j = 1, \ldots, N_q, \tag{9}$$

and

$$\sum_{j=1}^{N_q} \epsilon_i^j \leq N_q - 1, \tag{10}$$

where $M \in \mathbb{R}^6$ is a large arbitrary positive vector, $\epsilon_i^j \in \{0,1\}$ is the extra binary variable, and $N_q \in \mathbb{N}$ the number of cubes of each trajectory. With this approach, if the $j$-th constraint of the $i$-th link is not satisfied, the corresponding binary variable $\epsilon_i^j$ is equal to 1. The constraint (10) ensures that at least one of the constraints (9) is satisfied, which represents the cube where the $i$-th link lies in. It is worth noting that the formulation of the constraint is still linear.

Finally, the input is computed solving the following optimization problem:

$$\begin{aligned}
\min_{\mathbf{u}^v, \epsilon} \quad & \|\dot{\mathbf{q}}_{des}^v(\cdot) - \mathbf{u}^v\|^2 \\
s.t. \quad & A_i^j \mathbf{u}^v - M\epsilon_i^j \leq b_i^j \quad j = 1, \ldots, N_q \\
& \sum_{j=1}^{N_q} \epsilon_i^j \leq N_q - 1 \\
& \dot{\mathbf{q}}_{min} < \mathbf{u}^v < \dot{\mathbf{q}}_{\max} \\
& J_{rh}^i \mathbf{u}^v < \beta_{rh}^i \\
& i = 1, \ldots, n,
\end{aligned} \tag{11}$$

where $\dot{\mathbf{q}}_{\max} \in \mathbb{R}^n$ and $\dot{\mathbf{q}}_{min} \in \mathbb{R}^n$ represents the maximum and minimum joint velocity of the robot, respectively, $J_{rh}^i = n_{rh,i}^T J_i \in \mathbb{R}^{1\times n}$ is the Jacobian of the $i$-th link towards the human, where $n_{rh,i}$ is the unit vector representing the direction from the $i$-th link to the human operator. This vector can be computed using standard techniques already available, such as distance between spheres [13]. $\beta_{rh}^i$ is the velocity limit of the $i$-th link computed as in (3) and:

$$\dot{\mathbf{q}}_{des}^v = \dot{\mathbf{q}}_{des} + K^v(\mathbf{q}^v - \mathbf{q}_{des}) + D^v(\dot{\mathbf{q}}^v - \dot{\mathbf{q}}_{des}), \tag{12}$$

where $K^v \in \mathbb{R}^{n\times n}$ and $D^v \in \mathbb{R}^{n\times n}$ are the proportional and derivative gains, respectively, introduced to allow the system to be attracted to the desired trajectory after a deviation induced by the optimizer, e.g. to avoid the human.

At each control cycle, the state of the robot $\mathbf{q}^v$ is updated using the model (1), and the reference speed $\dot{\mathbf{q}}_{ref}$ is set equal to $\mathbf{u}^v$ returned by the optimizer.

## IV. Experiments

The proposed framework has been experimentally validated in two distinct scenarios, thereby showcasing its efficacy and generality. The first scenario encompasses a collaborative mobile

robot, comprising an UR10e and a Neobotix MPO-500, while the second scenario involves a collaborative manipulator, specifically the UR5e model. In both instances, the robot's objective is to move from its current configuration to a desired configuration, while adhering to the safety constraints outlined by ISO/TS 15066. Comparative analysis against the approach presented in [14] reveals better performance across both scenarios.

In order to track the human operator[1] and detect obstacles within the environment, we employed seven OptiTrack Prime$^X$ cameras along with the Motive software. These tracking components were integrated into the system, which was developed using ROS Kinetic Kame meta-operating system and executed on a Intel(R) Core(TM) i7-7700HQ with Ubuntu 16.04 operating system. The trajectory planner layer is based on the RRT-Connect algorithm [34] and it is implemented exploiting OMPL and FCL libraries, while the MILP problem in (11) is solved online exploiting the Gurobi solver.

Concerning the frequencies, the communication with both UR10e and UR5e works at 500 Hz, while the one with the Neobotix MPO-500 works at 50 Hz. The OptiTrack, instead, works at a frequency of 240 Hz.

Solving the optimization problem as in (11) considering (9) and (10) for all the links may be computationally demanding. For this reason we choose to apply (9) and (10) to 4 links only, which are the ones reported in Figs. 5 and 10. With 4 links, the optimization problem it is solved in approximately 2.5 ms.

Moreover, to further reduce the computational load, (9) and (10) were applied only to the subset of cubes located in close proximity to the current position of each link. To compute such subset it has been ensured that the future position of each link belongs to at least one cube in case it moves at maximum speed towards or in the opposite direction of the trajectory.

The number of cubes $N_q$ used to fit the tube of each link is calculated by initially positioning a cube at intervals of $0.25\lambda_h$ along its trajectory. Adjustments are made to reduce sparsity as necessary, ensuring proper overlap between consecutive cubes.

Lastly, the gain matrices $K^v$ is equal to identity matrix, while $D^v$ is a zero matrix.

In order to validate the efficacy of the proposed framework, a series of experiments were conducted in each scenario. These experiments aimed to evaluate the performance under different conditions, considering both a *confident* human operator who permits significant deviations from the planned path, and an *unconfident* human operator who lacks trust in the robot and allows minimal deviations. The outcomes of these experiments, along with supplementary material, provide a comprehensive understanding of the results obtained.

### A. Mobile Collaborative Robot

In this scenario, the UR10e robot is positioned atop the Neobotix MPO-500 mobile platform. The robot's kinematic structure corresponds of 7 links, 6 for the UR10e and 1 that embeds the entire mobile base. Regarding control, it becomes necessary to account for the increased degrees of freedom introduced by the mobile base, thereby the Jacobian matrix exploited in the optimization problem (11) is the augmented Jacobian of the mobile manipulator [35]:

$$J(\mathbf{q}) = \begin{bmatrix} J_b(\mathbf{q}_b) & J_m(\mathbf{q}_m) \end{bmatrix}, \tag{13}$$

[1]For the sake of simplicity, the human operator is approximated only with a capsule embedding the arm.

Fig. 3. Setup of the experiments with two obstacles (yellow circles) and a wrist band tracking the human operator (red circle).

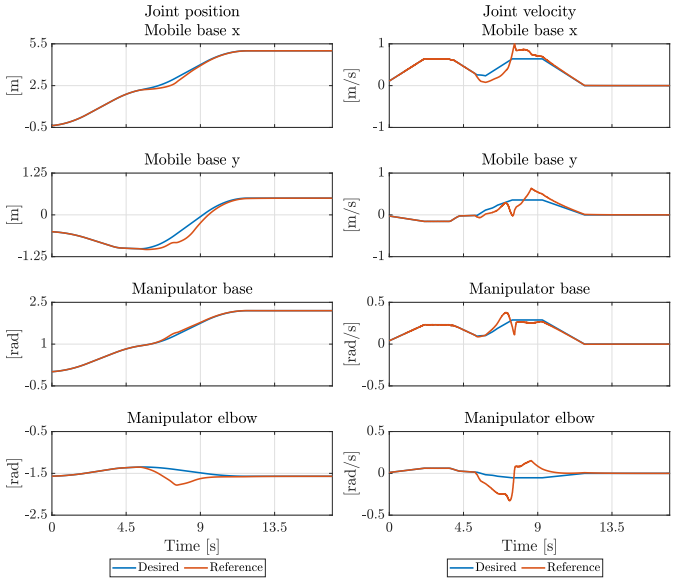


Fig. 4. Joint positions and velocities in a scenario with a confident human operator.

where $J_b(\mathbf{q}) \in \mathbb{R}^{6\times3}$ and $J_m(\mathbf{m}) \in \mathbb{R}^{6\times6}$ are the Jacobian of the mobile base and manipulator, respectively.

The complete setup is illustrated in Fig. 3(left).

In the first experiment, the robot has to reach a desired final configuration avoiding the obstacles while collaborating with confident human operator, i.e. large cubes size. The planner successfully plans a trajectory and the robot starts to follow it. At $4.5 \leq t \leq 9.5$, the human operator approaches the robot causing a drop of the velocity, see (3), and making the planned path inefficient. Since the operator is classified as confident, $\lambda_h$ is big and the optimizer is capable of computing a set of inputs that deviate from the desired path to obtain a better trajectory, while ensuring the safety. Fig. 4 shows the evolution of the joints that reports the most interesting behaviour, while Fig. 5 demonstrates that the respective links fulfill the constraints, i.e. they remains inside the tube (set of cubes) and the speed is compliant with (3).

An additional set of experiments was conducted involving an unconfident human operator, characterized by a smaller tolerance for deviations from the planned path. In this scenario, the robot was constrained to maintain close proximity to the intended trajectory. Unlike previous experiments, when the human approached the robot, the only permissible solution was to stop until the human operator moved away from the robot. The details are reported in Fig. 6.

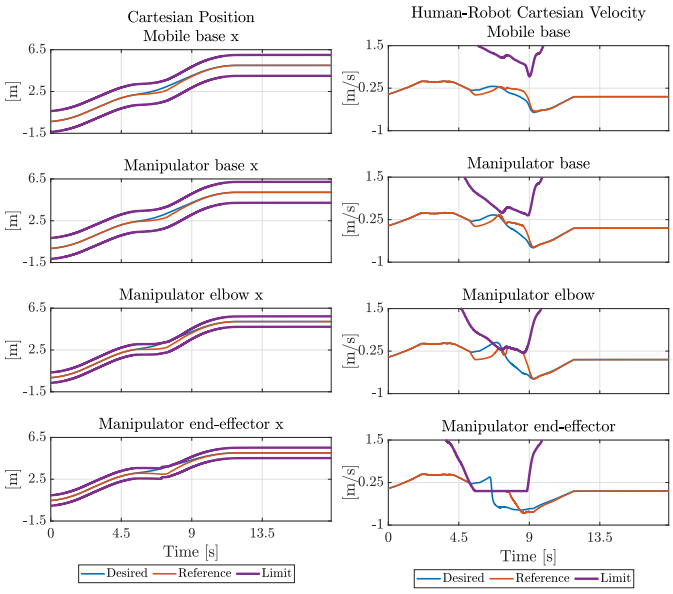


Fig. 5. Constraints in a scenario with a confident human operator. On the left, the links Cartesian position and their respective upper and lower bounds due to the cubes. On the right, the links velocity towards the human operator with the velocity limit imposed by the ISO/TS 15066.

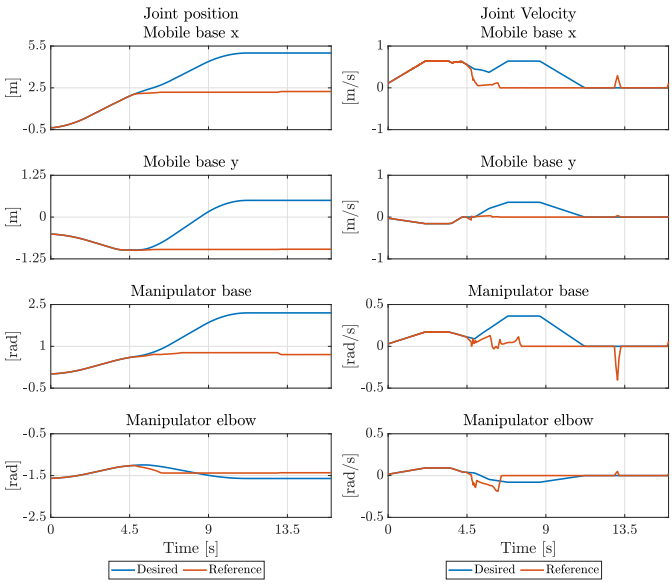


Fig. 6. Joint positions and velocities in a scenario with an unconfident human operator. When the human operator approaches the robot, the optimizer commands a zero velocity since $\lambda_h$ is low.

To better understand the effectiveness and the performances of the proposed solution, the proposed approach is compared with the framework developed in [14] has been performed. To have a more reliable comparison the experiment has been performed under these conditions:

- [14] has been extended to work also with the mobile base.
- The obstacles have been removed.
- The human operator is considered confident, in order to allow changing in the path as [14].

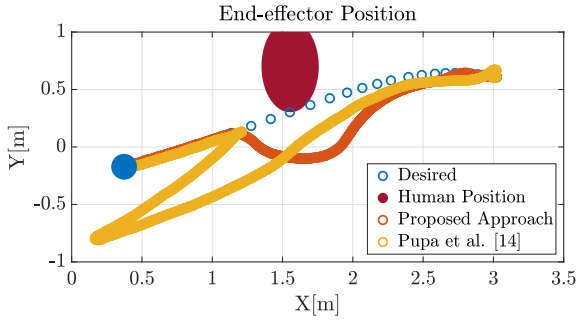


Fig. 7. Comparison between the cartesian position of the end-effector with the two approaches.

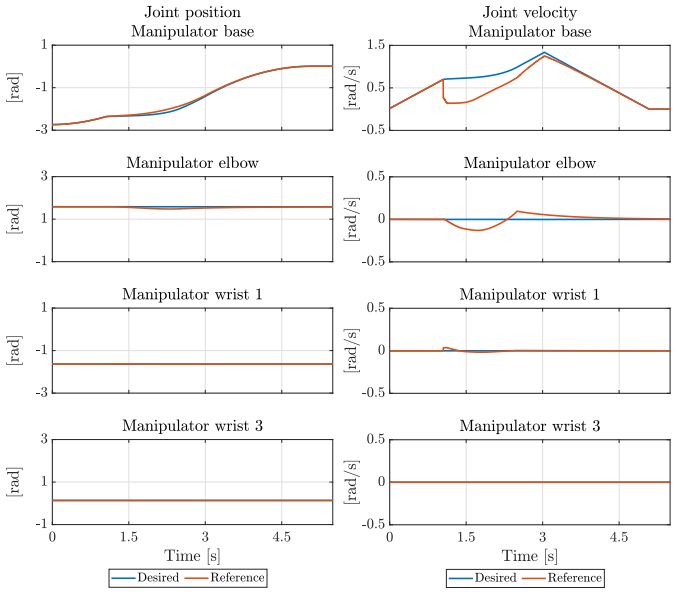


Fig. 8. Joint positions and velocities in a scenario with a confident human operator.

TABLE I
QUANTITATIVE ANALYSIS OF THE RESULTS

| | Mobile Base | | Manipulator | |
|---|---|---|---|---|
| | Length $[m]$ | Time $[s]$ | Length $[m]$ | Time $[s]$ |
| *Pupa et al. [14]* | 6.63 | 19.79 | 1.32 | 10.20 |
| *Proposed Work* | 3.83 | 12.66 | 1.11 | 5.14 |
| **Improvement** | 42.23% | 36.03% | 15.91% | 49.41% |

Fig. 7 shows the resulting end-effector trajectories achieved with the two strategies, while Table I presents the corresponding quantitative results. It is noteworthy that the proposed framework outperforms the approach presented in [14], exhibiting better performance in terms of both path length reduction and decreased execution time.

It is important to highlight that the proposed approach represents a generalization of the work presented in [14]. Specifically, when $\lambda_h = 0$, the proposed approach effectively reduces to [14], where replanning is not incorporated. By employing the architecture introduced in this study, it is possible to achieve the same level of safety as in [14], while simultaneously offering

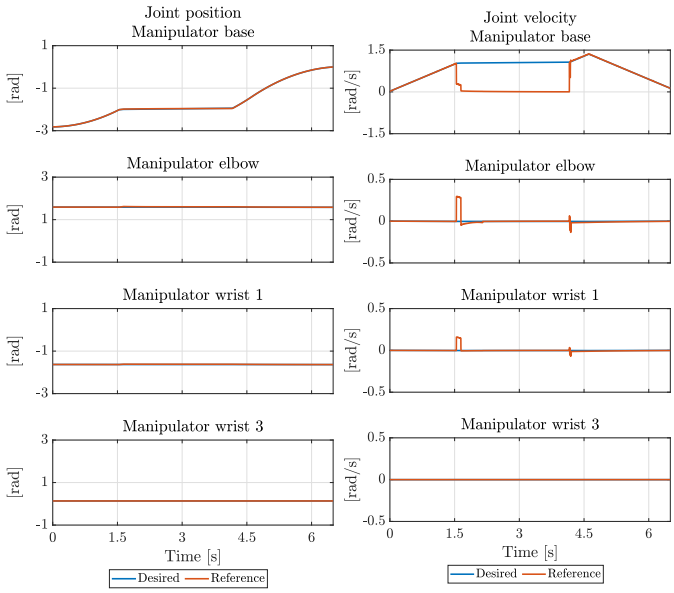


Fig. 9. Joint positions and velocities in a scenario with an unconfident human operator. When the human operator approaches the robot, the optimizer commands a zero velocity until the human operator moves away.

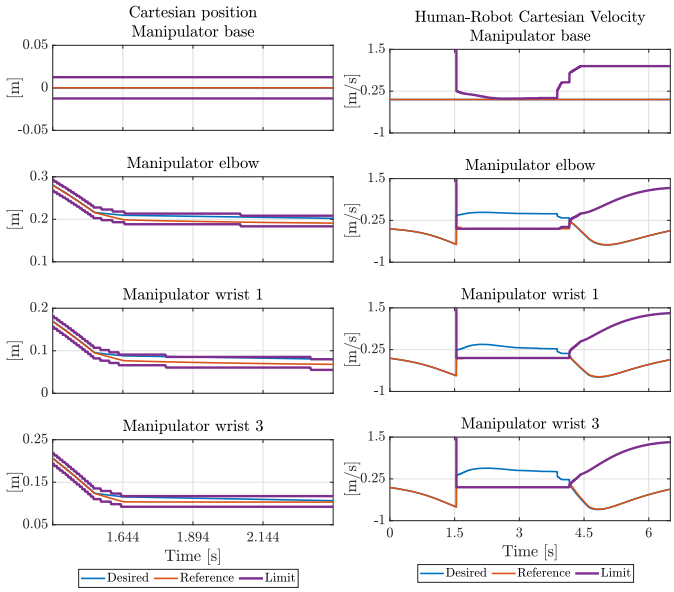


Fig. 10. Constraints in a scenario with an unconfident human operator.

enhanced flexibility to accommodate varying degrees of human confidence.

### B. Collaborative Manipulator

To enhance and further validate the framework, the identical set of experiments was conducted in a scenario involving solely the collaborative manipulator UR5e, which is illustrated in Fig. 3(right). From the control point of view, instead, the Jacobian of the robot is equal to the one of the solely manipulator:

$$J(\mathbf{q}) = J_m(\mathbf{q}_m), \tag{14}$$

The evolution of the manipulator joints when approaching a confident human operator is depicted in Fig. 8. Additionally, Figs. 9 and 10 provide detailed insights into the experiments involving an unconfident human operator. It should be noted that due to the small size of the cube, the validity of the constraint is demonstrated only for a portion of the experiment.

Regarding the comparison with [14], instead, the results are reported in Table I, which demonstrates that also in this case the proposed approach performs better.

## V. Conclusions and Future Works

In this letter, a novel framework for safe human-robot collaboration has been proposed. Firstly, the trajectory planner is responsible of computing a collision-free trajectory for the robot. Subsequently, exploiting the human characteristics and preferences and investigating the environment, a set of tubes are build around the Cartesian paths of each robot link. The size of this tubes depends on how much deviation from the original path the human operator may accept. These tubes are then approximated with a set of cubes, that can be expressed with a proper set of convex constraints. Then, an optimization problem is solved online to achieve an optimal behaviour while ensuring both to be compliant the safety imposed by the ISO/TS 15066 and to stay inside the set of cubes. The results show that the framework is capable of achieving optimal behaviour, outperforming the work proposed in [14].

Future works aim at improving the human factors block, extending the optimization problem to allow handling an online variation of $\lambda_h$. Such online computation of $\lambda_h$ can be associated to the evolution of human stress and emotions during the collaboration [36], [37]. Next, it would be better to approximate the tubes with a series of time-varying constraints, such as Control Barrier Functions, to further integrate the characteristics of the human operator. Finally, the method for assessing predictability can be enhanced to create a more comprehensive metric, which can also be validated through an extensive user study.

## References

[1] V. Villani, F. Pini, F. Leali, and C. Secchi, "Survey on human–robot collaboration in industrial settings: Safety, intuitive interfaces and applications," *Mechatronics*, vol. 55, pp. 248–266, 2018.

[2] *Robots and Robotic Devices–Safety Requirements for Industrial Robots–Part 1: Robots*, Rec. ISO 10218-1:2011(E), International Organization for Standardization, Geneva, Switzerland, vol. 2011, Jul. 2011. [Online]. Available: https://www.iso.org/standard/51330.html

[3] *Robots and Robotic Devices–Safety Requirements for Industrial Robots–Part 2: Robot Systems and Integration*, Rec. ISO 10218-2:2011(E), International Organization for Standardization, Geneva, Switzerland, vol. 2011, Jul. 2011. [Online]. Available: https://www.iso.org/standard/41571.html

[4] *Robots and Robotic Devices–Collaborative Robots*, Rec. ISO/TS 15066:2016(E), International Organization for Standardization Technical Specification, Geneva, Switzerland, vol. 2016, Feb. 2016. [Online]. Available: https://www.iso.org/standard/62996.html

[5] M. Ragaglia, A. M. Zanchettin, and P. Rocco, "Safety-aware trajectory scaling for human-robot collaboration with prediction of human occupancy," in *Proc. IEEE Int. Conf. Adv. Robot.*, 2015, pp. 85–90.

[6] M. Lippi and A. Marino, "Human multi-robot safe interaction: A trajectory scaling approach based on safety assessment," *IEEE Trans. Control Syst. Technol.*, vol. 29, no. 4, pp. 1565–1580, Jul. 2021.

[7] A. Levratti, G. Riggio, C. Fantuzzi, A. De Vuono, and C. Secchi, "Tirebot: A collaborative robot for the tire workshop," *Robot. Comput.- Integr. Manuf.*, vol. 57, pp. 129–137, 2019.

[8] J.-H. Chen and K.-T. Song, "Collision-free motion planning for human-robot collaborative safety under Cartesian constraint," in

*Proc. IEEE Int. Conf. Robot. Automat.*, 2018, pp. 4348–4354, doi: 10.1109/ICRA.2018.8460185.

[9] F. Ferraguti, N. Preda, M. Bonfe, and C. Secchi, "Bilateral teleoperation of a dual arms surgical robot with passive virtual fixtures generation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2015, pp. 4223–4228.

[10] K. Merckaert, B. Convens, C. ju Wu, A. Roncone, M. M. Nicotra, and B. Vanderborght, "Real-time motion control of robotic manipulators for safe human–robot coexistence," *Robot. Comput.- Integr. Manuf.*, vol. 73, 2022, Art. no. 102223. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0736584521001022

[11] H.-C. Lin, C. Liu, Y. Fan, and M. Tomizuka, "Real-time collision avoidance algorithm on industrial manipulators," in *Proc. IEEE Conf. Control Technol. Appl.*, 2017, pp. 1294–1299.

[12] F. Ferraguti, M. Bertuletti, C. T. Landi, M. Bonfè, C. Fantuzzi, and C. Secchi, "A control barrier function approach for maximizing performance while fulfilling to ISO/TS 15066 regulations," *IEEE Robot. Automat. Lett.*, vol. 5, no. 4, pp. 5921–5928, Oct. 2020.

[13] F. Ferraguti et al., "Safety and efficiency in robotics: The control barrier functions approach," *IEEE Robot. Automat. Mag.*, vol. 29, no. 3, pp. 139–151, Sep. 2022.

[14] A. Pupa, M. Arrfou, G. Andreoni, and C. Secchi, "A safety-aware kinodynamic architecture for human-robot collaboration," *IEEE Robot. Automat. Lett.*, vol. 6, no. 3, pp. 4465–4471, Jul. 2021.

[15] R. J. Kirschner, H. Mayer, L. Burr, N. Mansfeld, S. Abdolshah, and S. Haddadin, "Expectable motion unit: Avoiding hazards from human involuntary motions in human-robot interaction," *IEEE Robot. Automat. Lett.*, vol. 7, no. 2, pp. 2993–3000, Apr. 2022.

[16] M. Eckhoff, R. J. Kirschner, E. Kern, S. Abdolshah, and S. Haddadin, "An MPC framework for planning safe & trustworthy robot motions," in *Proc. Int. Conf. Robot. Automat.*, 2022, pp. 4737–4742.

[17] S. J. Baltrusch, F. Krause, A. W. de Vries, W. van Dijk, and M. P. de Looze, "What about the human in human robot collaboration?," *Ergonomics*, vol. 65, no. 5, pp. 719–740, 2022, doi: 10.1080/00140139.2021.1984585.

[18] M. Koppenborg, P. Nickel, B. Naber, A. Lungfiel, and M. Huelke, "Effects of movement speed and predictability in human–robot collaboration," *Hum. Factors Ergonom. Manuf. Serv. Ind.*, vol. 27, no. 4, pp. 197–209, 2017.

[19] X. Ren, Z. Li, M. Zhou, and Y. Hu, "Human intention-aware motion planning and adaptive fuzzy control for a collaborative robot with flexible joints," *IEEE Trans. Fuzzy Syst.*, vol. 31, no. 7, pp. 2375–2388, Jul. 2023.

[20] B. Sadrfaridpour and Y. Wang, "Collaborative assembly in hybrid manufacturing cells: An integrated framework for human–robot interaction," *IEEE Trans. Automat. Sci. Eng.*, vol. 15, no. 3, pp. 1178–1192, Jul. 2018.

[21] S. Moon, Y. Park, D. W. Ko, and I. H. Suh, "Multiple kinect sensor fusion for human skeleton tracking using Kalman filtering," *Int. J. Adv. Robot. Syst.*, vol. 13, no. 2, pp. 65–74, 2016.

[22] J. Kofman, X. Wu, T. J. Luu, and S. Verma, "Teleoperation of a robot manipulator using a vision-based human-robot interface," *IEEE Trans. Ind. Electron.*, vol. 52, no. 5, pp. 1206–1219, Oct. 2005.

[23] J. Fan, W. Xu, Y. Wu, and Y. Gong, "Human tracking using convolutional neural networks," *IEEE Trans. Neural Netw.*, vol. 21, no. 10, pp. 1610–1623, Oct. 2010.

[24] S. M. Rahman and Y. Wang, "Mutual trust-based subtask allocation for human–robot collaboration in flexible lightweight assembly in manufacturing," *Mechatronics*, vol. 54, pp. 94–109, 2018.

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

[26] A. Raatz, S. Blankemeyer, T. Recker, D. Pischke, and P. Nyhuis, "Task scheduling method for HRC workplaces based on capabilities and execution time assumptions for robots," *CIRP Ann.*, vol. 69, no. 1, pp. 13–16, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0007850620300524

[27] A. Pupa, W. Van Dijk, and C. Secchi, "A human-centered dynamic scheduling architecture for collaborative application," *IEEE Robot. Automat. Lett.*, vol. 6, no. 3, pp. 4736–4743, Jul. 2021.

[28] A. Pupa, W. Van Dijk, C. Brekelmans, and C. Secchi, "A resilient and effective task scheduling approach for industrial human-robot collaboration," *Sensors*, vol. 22, no. 13, 2022, Art. no. 4901. [Online]. Available: https://www.mdpi.com/1424-8220/22/13/4901

[29] S. LaValle et al., "Rapidly-exploring random trees: Progress and prospects," *Algorithmic Comput. Robot: New Directions*, vol. 5, pp. 293–308, 2001.

[30] L. Jaillet and T. Siméon, "A PRM-based motion planner for dynamically changing environments," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2004, pp. 1606–1611.

[31] N. Ratliff, M. Zucker, J. A. Bagnell, and S. Srinivasa, "CHOMP: Gradient optimization techniques for efficient motion planning," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2009, pp. 489–494.

[32] T. Kunz and M. Stilman, "Time-optimal trajectory generation for path following with bounded acceleration and velocity," in *Proc. Robot.: Sci. Syst.*, Sydney, Australia, Jul. 2012, doi: 10.15607/RSS.2012.VIII.027.

[33] T. Schouwenaars, B. De Moor, E. Feron, and J. How, "Mixed integer programming for multi-vehicle path planning," in *Proc. IEEE Eur. Control Conf.*, 2001, pp. 2603–2608.

[34] J. J. Kuffner and S. M. LaValle, "RRT-connect: An efficient approach to single-query path planning," in *Proc. Millennium Conf. IEEE Int. Conf. Robot. Automat. Symp. Proc. (Cat. No 00CH37065)*, 2000, pp. 995–1001.

[35] A. Pupa, F. Breveglieri, and C. Secchi, "An optimal human-based control approach for mobile human-robot collaboration," in *Proc. Hum.-Friendly Robot.: HFR: 15th Int. Workshop Human-Friendly Robot.*, 2023, pp. 30–44.

[36] M. Lagomarsino, M. Lorenzini, E. De Momi, and A. Ajoudani, "An online framework for cognitive load assessment in industrial tasks," *Robot. Comput.- Integr. Manuf.*, vol. 78, 2022, Art. no. 102380. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0736584522000679

[37] V. Villani, M. Righi, L. Sabattini, and C. Secchi, "Wearable devices for the assessment of cognitive effort for human–robot interaction," *IEEE Sensors J.*, vol. 20, no. 21, pp. 13047–13056, Nov. 2020.