

---

# On-line collision avoidance for collaborative robot manipulators by adjusting off-line generated paths: an industrial use case

Mohammad Safeea<sup>1</sup>, Pedro Neto<sup>2,\*</sup>, Richard Bearee<sup>1</sup>

**1 University of Coimbra, Department of Mechanical Engineering, 3030-788 Coimbra, Portugal**

**2 Arts et Métiers, LISPEN, 59800 Lille, France**

\* [pedro.neto@dem.uc.pt](mailto:pedro.neto@dem.uc.pt)

## Abstract

Human-robot collision avoidance is a key in collaborative robotics and in the framework of Industry 4.0. It plays an important role for achieving safety criteria while having humans and machines working side-by-side in unstructured and time-varying environment. This study introduces the subject of manipulator's on-line collision avoidance into a real industrial application implementing typical sensors and a commonly used collaborative industrial manipulator, KUKA iiwa. In the proposed methodology, the human co-worker and the robot are represented by geometric primitives (capsules). The minimum distance and relative velocity between them is calculated, when human/obstacles are nearby the concept of hypothetical repulsion and attraction vectors is used. By coupling this concept with a mathematical representation of robot's kinematics, a task level control with collision avoidance capability is achieved. Consequently, the off-line generated nominal path of the industrial task is modified on-the-fly so the robot is able to avoid collision with the co-worker safely while being able to fulfill the industrial operation. To guarantee motion continuity when switching between different tasks, the notion of repulsion-vector-reshaping is introduced. Tests on an assembly robotic cell in automotive industry show that the robot moves smoothly and avoids collisions successfully by adjusting the off-line generated nominal paths.

## 1 Introduction

Industrial robots are traditionally working inside fences, isolated from humans. The ability to have robots sharing the workspace and working side-by-side with human co-workers is a key factor for the materialization of the Industry 4.0 concept. The paradigm for robot usage has changed in the last few years, from an idea in which robots work with complete autonomy to a scenario where robots cognitively collaborate with human beings. This brings together the best of each partner, robot and human, by combining coordination, dexterity and cognitive capabilities of humans with the robots' accuracy, agility and ability to produce repetitive work. For example, robots can help humans in carrying and manipulating sensitive/heavy objects safely [35] and positioning them precisely by hand-guiding [27]. In this scenario the robot can play the role of a force magnifier while moving compliantly according to the haptic feedback from the human.

---

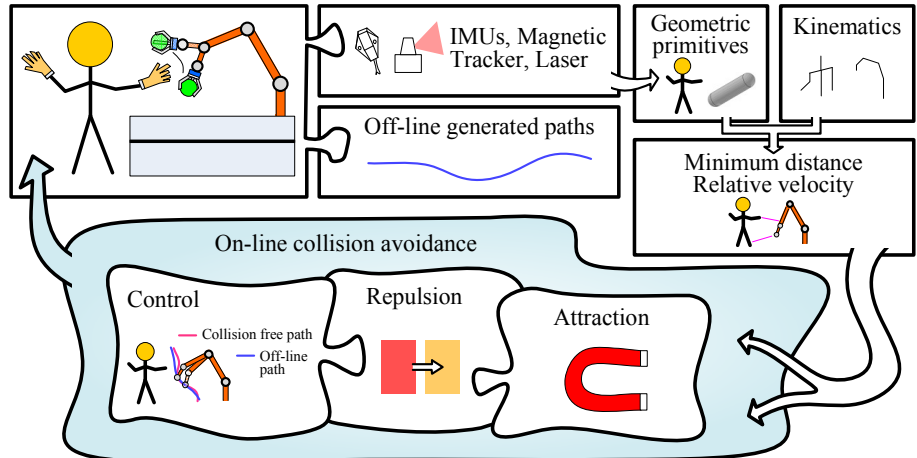
Reaching the goal of developing/creating safe collaborative robots will allow a greater presence of robots in our society, with a positive impact in several domains, including industry [20]. Nowadays, industrial collaborative robots, which are not operating inside fences, do not have autonomy to perceive its unstructured and time-varying surrounding environment, nor the ability to avoid collisions with human co-workers in real-time while keeping the task target defined by the off-line generated paths. On the contrary, they stop when a predefined minimum separation distance is reached. Due to this issue, the full potentialities of collaborative robots in industrial environment are not totally explored. The increasing demand by industry for collaborative robot-based solutions makes the need for advanced collision avoidance strategies more visible. To have them working safely alongside with humans, robots need to be provided with biological-like reflexes, allowing them to circumvent obstacles and avoid collisions. This is extremely important in order to give robots more autonomy and minimum need for human intervention, especially when robots are operating in dynamic environment and interacting/collaborating with human co-workers [21]. The requirements for safe collaborative robots, including physical human-robot interaction (pHRI), are detailed in [9], where collision avoidance is listed as a factor, among others, which is important for human-robot safety. The new standard ISO 10218 and the technical specification TS 15066 define the safety requirements for collaborative robots [26]. Apart from industrial domain and human-robot collaboration, collision avoidance is also being investigated for aerospace applications, including robotic arms mounted on space maneuverable platforms [4] and aerial manipulators mounted on drones [12].

For a proper implementation of collision avoidance in Human-Robot Interaction (HRI) scenario, the motion of the human co-worker shall be predicted, and his/her configuration shall be captured. For this purpose, researches have utilized various methods and sensors. In [36], the authors presented a method for human pose estimation (in a plane) based on laser range measurements, the method was applied successfully in HRI scenario between a human and a mobile robot. The intelligent space (iSpace) project was introduced in [11], iSpace implements a distributed network of sensors for tracking the motion of humans, the iSpace provides a platform for guiding mobile robots in human centered environment. In [7] the authors used the depth data from a Kinect camera for calculating distances between the human and reference points on the robot. In [14] the authors presented a new method, by using a camera mounted on the EEF (eye-in-hand) or the worker's head, a coordinated motion between a robot and a human was achieved, for future work the authors plan to extend the method for achieving human-robot collision avoidance.

After capturing the motion of the human(s)/obstacle(s), the collision avoidance shall be developed. In the robotics literature various studies have been proposed. In the pioneering work of Khatib [15], a real-time obstacle avoidance approach based on the classical artificial potential field (PF) concept is introduced. In PF-based methods, the robot is in a hypothetical vector field, and its motion is influenced by forces of attraction that guide the robot towards the goal and forces of repulsion that repel it away from obstacles. Subjected to these forces the robot finds its way to the goal while avoiding collisions. Recently, a depth space approach for collision avoidance proposes an improved implementation of the PF method in which an estimation of obstacle's velocity was taken into consideration when computing the repulsion vector [5]. PF-based robot self-collision avoidance has been studied, as well as the development of collision avoidance techniques for redundant robots [8]. A distributed real-time approach to collision avoidance considering not only the robot tool centre point over the objects in the cell, but also the body of the tool mounted on the robot flange is in [6]. A passivity-based control scheme for human-robot safe cooperation is proposed in [37]. A collision free trajectory generating method for a robot operating in a shared workspace

in which a neural network is applied to create the way points required for dynamic obstacles avoidance is proposed in [18]. In [25], the authors presented a method for calculating collision free optimal trajectories for robotic manipulators with static obstacles. The proposed algorithm takes into consideration the maximum limits of jerk, torques, and power for each actuator. Tests have been carried out in simulation in a PUMA 560 robot. In [24], it is presented a collision avoidance algorithm between robotic manipulators and mobile obstacles validated in simulation environment. Using the variation principal, it is proposed a path planner for serial manipulators with high degrees of freedom operating in constrained work spaces where the planner produces monotonically optimal collision free paths [32]. Based on fuzzy rules [23], the authors presented a method for resolving internal joint angles in redundant manipulators. The method allows the EEF to follow the desired path, while the internal motion manifold is used to perform other objectives including collision avoidance with surrounding obstacles.

While artificial PF is inspired by electric field phenomena, other approaches, inspired by electromagnetism (circular fields) were investigated [10,33]. Other researchers have approached robot collision avoidance using optimization techniques [2], by formulating the problem as a graph search using Probabilistic Roadmaps (PRM) [13], and considering dynamic changing environment [17]. However, the number of existing studies dedicated to on-line human-robot collision avoidance for manipulators in industrial setups is very limited, and when existing, results are presented in simulation environment. Some of these studies, especially the ones with more direct industrial application, approach collision avoidance by stopping the robot or reducing its velocity when a human reaches a given distance threshold [31]. An interesting work defines four safety strategies for workspace monitoring and collision detection: the system alerts the operator, stops the robot, moves the robot away, or modifies the robot's trajectory from an approaching operator [19].



**Figure 1.** Proposed framework for on-line human-robot collision avoidance.

In our study, an industrial task is defined by the off-line generated robot paths (task primitives), which can be programmed by a Computer-Aided Design (CAD) software, or by using more sophisticated methods including Programming by Demonstration (PbD) [34]. Consequently, a safe human robot interaction is achieved through real-time modification of the off-line generated paths, Fig. 1. In this scenario of shared workspace, the human co-worker focuses on the collaborative task he/she is performing rather than the potential danger from the robot. Using external sensors (IMUs, a magnetic tracker and a laser scanner) the pose of the human body is captured and

---

approximated by hemisphere-capped cylinders designated in this paper by capsules. The robot is represented by three capsules. The analytical minimum distance between capsules representing the robot and the human(s) is calculated using the method in [30]. The human-robot minimum distance and relative velocity are used as inputs for the proposed collision avoidance controller, where hypothetical attraction and repulsion vectors attract the robot towards the goal/target while repelling it away from obstacles. We also introduce the notion of repulsion-vector-reshaping to avoid control discontinuity. By coupling these concepts with a mathematical representation of robot's kinematics we can achieve a task level control with smooth collision avoidance capability. The proposed framework is tested in three different configurations, including a real use case for assembly in automotive industry using real sensors and a collaborative industrial manipulator. Results indicate that the robot reacts smoothly by modifying its off-line generated paths to avoid collision with the human co-worker. Consequently, our study is the first of its kind (according to our knowledge) that satisfies all the following points combined:

1. In our study a real industrial robot (not experimental) is used for performing a typical industrial task;
2. The proposed method for performing the collision avoidance is tailored for industrial use, by combining an off-line path of the EEF (important for industrial applications) with an on-line reactive collision avoidance (required for dynamic collision avoidance);
3. In our method a human co-worker is moving freely around the robot, the whole configuration of his upper body is captured using real sensors. While, most other studies utilize simulation, the few that approached human-robot collision avoidance in a real scenario utilize an experimental robot, and mostly vision sensing which suffers from occlusion. In addition, other studies focused on the collision avoidance itself, without showing results in a real industrial operation;
4. Unlike other studies, we realized that in a real industrial scenario, the control shall switch between different operation modes (as shown later in Fig. 11 and Algorithm Collision avoidance – automotive sample case), leading to repulsion action discontinuity. We solved this issue by proposing the repulsion-vector-reshaping (described in Algorithm Modified repulsion vector).

## 2 Challenges and problem Specification

Two major problems in on-line human-robot collision avoidance can be identified. The first is related with the reliable acquisition of the human pose in unstructured environments. The second is due to the difficulty in achieving smooth continuous robot motion while generating collision avoidance paths. For capturing the configuration of the human the method in [28] is implemented. On the other hand, our study presents solutions concerning the difficulty in achieving collision free and smooth continuous robot motion, which is particularly visible in an industrial setup where the control algorithm switches between different controllers depending on the task-in-hand. In summary, several challenges can be pointed out:

1. Accurate definition of humans/obstacles and robot pose in space using geometric primitives, and calculation of the minimum distance between them;
2. Achieving reliable autonomous human-robot collision avoidance in which the robot adapts the off-line generated nominal paths while keeping the task goal/target. In

such a case, instead of stopping or reducing robot's velocity when humans are nearby, the robot has to continue its motion while avoiding the humans/obstacles;

3. The control strategy shall produce continuous motion of robot's reaction when it adjusts the path to avoid collision. This continuity shall be guaranteed even when switching between different controllers;
4. Industrial applications require high-performance control in terms of motion accuracy and agility;
5. Collision free robot motion should be possible and reliable in the entire working volume of the robot.

Experiments demonstrated the ability of the proposed solution to achieve on-line human-robot collision avoidance materialized in the following contributions:

1. Reliable and smooth human-robot collision avoidance in which the robot adapts the off-line generated nominal paths (defined in the initial robot program) while keeping the task target. The robot finds a way to get around the human(s)/obstacles when they are nearby. Human-robot minimum distance and relative velocity are used as inputs to the implemented collision avoidance algorithm;
2. Successfully applying on-line collision avoidance on a real industrial collaborative robot performing industrial assembly tasks in collaboration with a human co-worker.

### 3 Collision Avoidance Strategy

Hypothetical attraction and repulsion vectors attract the robot towards the goal/target (defined by the off-line generated nominal paths) while repelling it away from human(s)/obstacles. By coupling this concept with a mathematical representation of robot's kinematics we can achieve a task level control with collision avoidance capability.

#### 3.1 Repulsion

A vector  $\mathbf{v}_{cp.rep}$  acts on the point of the robot closest to the obstacle (CP) repelling it away from collision. This vector is defined considering a magnitude  $v_{rep.mod}$  (calculated from a base repulsion amplitude  $v_{rep}$ ) and a direction  $\mathbf{s}$ :

$$\mathbf{v}_{cp.rep} = v_{rep.mod}\mathbf{s} \quad (1)$$

The direction of the repulsion vector  $\mathbf{s}$  is taken to be aligned with the line segment associated with the minimum distance:

$$\mathbf{s} = \frac{\mathbf{r}_1 - \mathbf{r}_2}{|\mathbf{r}_1 - \mathbf{r}_2|} \quad (2)$$

Where  $\mathbf{r}_1$  is the position vector of the point of the robot closest to the obstacle and  $\mathbf{r}_2$  is the position vector of the point of the obstacle closest to the robot. For calculating the base repulsion amplitude  $v_{rep}$  we propose to superimpose the repulsion due to the minimum distance ( $v_{rep1}$ ) and the repulsion due to the relative velocity between the human and the robot ( $v_{rep2}$ ), so that  $v_{rep} = v_{rep1} + v_{rep2}$ . Here,  $v_{rep1}$  is calculated from the minimum distance  $d_{min}$ :

$$v_{rep1} = \begin{cases} k_1 \left( \frac{d_0}{d_{min} - d_{cr}} - 1 \right), & \text{if } d_{min} - d_{cr} < d_0 \\ 0 & \text{if } d_{min} - d_{cr} \geq d_0 \end{cases} \quad (3)$$

Where  $k_1$  is a constant,  $d_0$  is an offset distance around the obstacle's capsule, it specifies the area around the obstacle where the repulsion vector is activated, and  $d_{cr}$  is a critical distance below which the robot is not allowed to be near the human. To enhance the responsiveness of the robot, we propose a dynamical reshaping of the size of the area of influence around the obstacle  $d_0$ , such that the value of  $d_0$  increases when the relative velocity between the robot and the obstacle increases:

$$d_0 = \begin{cases} d_1 - c_v v_{rel} & v_{rel} < 0 \\ d_1 & v_{rel} \geq 0 \end{cases} \quad (4)$$

Where  $v_{rel}$  is the human-robot relative velocity,  $c_v$  is a constant and  $d_1$  is the minimum value of the area of influence around the obstacle. For  $v_{rep2}$  we have:

$$v_{rep2} = \begin{cases} -c k_2 v_{rel} & v_{rel} < 0 \\ 0 & v_{rel} \geq 0 \end{cases} \quad (5)$$

Where  $k_2$  is a damping constant and  $c$  is a coefficient that takes into consideration the proximity of the obstacle from the robot:

$$c = \begin{cases} 1 & d_{min} < l_1 \\ \frac{1 + \cos(\pi \frac{d_{min} - l_1}{l_2 - l_1})}{2} & l_1 < d_{min} < l_2 \\ 0 & l_2 < d_{min} \end{cases} \quad (6)$$

Where  $l_1$  and  $l_2$  are constant distances that define the range around the robot where the damping force is activated. The intuition of using  $c$  is that obstacles far away from the robot shall not affect robot's motion since that they do not pose any risk of collision.

The modified repulsion magnitude  $v_{rep.mod}$  is calculated from  $v_{rep}$  according to Algorithm "Modified repulsion vector". For complex industrial collaborative operations, the collision avoidance controller is typically embedded in a state machine, where the collision avoidance functionality is activated/deactivated according to the tasks being performed. In such a case, discontinuity could appear when calculating the repulsion action. As an example, if the controller is switched (from the collision avoidance deactivation to the collision avoidance activation) while the co-worker is near the robot, discontinuity appears. In such a case, a high magnitude of the repulsion vector will act on the robot suddenly. To solve this problem, the repulsion action is proposed to be time dependent, by introducing the concept of repulsion-vector-reshaping coefficient  $\gamma$ , such that when the control scheme is switched the repulsion magnitude is allowed to increase monotonically starting from zero up to its stable value. In the Algorithm,  $now$  is a function returning the current time,  $\tau$  is a time constant that can be calculated from  $v_{max}/(5a_{max})$ , where  $v_{max}$  and  $a_{max}$  are the maximum curvilinear velocity and acceleration of the end-effector (EEF) used during collision avoidance, respectively.

Figure 2 shows a block diagram illustrating the proposed method for calculating the modified repulsion vector  $v_{rep.mod}$  and its relationship to  $v_{rep1}$  and  $v_{rep2}$ .

### 3.2 Attraction

An attraction velocity vector  $\mathbf{v}_{e.att}$  attached to the EEF guides the robot towards the goal/target, Fig. 3. This vector is a function of the error  $\mathbf{e}$  between EEF's position  $\mathbf{p}_e$

---

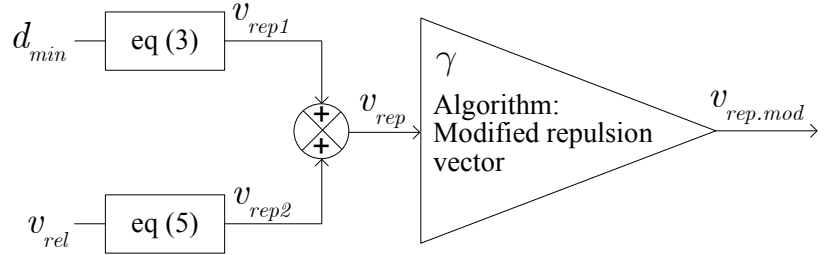
**Algorithm 1** Modified repulsion vector
 

---

```

1: for each time step  $\Delta t$  do
2:   if controller switched then
3:      $t_0 = now$ 
4:   end if
5:    $t = now - t_0$ 
6:    $\gamma = 1 - \exp(-t/\tau)$ 
7:    $v_{rep.mod} = \gamma v_{rep}$ 
8: end for
  
```

---



**Figure 2.** Block diagram showing the proposed method for calculating the magnitude of the modified repulsion vector.

and the goal position  $\mathbf{p}_g$  (defined in the off-line generated nominal paths):

$$\mathbf{e} = \mathbf{p}_e - \mathbf{p}_g \quad (7)$$

The attraction velocity is calculated from a proportional term ( $\psi_p$ ) and a quasi-integral term ( $\psi_i$ ):

$$\mathbf{v}_{e.att} = \beta(\psi_p + \psi_i) \quad (8)$$

Where  $\psi_p$  is a pure proportional term:

$$\psi_p = -\mathbf{K}_p \mathbf{e} \quad (9)$$

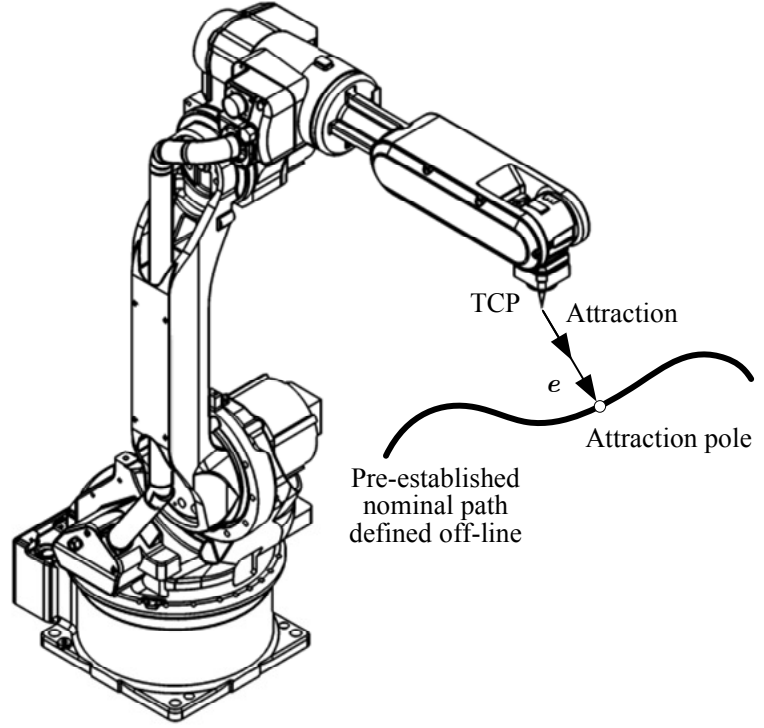
In which  $\mathbf{K}_p$  is the proportional coefficient. The quasi-integral term  $\psi_i$ , Algorithm “Integral term of the attraction vector”, prevents the quasi-integral from accumulating when the human-robot distance is less than a predefined safety distance  $d_0$ . In the Algorithm,  $d_{min}$  is the human-robot minimum distance and the integral term is calculated numerically using a simple Euler scheme (more sophisticated Runge-Kutta methods could also be used). The term  $\beta$  is used to reduce the magnitude of the attraction vector. This term has the effect of detaching the EEF gradually from the goal when the human co-worker is closer to the robot:

$$\beta = \left( \frac{2}{1 + e^{-\left(\frac{d_{min} - d_{cr}}{d_0}\right)^2}} - 1 \right) \quad (10)$$

Figure 4 shows a block diagram illustrating the proposed method for calculating the attraction vector.

### 3.3 Controller

The robot is controlled at the joint velocity level. The repulsion and attraction vectors are considered velocity vectors in which the repulsion velocity is calculated from (1),



**Figure 3.** The nominal path curve defined off-line, the attraction pole, and the error vector.

---

**Algorithm 2** Integral term of the attraction vector

---

```

1: for each time step  $\Delta t$  do
2:   if  $d_{min} < d_{cr} < d_0$  then
3:      $\psi_i = \psi_i - \mathbf{K}_i \int_t^{t+\Delta t} e dt$ 
4:   else
5:      $\psi_i = \psi_i$  ▷ to prevent windup of integral term
6:   end if
7: end for

```

---

and the attraction velocity at the EEF  $\mathbf{v}_{e.att}$  is calculated from (8). Calculating the overall angular velocities of the joints requires superimposing the angular velocities due to repulsion and attraction. Thus, the angular velocities due to  $\mathbf{v}_{cp.rep}$  that acts at CP is calculated using the Damped Least Squares [3]:

$$\dot{\mathbf{q}}_{rep} = \mathbf{J}_{cp}^T (\mathbf{J}_{cp} \mathbf{J}_{cp}^T + \lambda^2 \mathbf{I})^{-1} \mathbf{v}_{cp.rep} \quad (11)$$

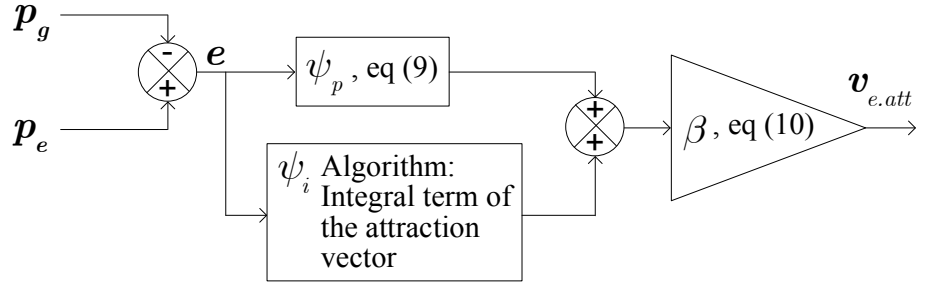
Where  $\dot{\mathbf{q}}_{rep}$  is the joint velocities vector due to the repulsion action,  $\mathbf{J}_{cp}$  is the partial Jacobian associated with CP on the robot,  $\lambda$  is a damping constant, and  $\mathbf{I}$  is the identity matrix.

The angular velocities due to  $\mathbf{v}_{e.att}$  that act at the EEF are calculated from:

$$\dot{\mathbf{q}}_{att} = \mathbf{J}_e^T (\mathbf{J}_e \mathbf{J}_e^T + \lambda^2 \mathbf{I})^{-1} \mathbf{v}_{e.att} \quad (12)$$

Where  $\dot{\mathbf{q}}_{att}$  is the joint velocities vector due to the attraction action and  $\mathbf{J}_e$  is the Jacobian associated with the EEF. Thus, the total angular velocities of the joints sent to the robot:





**Figure 4.** Block diagram showing the proposed method for calculating the attraction vector. Term  $\psi_i$  is calculated using the Algorithm Integral term of the attraction vector, which is used to avoid windup problem.

$$\dot{\mathbf{q}}_{total} = \dot{\mathbf{q}}_{att} + \dot{\mathbf{q}}_{rep} \quad (13)$$

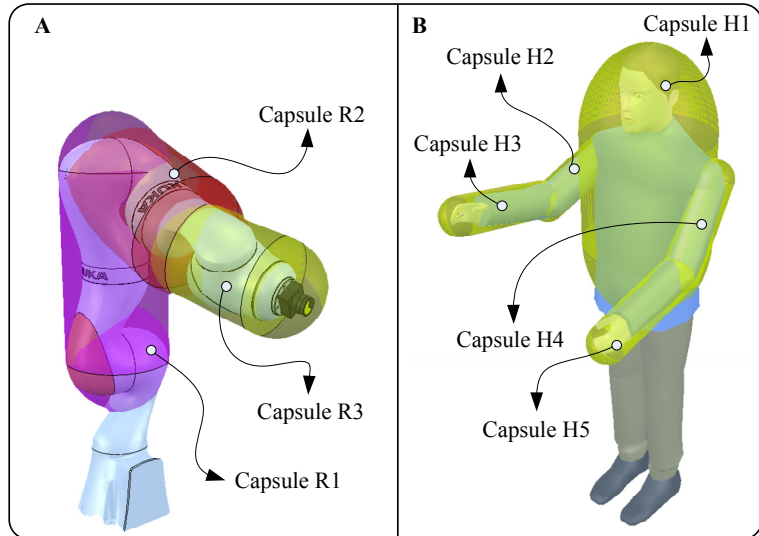
## 4 Experiments and Results

Experiments are conducted in three main configurations:

1. Configuration 1: the human arm acts as an obstacle for the robot that is performing a straight line path (off-line generated nominal path);
2. Configuration 2: the human approaches the robot from the side while the robot is stopped at a predefined home position;
3. Configuration 3: an industrial collaborative assembly operation for automotive industry in which the human co-worker approaches the robot to place a sticker in a car door card while the robot is inserting trim clips in the same door card.

### 4.1 Setup and Data Acquisition

The three experimental configurations were tested using different sensors for capturing the human pose in space. In configuration 1 and configuration 2, the proposed solution was tested with Polhemus Liberty magnetic tracking sensors attached to the human upper body (arm, forearm and chest) to acquire 6 DOF pose (position and orientation) of each body part in space. In configuration 3, the method proposed in [28] is used for capturing the human body pose from five IMUs (Technaid MCS) attached to the arms/forearms and the chest, and a laser scanner (SICK TiM5xx) at the level of the legs. An external computer Intel Core i7 with 32 GB of RAM running MATLAB<sup>®</sup> was used for performing the required computations: sensor data acquisition, capsules configuration calculation, minimum-distance and relative-velocity calculation, collision avoidance algorithms, and robot control using the KUKA Sunrise Toolbox (KST) [29]. In such a case, the robot is controlled at the kinematics level without considering its dynamics explicitly. The low-level control is built on the DirectServo library [16] provided by the manufacturer (Kuka Roboter), a 25Hz angular position update of the real-time system of the robot controller is used. Hence, the industrial joint servo is used for controlling the joint error dynamics, while the commanded joint position is calculated according to the manufacturer's data in order not to exceed the maximum allowable angular velocity. The demonstration of an efficient and fast obstacle avoidance responsiveness is an objective of this study. However, the joint trajectories may be smoothed by filtering [1], to adjust the HRI acceptance and/or to improve the dynamic accuracy.



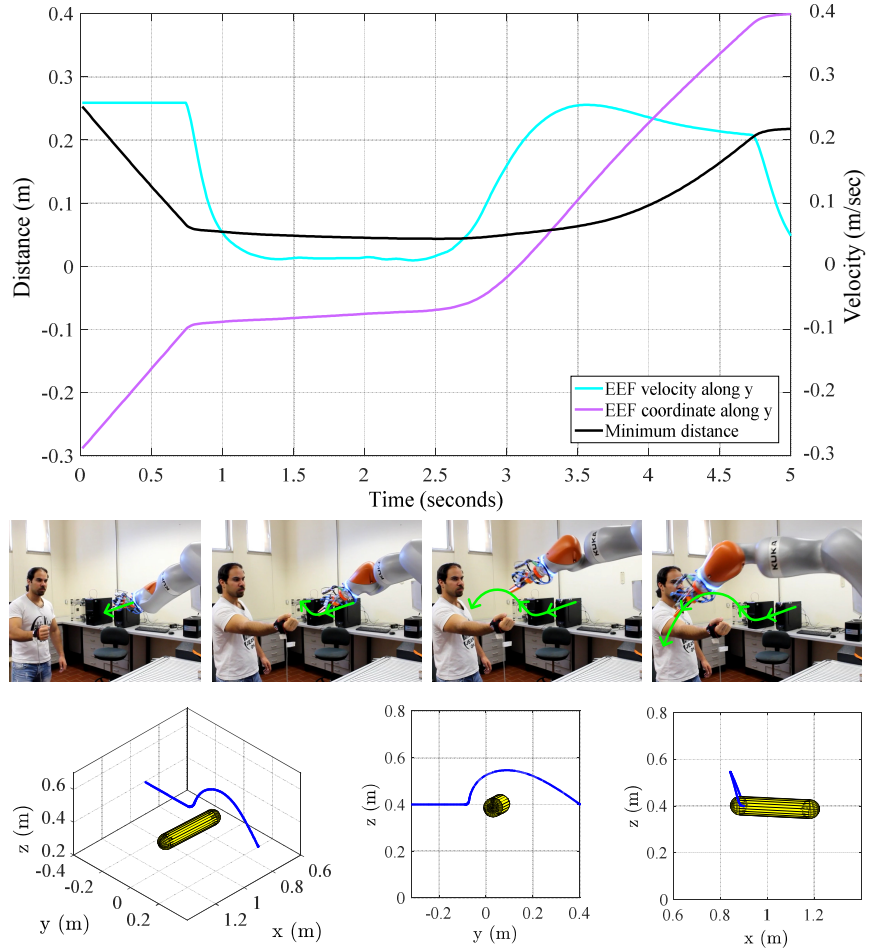
**Figure 5.** Robot (A) and human (B) represented by capsules.

## 4.2 Human and Robot Representation

The human is represented by five capsules, Fig. 5, four capsules used to cover the right/left upper arm and forearm, while the fifth capsule is used to cover the torso up to the head [28]. The robot (KUKA iiwa with 7 DOF) is represented by three capsules, Fig. 5. Capsule R3 also incorporates the tool attached to the robot. The pose of the capsules are defined by applying the forward kinematics on the joint angles acquired from the controller of the robot.

## 4.3 Results and Discussion

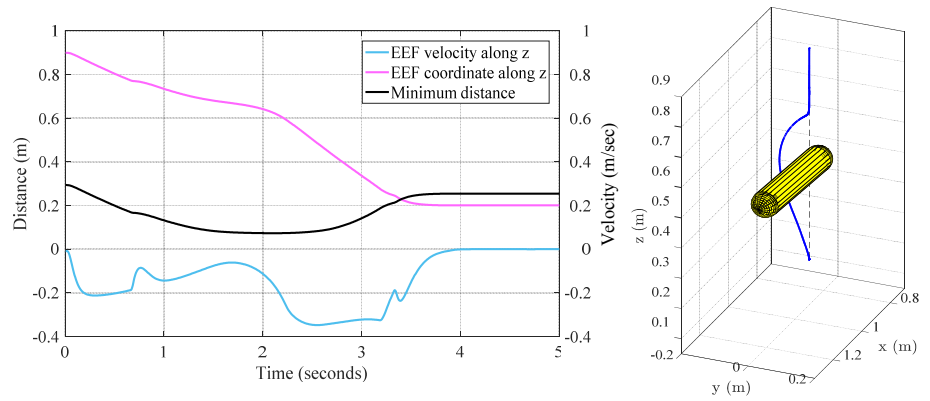
Figure 6 shows the results for configuration 1. The human forearm, represented by a capsule, is extended (almost parallel to the  $x$  axis of the robot base frame) and acts as an obstacle. The robot is moving on a straight line path (nominal path defined off-line) along the  $y$  direction of its base frame. While moving on the straight line, the minimum distance between the human-arm and the robot decreases. Consequently, the robot adapts the nominal path smoothly circumventing the human arm. At the top of Fig. 6, the graph shows the minimum distance, the velocity of the EEF and its position in Cartesian coordinates along  $y$  axis. These quantities are reported as function of time. We can notice from the plot that at the beginning the human arm is in a resting position and the robot is moving with a constant velocity of about 0.26 m/sec, along  $y$  direction, towards the human arm. When the robot EEF approaches the human arm the minimum distance decreases to a minimum of about 5 centimeters. The EEF velocity is constant until a threshold minimum distance is reached. In such scenario the EEF velocity decreases to start circumventing the obstacle and then accelerates to reach a velocity close to the nominal velocity of 0.26 m/s. We also tested the system in configuration 1 but with different initial conditions through changing the robot's initial configuration, the off-line generated path and the position/orientation of the obstacle (human's arm). Figure 7 shows the results in the case where the off-line generated path is parallel to the  $z$  axis, the robot is required to move on the path in the negative  $z$  direction, the human forearm (almost parallel to the  $x$  axis of the robot base frame) is extended in the way of the off-line generated path. From Fig. 7 (right) it is noticed that the robot adapts the off-line path successfully using the proposed method. Figure 7



**Figure 6.** Configuration 1 results (first initial condition, off-line generated path is parallel to  $y$  direction). Plot shows the minimum distance, EEF velocity and position along  $y$  axis (top). Snapshot of collision avoidance testing and collision avoidance path in 3D and 2D space (middle and bottom).

(left) shows the minimum distance, the velocity and the position of the EEF along the  $z$  axis (following the robot motion from  $z = 0.9$  m at top to  $z = 0.2$  m at the bottom), the minimum distance reached is around 7 cm, the velocity profile differs a little bit from the previous case (motion along the  $y$  axis), but the general behaviour is the same. In Fig. 8 the same test (configuration 1) was repeated with yet another initial condition, the obstacle (human arm) is inclined, and the off-line generated path is parallel to the  $xy$  plane of the robot base (starting from point  $x = 0.38$ ,  $y = -0.26$ ,  $z = 0.25$  to the point  $x = 0.8$ ,  $y = 0.3$ ,  $z = 0.25$ ). Figure 8 at the top shows the coordinates of the EEF, the minimum distance and the magnitude of EEF's velocity acquired during the collision avoidance motion, it can be noticed that the minimum distance reached is around 6.5 cm. Figure 8 at bottom shows the obstacle and robot's path in perspective view (left) and in the  $xy$  plane (right). From previous results it can be concluded that the robot manages to avoid collision with the co-worker successfully and the collision avoidance controller smoothly reacts to avoid collision while reaching the task target.

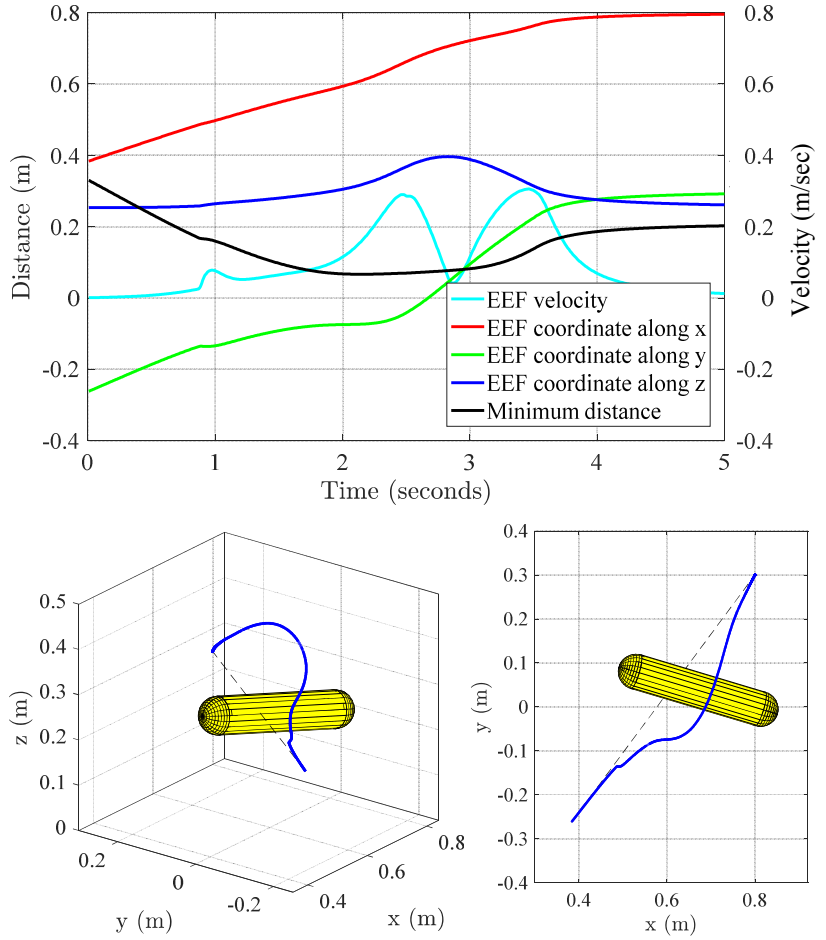
In configuration 2 the human approaches the robot from the side while the robot is stopped at a pre-defined home position. As the human approaches the robot the human-robot minimum distance decreases and the robot reacts in an agile-smooth



**Figure 7.** Configuration 1 results (second initial condition, off-line generated path is parallel to  $z$  axis). Plot shows the minimum distance, EEF velocity and position along  $z$  axis (left). Path of EEF in 3D (right).

behaviour to avoid collision. At the top of Fig. 9, the same quantities presented for configuration 1 show that at the beginning the human starts walking towards the robot, when the minimum distance reaches 0.5 meters, the robot reacts to avoid collision. When the human goes away the robot returns to the initial home position. The robot successfully avoided collision as in the snapshots at the bottom of Fig. 9 and in the video in extra multimedia materials.

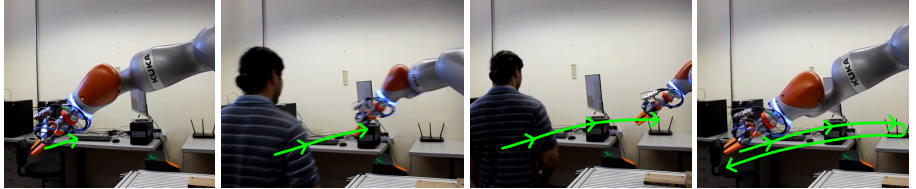
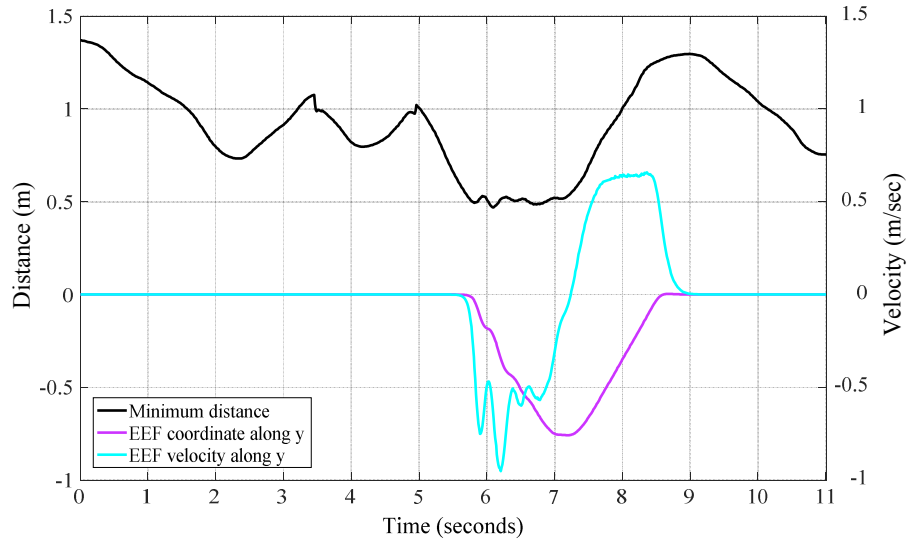
Flexible manufacturing, and industrial assembly processes in particular, present several challenges due to the unstructured nature of the industrial environment. Some tasks are more suited to be executed by humans, others by robots, and others by the collaborative work between human and robot. The ability to have humans and robots working side-by-side will bring enormous efficiency benefits to flexible manufacturing. However, this scenario is challenging, due to the requirement of having the robot avoiding collisions with the co-worker in real-time, allowing him/her to focus on the manufacturing tasks and not on the potential danger from the robot side. In this context, we tested the proposed system, in configuration 3, in an automotive sample case in which the human co-worker approaches the robot to place a sticker in a car door card while the robot is inserting trim clips in the same door card, Fig. 10 (video in extra multimedia materials). This flexible collaborative task allows the co-worker to manage his/her working time and sequencing of operations since he/she is free to place the sticker in the door card at any time and devotes attention to other tasks that he/she has to take care of. Meanwhile the robot continues inserting the trim clips in the door card by using its force feedback to compensate for deviations in the door card positioning. When the human co-worker approaches the robot it adapts the nominal path to avoid collisions in a smooth way while keeping its task. For this sample case, the pre-established nominal path is divided in 3 sub-path segments, Fig. 11 (A). In segments 1 and 3 (green lines) the collision avoidance control is activated (collision avoidance (CA) paths) while in segment 2 (red and blue lines) is deactivated. This is because this path is defined to be the working path where the robot is inserting the trim clips at relative reduced velocity. Starting from a given home position coincident with the beginning of segment 1 and the tool centre point (TCP) the system behaves as in Algorithm “collision avoidance - automotive sample case”. In Fig. 11 (B) the robot and goal point move along segment 1 so that an error vector is established. If the human is detected in the safety zone the collision avoidance is activated and the goal point stops moving, Fig. 11 (C). When the human is not in the safety zone the robot returns back to the goal point that starts moving with the robot, Fig. 11 (D). When the



**Figure 8.** Configuration 1 results (third initial condition, off-line generated path is inclined in a plane parallel to  $xy$  of the robot base). Plot shows the minimum distance, EEF velocity and position (top). Path of EEF in 3D (bottom).

robot reaches segment 2 the collision avoidance is deactivated, Fig. 11 (E). Due to the use of IMUs to capture the configuration of the human, in practice, the noise in the measurements shall be filtered. In our study, we utilized the Technaid IMU system, where filters [22] are implemented on an external processing unit (inside the system’s hub). Consequently, using the manufacturer’s API (Application Programming Interface) we can acquire the orientation of each IMU from an external PC, wirelessly through a Bluetooth connection. Since that the accuracy of the IMU measurements is represented by a maximum orientation error (for example, the Technaid data sheets specify a one-degree angular error), then based on the kinematics of the human body [28], an estimate of the resulting maximum Cartesian error can be calculated. In such a case, our algorithm allows minimizing the effect of the Cartesian error (due to the inaccuracy of the IMU angular measurement) simply by increasing  $d_{cr}$  (equation 3) to include the maximum Cartesian error. In such a case, the uncertainty in the calculated minimum distance (resulting from IMU inaccuracy) will be less than the critical distance  $d_{cr}$  (the minimum distance below which the robot cannot be near the human), consequently, we can be sure that the robot does not touch the human even when a maximum measurement error is present.

Based on the tests, all the users indicate that the system does not appear to be

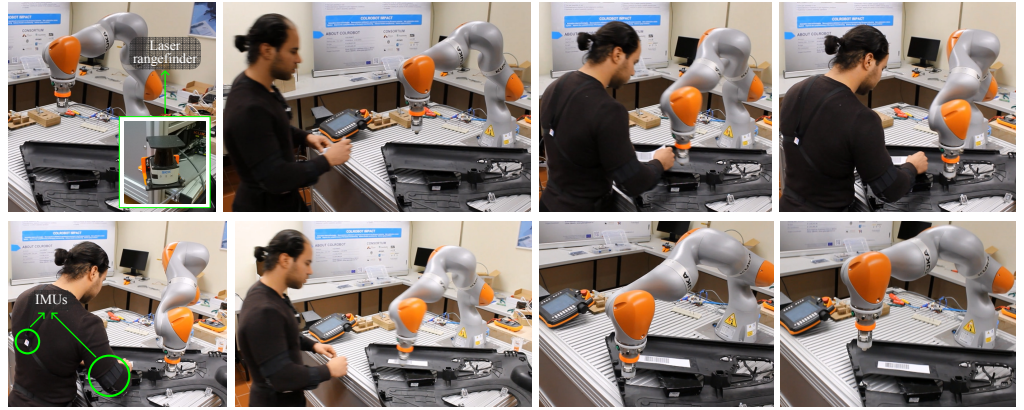


**Figure 9.** Configuration 2 results. Minimum distance, EEF velocity and position along  $y$  axis (top). Snapshot of collision avoidance testing (bottom).

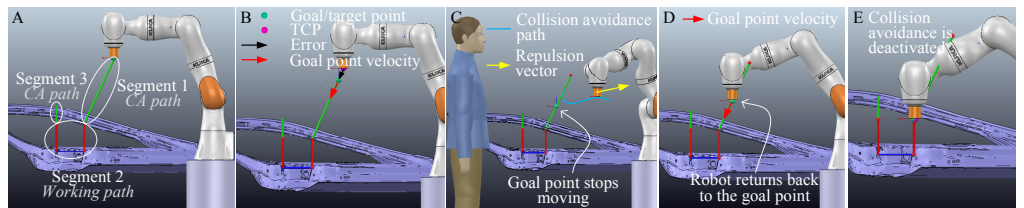
dangerous virtue to the collision avoidance motion which is perceived as smooth and natural. It is also demonstrated that the system performs well even in situations where the human is showing a hesitation or a back-and-forth motion, this is shown during tests (in the attached video segment, from seconds 43 to 47), where the co-worker is moving his hand back and forth towards the robot while the robot is reacting to avoid collisions smoothly. However, a final determination of the effect on the co-worker's psychology (feelings of danger, fear, security, distraction) will require a dedicated study in collaboration with psychologists, involving more users and taking into consideration various factors (including: age and background), as such it will be left open for future work.

It is hard to perform a quantitative comparison between our algorithm and others, since that various studies have utilized different hardware architectures (robots, control systems), different type of sensors, and different scenarios. However, qualitatively speaking, our study, unlike others, demonstrates the feasibility of on-line collision avoidance in HRI situation for a real industrial use case while taking into consideration various important aspects:

1. Unlike other methods, our method is tailored for industrial application, our algorithm implements off-line generated paths of the EEF (important for performing industrial operation) combined with an on-line reactive collision avoidance, to avoid collisions with the human co-worker;
2. The algorithm takes into consideration various aspects to guarantee the continuity of the motion (due to the necessity of switching between different controllers imposed by the industrial task);
3. In our tests, it is used a real industrial manipulator KUKA iiwa (certified for



**Figure 10.** Automotive sample case for car door card assembly (video in extra multimedia materials).



**Figure 11.** Pre-established sub-path segments for the automotive sample case. This process is detailed in the algorithm Collision avoidance - automotive sample case.

industrial use), and utilizing commonly-used sensors (well-established technologies);

4. A unique feature of our application is that we developed the low-level control of our algorithm using the DirectServo library of the KUKA iiwa. In such a case, all the security features of the Sunrise.OS (operating system of the KUKA iiwa controller) are running in parallel with our developed controlling program. This includes the collision detection feature, which is activated by our control program during the free collision avoidance motion, adding an additional safety layer to the system;
5. Unlike most of the studies, we demonstrated our algorithm in the context of a real-life (not simulation) industrial scenario (assembly operation).

## 5 Conclusion

This paper presented a novel method for human-robot collision avoidance for collaborative robotics tailored for industrial applications. The collision avoidance controller demonstrated on-line capabilities to avoid collisions while the robot continues working by keeping the task target. The concept of repulsion-vector-reshaping was introduced to guarantee the continuity of the generated motion when switching between controllers. Experiments indicated that the robot reaction to avoid collisions is well perceived by the co-worker, smooth, natural and effective.

---

**Algorithm 3** Collision avoidance - automotive sample case

---

```
1: for each time step do
2:   if human is not detected in the safety zone then
3:     if CA path then
4:       Goal point moves along path segment
5:       Error vector is generated between TCP and the goal, Fig. 11 (B)
6:       Attraction velocity is generated from the error vector
7:       if goal point reaches the segment end then
8:         Goal point stops moving
9:         The robot TCP reaches the goal point
10:      end if
11:     else
12:       Collision avoidance controller is deactivated
13:       Controller to insert trim clip is activated
14:     end if
15:   else human is detected in the safety zone
16:     if CA path then
17:       Goal point stops moving
18:       Integral term stops accumulating
19:       A repulsion velocity  $v_{rep}$  acts in the robot, Fig. 11 (C)
20:     else
21:       Collision avoidance controller is deactivated
22:       Controller to insert trim clip is activated, Fig. 11 (E)
23:     end if
24:   end if
25: end for
```

---



---

## 6 Acknowledgements

This research was partially supported by Portugal 2020 project DM4Manufacturing POCI-01-0145-FEDER-016418 by UE/FEDER through the program COMPETE 2020, and the Portuguese Foundation for Science and Technology (FCT) SFRH/BD/131091/2017 and COBOTIS (PTDC/EMEEME/ 32595/2017).

## References

1. P. Besset and R. Béarée. Fir filter-based online jerk-constrained trajectory generation. *Control Engineering Practice*, 66:169 – 180, 2017.
2. P. Bosscher and D. Hedman. Real-time collision avoidance algorithm for robotic manipulators. *Industrial Robot: An International Journal*, 38(2):186–197, 2011.
3. S. R. Buss. Introduction to inverse kinematics with jacobian transpose, pseudoinverse and damped least squares methods. *IEEE Journal of Robotics and Automation*, 17(1-19):16, 2004.
4. X. Chu, Q. Hu, and J. Zhang. Path planning and collision avoidance for a multi-arm space maneuverable robot. *IEEE Transactions on Aerospace and Electronic Systems*, 54(1):217–232, 2018.
5. F. Fabrizio and A. D. Luca. Real-time computation of distance to dynamic obstacles with multiple depth sensors. *IEEE Robotics and Automation Letters*, 2(1):56–63, Jan 2017.
6. A. Fenucci, M. Indri, and F. Romanelli. A real time distributed approach to collision avoidance for industrial manipulators. In *Proceedings of the 2014 IEEE Emerging Technology and Factory Automation (ETFA)*, pages 1–8, Sept 2014.
7. F. Flacco, T. Kröger, A. De Luca, and O. Khatib. A depth space approach to human-robot collision avoidance. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pages 338–345. IEEE, 2012.
8. F. Flacco, A. D. Luca, and O. Khatib. Control of redundant robots under hard joint constraints: Saturation in the null space. *IEEE Transactions on Robotics*, 31(3):637–654, June 2015.
9. S. Haddadin, A. Albu-Schäffer, and G. Hirzinger. Requirements for safe robots: Measurements, analysis and new insights. *The International Journal of Robotics Research*, 28(11-12):1507–1527, 2009.
10. S. Haddadin, R. Belder, and A. Albu-Schaffer. Dynamic motion planning for robots in partially unknown environments\*. *IFAC Proceedings Volumes*, 44(1):6842 – 6850, 2011. 18th IFAC World Congress.
11. H. Hashimoto. Intelligent interactive spaces - integration of it and robotics. In *IEEE Workshop on Advanced Robotics and its Social Impacts, 2005.*, pages 85–90, June 2005.
12. B. Jeon, H. Kim, and H. J. Kim. Collision avoidance of robotic arm of aerial manipulator. In *Control Conference (ASCC), 2017 11th Asian*, pages 1859–1864. IEEE, 2017.

- 
13. L. E. Kavraki, P. Svestka, J. . Latombe, and M. H. Overmars. Probabilistic roadmaps for path planning in high-dimensional configuration spaces. *IEEE Transactions on Robotics and Automation*, 12(4):566–580, Aug 1996.
  14. M. Khatib, K. A. Khudir, and A. De Luca. Visual coordination task for human-robot collaboration. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3762–3768, Sep. 2017.
  15. O. Khatib. Real-time obstacle avoidance for manipulators and mobile robots. *The international journal of robotics research*, 5(1):90–98, 1986.
  16. KUKA. *KUKA Sunrise.Connectivity Servoing 1.7*. KUKA Roboter GmbH, 2015.
  17. P. Leven and S. Hutchinson. A framework for real-time path planning in changing environments. *The International Journal of Robotics Research*, 21(12):999–1030, 2002.
  18. R. Meziane, M. J.-D. Otis, and H. Ezzaidi. Human-robot collaboration while sharing production activities in dynamic environment: Spader system. *Robotics and Computer-Integrated Manufacturing*, 48(Supplement C):243 – 253, 2017.
  19. A. Mohammed, B. Schmidt, and L. Wang. Active collision avoidance for human-robot collaboration driven by vision sensors. *International Journal of Computer Integrated Manufacturing*, 30(9):970–980, 2017.
  20. P. Neto, M. Simão, N. Mendes, and M. Safeea. Gesture-based human-robot interaction for human assistance in manufacturing. *The International Journal of Advanced Manufacturing Technology*, Oct 2018.
  21. S. Nikolaidis, P. Lasota, R. Ramakrishnan, and J. Shah. Improved human-robot team performance through cross-training, an approach inspired by human team training practices. *The International Journal of Robotics Research*, 34(14):1711–1730, 2015.
  22. S. L. Nogueira, S. Lambrecht, R. S. Inoue, M. Bortole, A. N. Montagnoli, J. C. Moreno, E. Rocon, M. H. Terra, A. A. G. Siqueira, and J. L. Pons. Global kalman filter approaches to estimate absolute angles of lower limb segments. *BioMedical Engineering OnLine*, 16(1):58, May 2017.
  23. R. Palm. Control of a redundant manipulator using fuzzy rules. *Fuzzy Sets and Systems*, 45(3):279 – 298, 1992.
  24. V. Perdureau, C. Passi, and M. Drouin. Real-time control of redundant robotic manipulators for mobile obstacle avoidance. *Robotics and Autonomous Systems*, 41(1):41–59, 2002.
  25. F. Rubio, C. Llopis-Albert, F. Valero, and J. L. Suñer. Industrial robot efficient trajectory generation without collision through the evolution of the optimal trajectory. *Robotics and Autonomous Systems*, 86:106–112, 2016.
  26. J. Saenz, N. Elkmann, O. Gibaru, and P. Neto. Survey of methods for design of collaborative robotics applications-why safety is a barrier to more widespread robotics uptake. In *4th International Conference on Mechatronics and Robotics Engineering*, 2018.
  27. M. Safeea, R. Bearee, and P. Neto. End-effector precise hand-guiding for collaborative robots. In *Iberian Robotics conference*, pages 595–605. Springer, 2017.

- 
28. M. Safeea and P. Neto. Minimum distance calculation using laser scanner and imus for safe human-robot interaction. *Robot and Computer-Integrated Manufacturing*, 54(1):217–232, 2018.
  29. M. Safeea and P. Neto. Kuka sunrise toolbox: Interfacing collaborative robots with matlab. *IEEE Robotics Automation Magazine*, 26(1):91–96, March 2019.
  30. M. Safeea, P. Neto, and R. Bearee. Efficient calculation of minimum distance between capsules and its use in robotics. *IEEE Access*, 7:5368–5373, 2019.
  31. B. Schmidt and L. Wang. Depth camera based collision avoidance via active robot control. *Journal of Manufacturing Systems*, 33(4):711 – 718, 2014.
  32. A. Shukla, E. Singla, P. Wahi, and B. Dasgupta. A direct variational method for planning monotonically optimal paths for redundant manipulators in constrained workspaces. *Robotics and Autonomous Systems*, 61(2):209–220, 2013.
  33. L. Singh, H. Stephanou, and J. Wen. Real-time robot motion control with circulatory fields. In *Proceedings., 1996 IEEE International Conference on Robotics and Automation*, volume 3, pages 2737–2742. IEEE, 1996.
  34. A. Skoglund, B. Iliev, B. Kadmiry, and R. Palm. Programming by demonstration of pick-and-place tasks for industrial manipulators using task primitives. In *2007 International Symposium on Computational Intelligence in Robotics and Automation*, pages 368–373. IEEE, 2007.
  35. J. E. Solanes, L. Gracia, P. Muñoz-Benavent, J. V. Miro, M. G. Carmichael, and J. Tornero. Human-robot collaboration for safe object transportation using force feedback. *Robotics and Autonomous Systems*, 2018.
  36. M. Svenstrup, S. Tranberg, H. J. Andersen, and T. Bak. Pose estimation and adaptive robot behaviour for human-robot interaction. In *2009 IEEE International Conference on Robotics and Automation*, pages 3571–3576, May 2009.
  37. A. M. Zanchettin, B. Lacevic, and P. Rocco. Passivity-based control of robotic manipulators for safe cooperation with humans. *International Journal of Control*, 88(2):429–439, 2015.