Minimum distance calculation using laser scanner and IMUs for safe human-robot interaction

Mohammad Safeea, Pedro Neto

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

Arts et Métiers, ParisTech, 59800 Lille, France

Abstract

In this study we investigate the use of a laser scanner/range-finder and inertial measurement units (IMUs) for the application of human-robot interaction in a dynamic environment with moving obstacles/humans. Humans and robots are represented by capsules, allowing to calculate the human-robot minimum distance on-the-fly. A major challenge is to capture the capsules pose. Data from a laser scanner and IMUs attached to the human body are fused to define the torso relative position and the upper body (arms and chest) configuration, respectively. Collision avoidance is achieved with a customized potential field's method that allows to adjust the pre-defined robot paths established off-line while keeping the task target. The proposed framework is validated in real environment using a SICK laser scanner, IMUs and a KUKA iiwa robot. Experiments demonstrated the robustness of the proposed approach in capturing human motion, calculating the human-robot minimum distance and the robot behaviour that smoothly avoids collisions with the human.

Keywords: Human tracking, Human-robot interaction, Collaborative robots, Sensor fusion, Laser scanner, IMU

1. Introduction

Robots operating around humans and sharing the workspace will become a reality in a near future. However, robots need to become safe in the way they interact and collaborate with humans. Safety is a major concern in collaborative robotics since robots and humans will coexist and share the same workspace. A collaborative robot shall be capable of detecting obstacles (including dynamic obstacles such as humans) by acquiring and processing sensor data related to the robot surrounding environment. These data can be used to calculate the proximity between robots and humans. The proximity is measured

Email address: ms@uc.pt, pedro.neto@dem.uc.pt (Mohammad Safeea, Pedro Neto)

by the human-robot minimum distance, which is the main input for most of the algorithms related to collision avoidance.

Industrial collaborative robots are a key element in the materialization of the Industry 4.0 concept. Safety issues are one of the main factors which can break down the boundaries that limit the direct contact between humans and robots. Nowadays, in the factory floor, manipulators have to be separated behind guarding fences and humans are not allowed to enter into the working area of the robot while the robot is in operation. This is due to the fact that industrial manipulators are still blind to their surroundings and pose a fundamental danger to humans.

The standard ISO 10218 and the technical specification TS 15066 provide guidelines for risk assessment implementation and define the safety requirements for collaborative robots. An overview for the speed and separation monitoring (SSM) methodology according to the TS 15066 is presented in [1]. This study includes analytical analyses and discusses considerations for implementing SSM in collaborative robotics. The directions for technological advancements toward standardization are also discussed. Different approaches to risk assessment for collaborative robots are presented in [2]. Risk assessment is required to evaluate and anticipate risks for the human in human-robot collaboration. The safety requirements and the potentialities of systems engineering allowing faster and more reliable deployment of collaborative robotics are discussed in [3]. Application use cases are detailed, namely for machine tending, automotive assembly and palletizing applications.

A method for controlling the velocity of a collaborative robot to ensure human safety even when the human-robot distance is smaller than the safe separation distance is proposed in [4]. The allowable maximum safe velocity is calculated using a collision model that predicts the collision peak pressure and the peak force in case of collision. The pressure and force threshold from ISO/TS 15066 are used to estimate the allowable maximum velocity of the robot as a function of the distance between the robot and the human. In [5], an optimization of safeguarded workspaces is introduced under the ISO/TS 15066 SSM mode. A trajectory-dependent dynamic speed and separation monitoring volume is considered for establishing the minimum safety volume. An interesting study addresses human localization to enable SSM safety according to ISO/TS 15066 [6]. Wireless sensor networks distributed in fixed positions inside the robotic cell are proposed to localize operators.

The number of existing studies approaching on-line collision avoidance according to ISO/TS 15066 is very limited. Most of them present results in simulation or do not discuss the implications of the ISO/TS 15066 in their laboratory setups. In this paper we partially address the ISO 10218 and TS 15066 in the sense that the collaborative scenario type is ruled by SSM. The robot on-line reacts to keep a separation distance to the human by on-line adjusting the nominal path and speed.

The potentialities of collaborative robots are not yet fully explored, opening a world of research and technology development opportunities. To achieve the long-sought goal of having robots in human centred environments, human safety

shall be guaranteed, and the possibilities of collision shall be eliminated. Thus, the subject of collision avoidance is one of the essential questions that need to be addressed for assuring the safety of the human co-worker when interacting with a robot. Yet, collision avoidance algorithms are hard to develop, especially due to the lack of reliable sensor data to estimate the human-robot minimum distance on-line. Capturing accurate information from multiple sensor systems in real-time and in an unstructured environment is still difficult to achieve. Existing solutions relying on marker-based visual tracking such as Vicon system are accurate but limited to relatively small areas and with significant cost [7].

Wearable inertial sensors have been used for human motion tracking [8], not requiring external cameras or markers. They can be used in both outdoor and indoor environment, with no light restrictions nor suffering from occlusions. Nevertheless, drift is a major problem associated to these sensors, especially over long periods of time. Some authors propose to correct the estimated quantities (for example the position of the body with respect to a coordinate system not fixed to the body) by updating these quantities based on biomechanical characteristics of the human body, detection of contact points of the body with an external world and adding other sensors to the system [9]. Our experience indicates that drift is a real problem if not correctly addressed.

In our study, the IMUs from TECHNAID are MEMS-based sensors, which are able to provide accurate data over short time intervals. However, over long time intervals accuracy degrades due to the effect of MEMS' characteristic errors. The determination and compensation of these characteristic errors (random noise, bias, drifts, orthogonality between axes, etc.) are contemplated in our TECHNAID IMUs [10]. Measuring the earth's magnetic field enables an estimate of the orientation without drift. However, its measurement may be affected by the presence of metal objects and electromagnetic noise. To solve this problem, the TECHNAID IMUs incorporate a compensation system that compensates the estimation of the orientation during transient magnetic disturbances.

A human motion tracking approach combining a mobile robot (equipped with a laser scanner) and an inertial motion capture system is proposed in [11]. The mobile robot is used to anchor the pose estimates of the human which is wearing a motion capture suit equipped with 17 Xsens IMUs to estimate the body posture. The system captures the motion of a human in large areas (outdoor) fusing data from laser and IMUs for more accurate tracking. The study presents the trajectory of the human in meters scale (outdoor) but the entire skeleton pose is not detailed. In our proposed approach we are using similar sensors, laser and IMUs, but with only 5 IMUs for upper-body tracking. Our aim is to detect the position of the entire human body in an area of about 2 meters maximum around the robot (the maximum reach of the robot manipulator is around 1 meter). In this context, it is relevant to have accuracies in centimetres scale for the whole human body tracking because, for example, the human chest can be at 1 meter of the robot but the arm if extended can be very close to the robot. A simple methodology to estimate the human torso position from the legs position is proposed, so that the information related to

human tracking is used to define the pose in space of the capsules representing the human. The human-robot minimum distance is estimated and a customized potential field's method is implemented for human-robot collision avoidance.

Different methods for representing humans and robots geometry (normally using spheres and capsules) have been proposed in literature. A computationally efficient way to represent robots and obstacles relies on the use of primitive shapes [12, 13]. Ellipses and spheres were used to represent robots and obstacles, as in [14]. In [15], a humanoid is represented by cylinders since that such representation allows for efficient calculation of the minimum distance between geometries to perform self-collision avoidance. In [16], the GPU processing power was used to calculate the minimum distance between objects represented by meshes. This method provides a precise representation of objects, but it is hard to implement. The skeletal algorithm proposed in [17] represents a framework for self-collision avoidance of a humanoid robot represented by spheres and cylinders. A robot represented by twelve bounding boxes (mainly cylinders) was proposed in [13]. An advanced collision map for performing point-to-point motion with collision avoidance capability in a robotic cell with two robotic manipulators in which each robot is represented by four cylinders is presented in [18]. In [19], the use of depth cameras is studied to improve the monitoring in a human-robot collaborative environment. This is important since the lack of sensors reporting reliable data is a major problem in this kind of applications. Different sensors have been developed for human tracking applied to robotics field. Many solutions are based on vision sensing, including RGB-D sensor tracking [19, 20]. An increasing number of approaches are using laser scanners, 2D and 3D [21, 22, 23]. From the previous studies, it can be concluded that the choice of the geometric primitive to represent humans and robots in a given environment is important for the accuracy of the representation and the computational cost required to compute the minimum distance. Critical importance is also associated to the sensors for human motion tracking.

An obstacle avoidance approach based on the artificial potential fields (PF) concept is introduced in the pioneering work of Khatib [24]. The robot is in a hypothetical vector field influenced by forces of attraction that guide the robot towards the target and forces of repulsion that repel it away from humans/obstacles. Subjected to these forces the robot finds its way to the target while avoiding collisions. Recently, a depth space approach for collision avoidance proposes an improved implementation of the potential fields method in which an estimation of obstacle's velocity was taken into consideration when computing the repulsion vector [20]. Robot self-collision avoidance has been studied, as well as the development of collision avoidance techniques for redundant robots [25]. A general framework for movement generation and mid-flight adaptation to obstacles is presented in [26]. Dynamic movement primitives are extended such that arbitrary movements in end-effector space can be represented. A method for motion generation and reactive collision avoidance is proposed in [27]. The method relies on physical analogies for defining attractor dynamics to generate smooth paths. The algorithm can run in the internal control loop of the robot, which is an important issue for safety. An on-line

collision avoidance system is proposed in [28]. Virtual 3D models of robots and real images of human operators from depth cameras are used for monitoring and collision detection purposes. It is presented a prototype for adaptive robot control in which the result of collision detection has four safety strategies: the system can alert an operator, stop a robot, move away the robot, or modify the robot's trajectory away from an approaching operator. These strategies are activated based on the operator's existence and location with respect to the robot. Collision avoidance algorithms imply complex computations applied on large amount of variables acquired in real-time from multiple sensors.

A reinforcement learning method applied to collision avoidance for manipulators using neural networks is proposed in [29], the networks were trained using data from simulations in Virtual Reality (VR), experimental tests were carried out on a 6 DOF manipulator where the position of the end-effector is controlled while the orientation of the end-effector is not taken into consideration. The application of Particle Swarm Optimization (PSO) on the problem of collision avoidance of redundant manipulators was studied in [30], tests were carried out in simulation using a 5 DOF planar manipulator where the position of the end-effector was controlled to reach a target position while avoiding collision with obstacles and the orientation of the end-effector was not taken into consideration. A method for discrete collision detection and obstacles proximity computation applied to collision avoidance of robotic manipulators is proposed in [31], the robot is approximated using flat end cylinders', obstacles are detected using Kinect camera and represented using convex hulls. Tests were carried out in simulation implementing Baxter robot in ROS (Moveit! and RViz are used for the simulation). Many of existing studies in human-robot collision avoidance still present results in simulation environment.

An alternative approach to solve the collision avoidance problem is proposed in [32]. However, such approach is restrictive since it assumes *a priori* complete knowledge of obstacle's trajectory. A study dedicated to collision avoidance between two manipulators is in [33]. The problem was addressed by dividing the work space of the manipulators into a shared work area, accessible to both manipulators, and an external work area accessible to only one manipulator. The authors added a processing layer into the control structure, in which point to point control commands are processed before being sent to the controllers. As consequence, the manipulators are allowed to operate in their own external work area at any time. However, the presence of one of the manipulators inside the shared work area will deny access to the other manipulator, causing it to wait until the shared work area is free from the other manipulator. In [34] it is proposed the Representation of Body by Elastic Elements (RoBE), which is a method used for avoiding robot self-collisions. In this method each link is covered by a fictitious elastic element, whenever the elements touch, a force is generated and collision avoidance is achieved.

In this paper we propose to fuse IMUs and laser scanner data for human tracking and consequent estimation of the human-robot minimum distance online for collision avoidance purposes. Human and robot are represented by capsules. Section 2 details how the poses of the capsules are defined in space

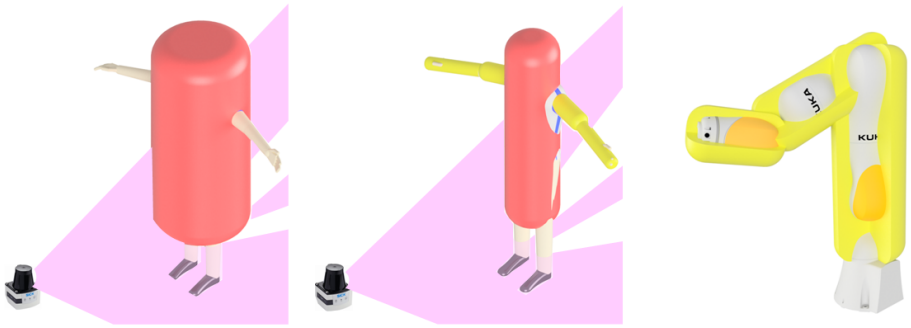

Figure 1: (a) Human represented by 1 capsule, (b) human represented by 5 capsules and (c) robot represented by 3 capsules.

using sensors data, where the position of the torso and the configuration of the upper body (arms and chest) are established. Section 3 discusses the proposed method to compute the capsule-capsule minimum distance. In Section 4, starting from the robot pre-defined paths established off-line during the teach-in phase, the proposed PF-based collision avoidance controller adjusts such paths to avoid collisions. When the obstacle goes away the robot keeps the target and continues the task. Section 5 presents the experiments and results obtained in real world experiments. Finally, conclusions are presented in Section 6.

2. Geometric representation

The human and the robot are represented in space by capsules. The more capsules we use, the higher is the accuracy and more challenging is to acquire sensor data to define the pose of each capsule and more computational power is required. Capsules are considered a good geometric primitive to represent a human. A human can be represented by a single capsule, Fig. 1 (a). In this scenario the arms are exposed such that the diameter of the capsule is greater and the collision avoidance has to be setup for a larger human-robot minimum distance. For this study, the torso and the arms are represented by 5 capsules, each arm is represented by 2 capsules and the torso and head by 1 capsule (we assume that the human is always standing straight), Fig. 1 (b). The robot (KUKA iiwa with 7 DOF) is described by 3 capsules representing the main robot links of this robot arm model, Fig. 1 (c). The pose of the robot capsules in space is obtained from the measured robot joint angles, establishing the beginning and end of each capsule.

The upper body configuration is captured using five IMUs attached to the chest and arms. The position of the legs is captured using a laser scanner (SICK TiM5xx) installed at the base of the robot working table. Assuming the human is standing straight, the position of the torso can be estimated from the legs positions.

Figure 2: Laser scanner mounted at the level of the operator's legs in the base of the robot table. If the robotic arm is installed on a mobile platform the solution is similar.

2.1. Torso capsule position

Data from the laser scanner mounted at the level of the operator's legs, Fig. 2, are utilized to define the relative position of the capsule representing the torso. Through TCP/IP connection the sensor (SICK TiM5xx) provides the radius measurements along the scan-angle with a range of 270 degrees, a scan-angular precision of 1 degree and with a maximum measurement radius range of 8 meters. The methodology behind the algorithm for calculating the torso position is divided into two steps:

1. Data acquisition and filtering;
2. Calculating the minima and the position of the torso.

2.1.1. Data acquisition and filtering

A TCP/IP server is implemented in MATLAB to acquire the measurements from the laser sensor and decodes the received message. The result is stored in an array of radius measurements against scan-angle. Fig. 3 shows the radius measurements along the scan-angle as acquired from the laser sensor corresponding to a scene where a human is standing in the scan field of the sensor. The radius measurements are clipped to 1400 mm away from the sensor and the contour of legs are projected into the plot as two minima. It is noticed that the data are noisy so that two filtering methods are proposed: (1) filtering through time and (2) filtering along the scan-angle.

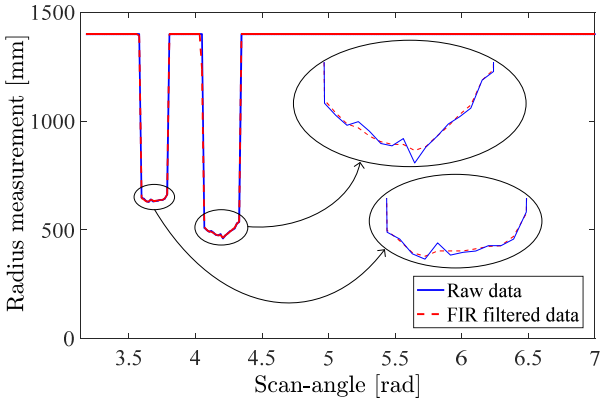

Figure 3: Raw data and FIR filtered of radius measurement with scan-angle, close ups show filtered data are smoother in the critical parts of the curves.

2.1.2. Filtering through time

To filter noise in the measurements from the laser sensor a finite impulse response (FIR) filter is utilized:

$$r_{(\theta,t)} = \alpha r^{m}_{(\theta,t)} + (1-\alpha) r_{(\theta,t-dt)} \tag{1}$$

Where $r_{(\theta,t)}$ is the filtered measurement value of the radius at angle θ and time t, $r^{m}_{(\theta,t)}$ is the measurement value of the radius at angle θ and time t, α is a scalar from zero to one (defined by an iterative process according to actual response), and dt is the update time interval between two consecutive scans. The results of the application of the FIR filter are in Fig. 3.

2.1.3. Filtering along the scan-angle

To smooth out short term fluctuations of the radius measurements along the scan-angle, a moving average (MVA) filter is used:

$$r^{f}_{(\theta,t)} = r^{f}_{(\theta-d\theta,t)} - \frac{r_{(\theta-nd\theta,t)}}{n} + \frac{r_{(\theta,t)}}{n} \tag{2}$$

Where $r^{f}_{(\theta,t)}$ is the value of the MVA at angle θ and time t, $d\theta$ is the angular resolution of the scanner, $r^{f}_{(\theta-d\theta,t)}$ the value of the MVA at angle $\theta - d\theta$ and time t, and n is the number of averaging steps.

2.1.4. Calculating the minima and the position of the torso

The xy (floor plane) position of the torso capsule is calculated based on the position of the legs. The polar coordinates of the legs correspond to the minima in the plot. The minima can be calculated from peak analysis on the plot, Fig. 4, mirrored with respect to the axes of the scan-angle θ. Using the peak analysis, the angle and the radius associated with the first leg (θ_1, r_1) and the second leg

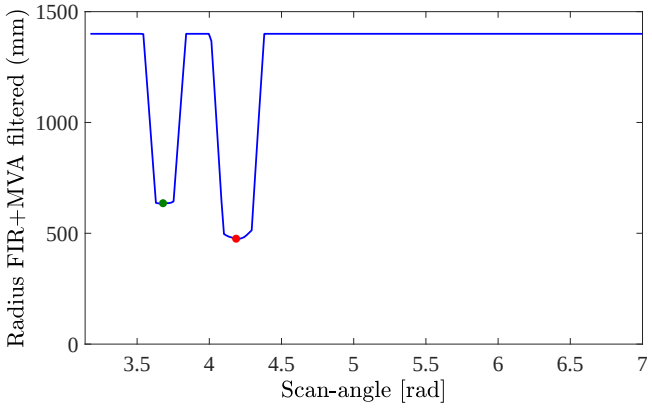

Figure 4: Filtered radius measurement with scan-angle. Minima are marked with green and red dots representing the human legs.

(θ_2, r_2) are acquired. Afterwards, the position of the first leg $\boldsymbol{x}_1$ in Cartesian space is calculated:

$$\boldsymbol{x}_1 = \begin{bmatrix} cos(\theta_1) \\ sin(\theta_1) \end{bmatrix} (r_1 + \rho_l) \tag{3}$$

Where ρ_l is the radius of the leg. The Cartesian position of the second leg $\boldsymbol{x}_2$ can be calculated in the same manner. The position of the torso capsule $\boldsymbol{x}_t$ can be approximately considered to be at the middle distance between the two legs:

$$\boldsymbol{x}_t = \frac{(\boldsymbol{x}_1 + \boldsymbol{x}_2)}{2} \tag{4}$$

Fig. 5 shows a laser scan with a human in the scan field of the sensor. The scan field span of 270 degrees is satisfactory given that the sensor is mounted at the corner of the robot table. Using the proposed algorithm the position of the legs is detected, red and green dots, and from the legs coordinates the torso position is approximated, black circle in Fig. 5.

2.2. Upper body configuration

To capture the configuration of the capsules representing the human upper body, an IMU sensor is attached to the chest (IMU 1) and the other four sensors are attached to the arms and the forearms (IMU 2, IMU 3, IMU 4 and IMU 5), Fig. 6. Each capsule is described by two vectors and a radius. The vectors represent the position of the beginning and end of each capsule.

The quaternion measurements provided by the IMU sensors coupled with the geometric information from the human co-workers body dimensions are used to estimate the position of the capsules covering the coworker's limbs in relation to robot base. The procedure for performing the calculations is divided into the following steps:

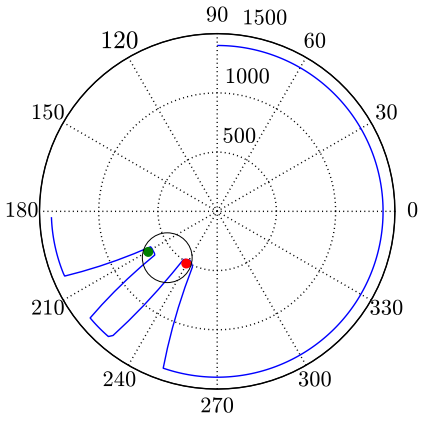

Figure 5: Human legs detected in the laser scan field. Red and green dot represent the legs and the black circle represents the torso.

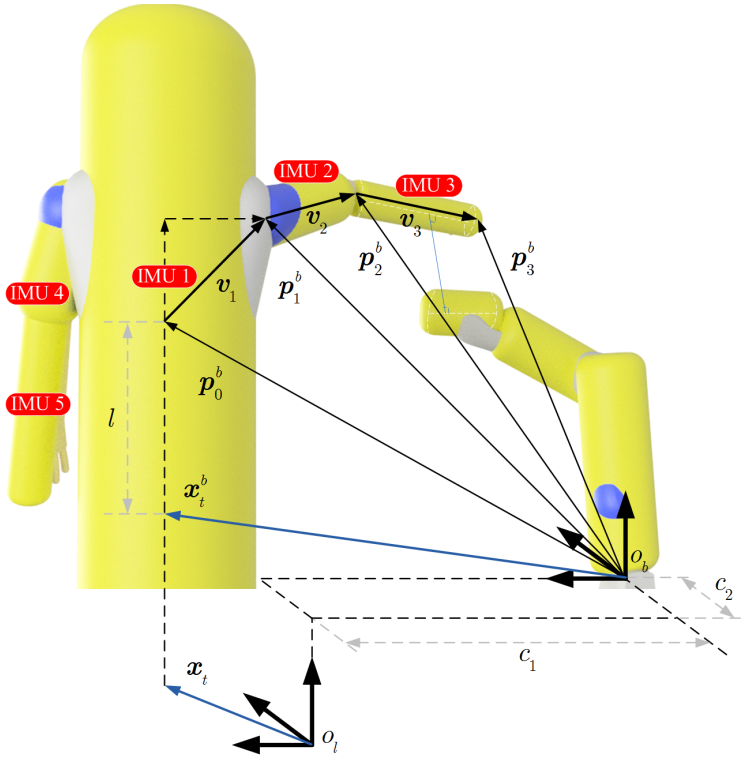

Figure 6: Minimum distance between two capsules.

1. Calibration phase;
2. Calculating rotations of limbs with respect to base frame of the robot;
3. Calculating the position of the limbs capsules with respect to the base frame of the robot.

2.2.1. Calibration

Each IMU measurement gives its orientation, in quaternion $\boldsymbol{w}_{imu}^{ref}$ [1], with respect to a pre-defined reference frame. For the collision avoidance algorithm the rotations shall be described in relation to the robot base frame using the quaternion $\boldsymbol{w}_{imu}^{b}$. To calculate $\boldsymbol{w}_{imu}^{b}$, the rotation quaternion from the reference frame to the robot base frame $\boldsymbol{w}_{ref}^{b}$ shall be calculated. This is achieved by performing an initial calibration phase. The IMUs are placed in a predefined orientation with respect to the robot base before the system is initiated. In such case the initial rotation quaternion $\boldsymbol{w}_{b}^{imu,init}$ of the IMU frame with respect to robot base is already known. By reading the initial IMU measurement $\boldsymbol{w}_{imu,init}^{ref}$, the quaternion $\boldsymbol{w}_{b}^{ref}$ is calculated:

$$\boldsymbol{w}_{b}^{ref} = \boldsymbol{w}_{imu,init}^{ref} \boldsymbol{w}_{b}^{imu,init} \tag{5}$$

As a result of the calibration phase, the rotation quaternion from the reference frame of the IMU to the base of the robot $\boldsymbol{w}_{ref}^{b}$ is calculated as the inverse of the quaternion $\boldsymbol{w}_{b}^{ref}$.

2.2.2. Limbs rotation

After calculating the orientation of the reference frame with respect to robot base $\boldsymbol{w}_{ref}^{b}$, the quaternion measurements describing the IMU orientation with respect to base frame of the robot $\boldsymbol{w}_{imu}^{b}$ is calculated from the orientation measurement of the IMU $\boldsymbol{w}_{imu}^{ref}$, as the following:

$$\boldsymbol{w}_{imu}^{b} = \boldsymbol{w}_{ref}^{b} \boldsymbol{w}_{imu}^{ref} \tag{6}$$

2.2.3. Limbs position

Five vectors on the human co-worker body are considered, Fig. 6. Owing to the symmetry of the human body, the following elaboration and equations are given for the right half of the co-worker's body. For the other half of the body identical methodology is applied. For the right half of the body three vectors are considered:

1. Vector $\boldsymbol{v}_3$: a vector that spans the length of the right forearm, from the elbow up to the wrist;
2. Vector $\boldsymbol{v}_2$: a vector that spans the right upper arm of the co-worker, from the shoulder up to the elbow;

[1] The superscript ref stands for reference frame, and the subscript imu stands for the frame of the IMU.

3. Vector $\boldsymbol{v}_1$: a vector that spans from the chest up to the shoulder.

The IMUs are mounted firmly on the body of the co-worker according to:

- The chest's IMU is mounted such that the x axis of the IMU is pointing vertically down when the co-worker is standing straight up. In such case the coordinates of the vector $\boldsymbol{v}_1^{imu_1}$ as described in upper chest's IMU frame are $[\ -d_1 \quad d_2 \quad 0\ ]$ for the left shoulder and $[\ -d_1 \quad -d_2 \quad 0\ ]$ for the right shoulder. d_2 is the width of the co-worker shoulder divided by two and d_1 is the length of co-worker's body taken vertically from the chest up to the shoulder.

- The upper arm's IMU is mounted such that the x axis of the IMU is aligned with the upper arm's length, in such case the coordinates of the vector $\boldsymbol{v}_2^{imu_2}$ as described in upper arm's IMU frame are $[\ d_3 \quad 0 \quad 0\ ]$. d_3 is the length of the upper arm measured from the shoulder to the elbow.

- The forearm's IMU is mounted such that the x axis of the IMU is aligned with the forearm's length. In such case the coordinates of the vector $\boldsymbol{v}_3^{imu_3}$ as described in forearm's IMU frame are $[\ d_4 \quad 0 \quad 0\ ]$. d_4 is the length of the forearm measured from the elbow to the wrist.

The previous vectors are rotated back to the base frame of the robot:

$$\boldsymbol{v}_i^b = \boldsymbol{w}_{imu_i}^b \boldsymbol{v}_i^{imu_i} \left(\boldsymbol{w}_{imu_i}^b\right)^{-1} \tag{7}$$

Where $\boldsymbol{v}_i^{imu_i}$, $i = 1, 2, ...5$, is the vector of human part described in the $i^{th} - imu$ frame. $\boldsymbol{w}_{imu_i}^b$ is the quaternion describing the rotation of $i^{th} - imu$ in relation to base frame of the robot Eq. (6) and $\left(\boldsymbol{w}_{imu_i}^b\right)^{-1}$ is the complex conjugate $\boldsymbol{w}_{imu_i}^b$.

The position vector of the shoulder point in the base frame of the robot $\boldsymbol{p}_1^b$ is calculated:

$$\boldsymbol{p}_1^b = \boldsymbol{v}_1^b + \boldsymbol{p}_0^b \tag{8}$$

Where $\boldsymbol{p}_0^b$ is the position of the chest point with respect to the base frame of the robot, given that the co-worker is standing up all the time the xy position of the co-worker's chest point is the same as the position of the torso acquired from the laser scanner. If we denoted l to the height of the co-worker's chest point from the xy plane of the base frame of the robot, then $\boldsymbol{p}_0^b$ is calculated:

$$\boldsymbol{p}_0^b = \begin{bmatrix} \boldsymbol{x}_t^b \\ l \end{bmatrix} \tag{9}$$

Where $\boldsymbol{x}_t^b$ is the xy position of the torso of the co-worker with respect to the base frame of the robot, calculated by transforming the torso position estimation (acquired from laser scanner measurement) $\boldsymbol{x}_t$ into the base frame of the robot. In this study the frame of the laser scanner is parallel to the base frame of the robot such that $\boldsymbol{x}_t^b$ is given by:

$$\boldsymbol{x}_t^b = \boldsymbol{x}_t + \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} \tag{10}$$

Where c_1 is the x coordinate of the origin O_l of the scanner's-measurement-frame in the base frame of the robot, and c_2 is the y coordinate of the origin O_l of the scanner's-measurement-frame in the base frame of the robot. The origin O_l and the dimensions (c_1, c_2) are shown in Fig. 6.

The position vector of the elbow point in the base frame of the robot $\boldsymbol{p}_2^b$ is given by:

$$\boldsymbol{p}_2^b = \boldsymbol{v}_2^b + \boldsymbol{p}_1^b \tag{11}$$

Accordingly, the position vector of the wrist point in the base frame of the robot p_3^b is:

$$\boldsymbol{p}_3^b = \boldsymbol{v}_3^b + \boldsymbol{p}_2^b \tag{12}$$

3. Minimum distance between capsules

The analytical minimum distance between capsules representing robot and human(s) is calculated recurring to QR factorization. The method is detailed in [35][36]. The process of calculating the minimum distance between two capsules is reduced to the calculation of the minimum distance between two line segments at the capsules axis. Each capsule can be defined by two vectors and a radius ρ. One vector defines the beginning of the capsule's axes-segment $\boldsymbol{p}$ and the other at the end of that capsule's axes-segment $\boldsymbol{u}$. After mathematical manipulation, the minimum distance is calculated from:

$$d_{min} = \sqrt{\boldsymbol{u}_{\boldsymbol{min}}^{\mathbf{T}} \boldsymbol{u}_{\boldsymbol{min}} + \boldsymbol{y}^{\mathbf{T}} \boldsymbol{y} - \boldsymbol{y}^{\mathbf{T}} \mathbf{Q} \mathbf{Q}^{\mathbf{T}} \boldsymbol{y}} - \rho_1 - \rho_2 \tag{13}$$

Where $\boldsymbol{u}_{min}$ is a 2×1 vector representing the point of the region of feasible solutions closest to the origin, $\mathbf{Q}$ is a 3×2 matrix whose column vectors are of unit length and mutually orthogonal, and ρ_1 and ρ_2 are the capsules radius. The QR method code is available in github[2].

The algorithm was implemented in MATLAB with a graphical user interface (GUI) to help to visualize the minimum distance between a human and a robot represented by capsules for any selected pose of both human and robot, Fig. 7.

4. Collision avoidance

Inspired by the potential fields (PF) method [24], we propose a customized version of the collision avoidance algorithm. Using this method the robot is

[2]https://github.com/Modi1987/Minimum-distance-between-capsules

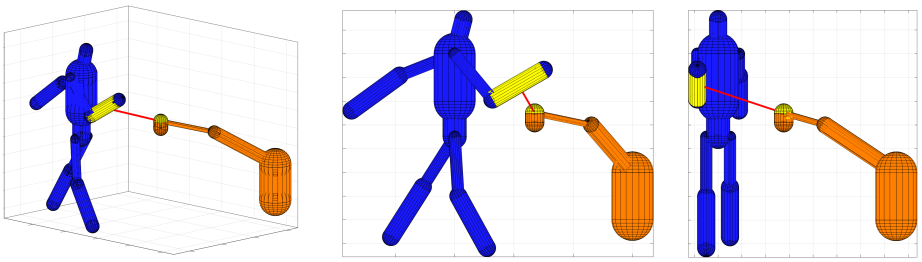

Figure 7: Human-robot minimum distance visualization in MATLAB GUI.

moving in a potential field, where the attraction vectors attract the end-effector towards the target and vectors of repulsion repel the robot away from obstacles. In this study the attraction vector acts on the end-effector and attracts it to the target. The target is the pre-established nominal path defined off-line in the robot teaching process (robot path considering that collision will not occur). This error vector is a function of the error:

$$\boldsymbol{e} = \boldsymbol{p}_e - \boldsymbol{p}_{target} \tag{14}$$

Where $\boldsymbol{e}$ is the error vector between the end-effector and the target point in the nominal path, $\boldsymbol{p}_e$ is the position vector of the end-effector (updated from the nominal path to avoid collision), and $\boldsymbol{p}_{target}$ is the position vector of the target (in the nominal path defined off-line). After calculating the error vector, an anti-windup [37] proportional integral controller Φ is utilized for calculating attraction vector $\boldsymbol{v}_{att}$:

$$\boldsymbol{v}_{att} = \Phi(K_p, K_i, \boldsymbol{e}) \tag{15}$$

Where K_p and K_i are the matrices of the proportional and integral coefficients. Using inverse kinematics methodologies [38], the angular velocities $\dot{\boldsymbol{q}}_{att}$ of the robot joints due to the attraction vector are calculated using the damped least squares [39]:

$$\dot{\boldsymbol{q}}_{att} = \mathbf{J}^T \left(\mathbf{J}\mathbf{J}^T + \lambda \mathbf{I}\right)^{-1} \boldsymbol{v}_{att} \tag{16}$$

Where $\mathbf{J}$ stands for the Jacobian of the robot associated with the Tool Center Point (TCP), the symbol T in the superscript stands for the matrix transpose operator, λ is a damping coefficient and $\mathbf{I}$ is the identity matrix.

For achieving collision avoidance, a repulsion vector shall act on the point of the robot closest to the obstacle/human. This vector repels the robot away from obstacles. The main input for calculating the repulsion vector is the minimum distance d_{min} between the obstacle and the robot, Eq. (13). After calculating the minimum distance, and the points of the robot and the obstacle associated with this minimum distance, the repulsion vector can be specified. The direction of the repulsion vector $\boldsymbol{s}$ is calculated:

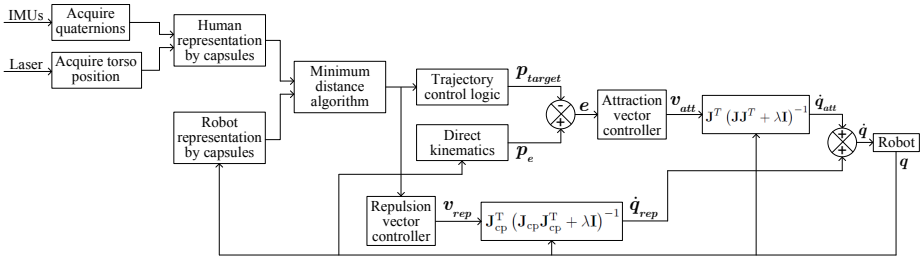

Figure 8: Collision avoidance general architecture.

$$\boldsymbol{s} = \frac{\boldsymbol{p}_r - \boldsymbol{p}_o}{|\boldsymbol{p}_r - \boldsymbol{p}_o|} \tag{17}$$

Where $\boldsymbol{p}_r$ is the position vector of the point of the robot closest to the obstacle and $\boldsymbol{p}_o$ is the point of the obstacle closest to the robot. After calculating the direction of the repulsion vector, its magnitude is calculated:

$$f_{rep} = \begin{cases} k_{rep}\left(\frac{d_o}{d_{min}-d_{cr}} - 1\right) & if\ d_{min} < d_o + d_{cr} \\ 0 & if\ d_{min} > d_o + d_{cr} \end{cases} \tag{18}$$

Where f_{rep} is the magnitude of the repulsive force, k_{rep} a repulsion constant, d_{cr} a critical distance below which the robot cannot be near the human, and $d_o + d_{cr}$ is the distance at which the repulsion vector is activated. The repulsion vector is calculated:

$$\boldsymbol{v}_{rep} = f_{rep}\boldsymbol{s} \tag{19}$$

And the angular velocities due to the repulsion vector are:

$$\dot{\boldsymbol{q}}_{rep} = \mathbf{J}_{\text{cp}}^{\text{T}}\left(\mathbf{J}_{\text{cp}}\mathbf{J}_{\text{cp}}^{\text{T}} + \lambda\mathbf{I}\right)^{-1}\boldsymbol{v}_{rep} \tag{20}$$

Where $\dot{\boldsymbol{q}}_{rep}$ are the angular velocities due to the repulsion vector and $\mathbf{J}_{\text{cp}}$ the Jacobian associated with the point of the robot closest to the obstacle. Then, the total angular velocities are calculated:

$$\dot{\boldsymbol{q}} = \dot{\boldsymbol{q}}_{att} + \dot{\boldsymbol{q}}_{rep} \tag{21}$$

Subjected to the angular velocities vector, the robot moves towards the target while avoiding collisions with the obstacles.

The architecture of the proposed approach is detailed in Fig. 8.

5. Experiments, results and discussion

The control algorithms were implemented in MATLAB and the interface with the KUKA iiwa was established using the KUKA Sunrise Toolbox provided

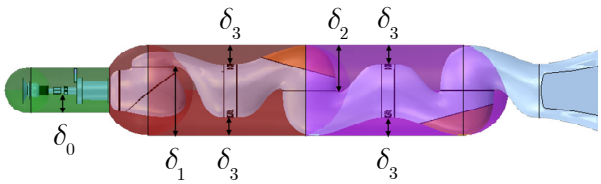

Figure 9: KUKA iiwa robot equiped with a suction-pad covered by three capsules.

Clearance values	
Dimension	Value (mm)
δ_0	45
δ_1	143
δ_2	124
δ_3	47

Table 1: Clearance values between the surface of the robot and the capsules covering it as shown in Fig. 9.

by us in GitHub [40]. The proposed collision avoidance framework was evaluated in a real world experiment using the collaborative manipulator KUKA iiwa 7 R800 with a pneumatic flange. Measurements from the sensors are interpolated with time considering the limitation of the slower device, the laser scanner, which provides data at 50 Hz. The IMUs provide data at about 300 Hz so that the implemented algorithms allow updating the robot state at a frequency of 275 Hz.

A pneumatic suction-pad is attached at the flange of the robot. Three capsules cover the links of the robot and the suction-pad under any configuration, Fig. 9. The upper arm and the lower arm of the robot are covered tightly with two identical capsules (height 400 mm and diameter 230 mm) and a third capsule is used to cover the suction-pad (height 250 mm and diameter 115 mm). The capsules fit the robot tightly, yet there is some error in the representation due to the deviation between the geometric surface of the robot and the surface of the capsules. To quantify those errors the dimensions are presented in Table 1.

In Experiment 1 the robot is operating in a pick-and-place operation, moving a box from a known location to another. The video that accompanies this article shows the experiment. Detailing, the experimental process is divided into 3 subtasks:

1. The human co-worker approaches the robot to place the box to be manipulated by the robot in a known pose. The robot is in a static home position

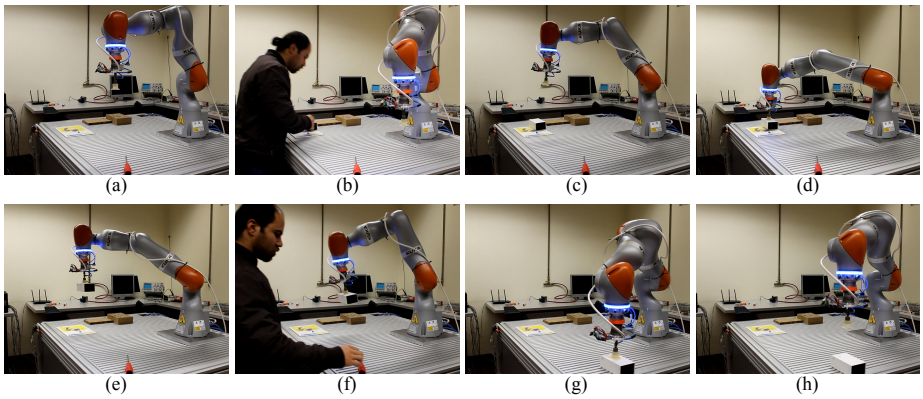

Figure 10: Experiment 1: the robot smoothly avoids collision when the human co-worker approaches and continues with the pick-and-place operation when the human goes away. This experiment is in the video that accompanies this article. 3D trajectories are shown in Fig. 12.

and smoothly reacts to avoid collision. When the human co-worker goes away, the robot automatically returns to the pre-planned path to pick up the box, Fig. 10;

2. The robot picks up the box and moves it to another location. During robot motion the human co-worker approaches the robot to pick up a tool on the top of the table. Again, the robot reacts to avoid collision, Fig. 10;
3. When the human goes way the robot continues the task and places the box in the desired position, Fig. 10.

When the human approaches the robot to place the box or to do any other task the robot automatically reacts by adjusting the pre-planned path (planned off-line) in what we call agile-smooth behaviour to avoid collision, Fig. 10. This means that the robot is agile to avoid collision when the human is at a given distance to the robot arm and presents a smooth behaviour as the human approaches the robot (the minimum distance decreases). When the human goes away the robot automatically continues its work, keeping the task target. If the robot is in a situation in which the collision is unavoidable the robot stops.

We conducted a quantitative analysis by recording the human-robot minimum distance, robot velocity and the robot end-effector position, Fig. 11. In Fig. 12, it is shown the paths of the Tool Center Point (TCP) of the end-effector, and the path of the torso of the human. At the beginning of the test the robot is stationary. When the human approaches the robot to place the box on the table the human-robot minimum distance decreases to a minimum of 431 mm. The robot velocity increases, maximum reached of 355 mm/sec, to compensate the human approach, smoothly reacting to avoid collision with the co-worker. When the human moves away the robot returns back to the pick-and-place operation. When the human approaches the robot again to pick up the tool the process is similar. Experimental tests also indicated that the system is well perceived

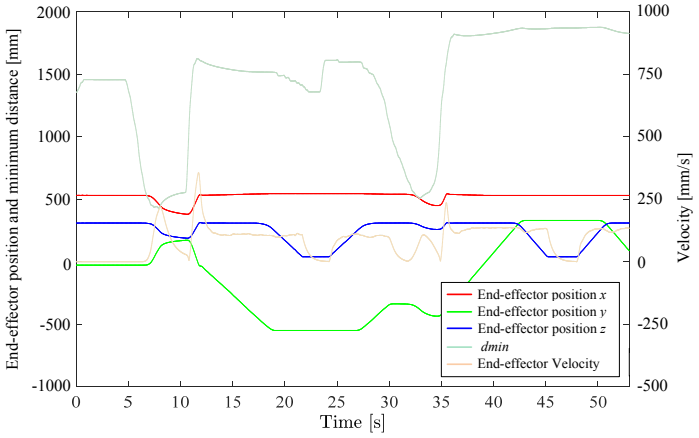

Figure 11: Human-robot minimum distance, robot end-effector position and velocity recorded during Experiment 1.

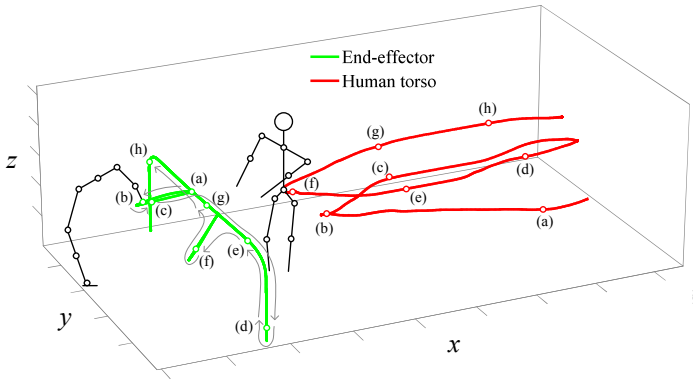

Figure 12: Robot end-effector path and the human torso trajectory related to expeirmental tests (Experiment 1) in Fig. 10, points (a), (b), (c), (d), (e), (f), (g) and (h). It is possible to visualize that the robot path is modified when the human approaches the robot and then the robot returns back to the nominal path. The human approaches the robot twice, to place the box on the table and to take the screwdriver from the table.

by the human. The collision avoidance motion is smooth and natural, so that the human does not perceive danger. The error from the representation of the human and robot by capsules exists. However, since the separation distance is higher than that error, it can be considered not problematic for the collision avoidance. The capsules length and radius have to be adapted to each different human and robot.

The proposed solution was also tested considering a human that passes in

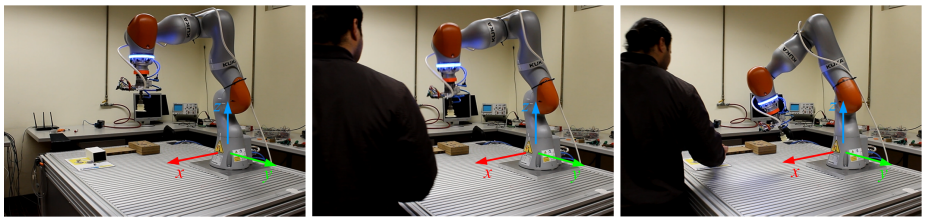

Figure 13: Experiment 2: the human approaching the robot from the side.

front of the robot (approaching the robot from the side) while the robot is working, Experiment 2, Fig. 13. The human will not directly interact with the robot but this is a common flexible manufacturing scenario where humans and robots share the workspace. In this context, the robot has to react to respect SSM and to avoid collision. At the beginning the robot is stationary in home configuration and the human is walking in the negative y direction towards the robot, Fig. 13. In such a case, the minimum distance decreases to a minimum of 393 mm and the robot reacts to avoid collision in which the maximum end-effector velocity reached is 548 mm/sec, Fig. 14. According to ISO/TS 15066, safety requirements for collaborative robots indicate a maximum robot velocity of 250 mm/sec. In our study, the robot reaches velocities superior to 250 mm/sec in both experiments. These velocity values are reached when the robot is moving away from the human to avoid collision. ISO/TS 15066 defines a velocity threshold not distinguishing if the robot is moving in the human direction or in the opposite direction to avoid collision. By defining 250 mm/sec as the maximum robot velocity the human may collide with the robot or at least the human has to reduce the walking and/or arms velocity. Experiments also demonstrate that the human-robot minimum distance is never less than 450 mm in Experiment 1 and 400 mm in Experiment 2. It is desirable that a future revision of ISO/TS 15066 can contemplate the direction of robot motion, i.e., if the robot is moving towards the human or if the robot is moving away from the human to avoid collision. If it is ensured that the robot is moving away from the human the maximum velocity allowed should be superior to 250 mm/sec. Moreover, such maximum velocity value should be defined according to the actual human-robot minimum distance.

The experiments demonstrated the following contributions:

1. Efficient representation of the human(s) upper body and robot using 5 and 3 capsules, respectively;
2. IMUs and laser scanner demonstrated accuracy and reliability to define each capsule pose in space and time. Error exists but is manageable for the collision avoidance success;
3. Agile-smooth robot reaction to collision avoidance in which the robot adapts the pre-established nominal paths (defined in the initial robot program off-line) while keeping the task target unchanged. The robot finds a way to get around the obstacles/humans and not to stop when they are

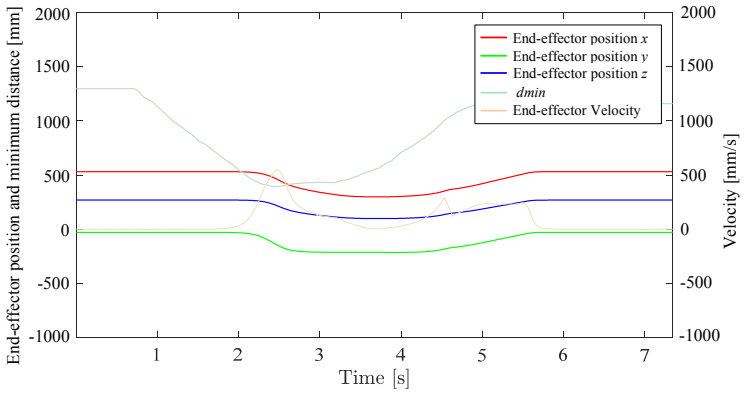

Figure 14: Human-robot minimum distance, robot end-effector position and velocity recorded during Experiment 2.

nearby;

4. Successfully testing with a real collaborative robot for a pick-and-place operation. According to our knowledge, until now, very few studies have implemented collision avoidance in real robot manipulators [19, 20, 16, 27], being this study a novel contribution in that domain.

6. Conclusion

This article successfully proposed utilizing a laser scanner and IMUs sensing technology for minimum distance calculation, an important input for the human-robot collaboration applications. The proposed methodology integrates into collision avoidance problem for collaborative robots sharing the space with humans. Humans and robots were successfully represented by capsules with data from a laser scanner and IMUs. QR factorization method was successfully applied to compute the minimum distance between capsules representing human and robot. A customized potential fields method that allows to adjust the pre-defined robot paths established off-line while keeping the task target was proposed for collision avoidance. Although the number of studies reporting collision avoidance with real collaborative robots while performing industrial tasks is low, the proposed framework was validated in real environment, using a real robot and sensors. Experiments demonstrated the robustness of the proposed approach in which the robot smoothly avoids collisions with the human co-worker while continues working keeping the task target.

7. 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).

8. References

[1] J. A. Marvel, R. Norcross, Implementing speed and separation monitoring in collaborative robot workcells, Robotics and Computer-Integrated Manufacturing 44 (2017) 144 – 155, ISSN 0736-5845, doi:https://doi.org/10.1016/j.rcim.2016.08.001, URL http://www.sciencedirect.com/science/article/pii/S0736584516302617.

[2] B. Matthias, S. Kock, H. Jerregard, M. Källman, I. Lundberg, Safety of collaborative industrial robots: Certification possibilities for a collaborative assembly robot concept, in: 2011 IEEE International Symposium on Assembly and Manufacturing (ISAM), ISSN Pending, 1–6, doi: 10.1109/ISAM.2011.5942307, 2011.

[3] J. Saenz, N. Elkmann, O. Gibaru, 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.

[4] H. Shin, K. Seo, S. Rhim, Allowable Maximum Safe Velocity Control based on Human-Robot Distance for Collaborative Robot, in: 2018 15th International Conference on Ubiquitous Robots (UR), 401–405, doi: 10.1109/URAI.2018.8441887, 2018.

[5] F. Vicentini, M. Giussani, L. M. Tosatti, Trajectory-dependent safe distances in human-robot interaction, in: Proceedings of the 2014 IEEE Emerging Technology and Factory Automation (ETFA), ISSN 1946-0740, 1–4, doi:10.1109/ETFA.2014.7005316, 2014.

[6] V. Rampa, F. Vicentini, S. Savazzi, N. Pedrocchi, M. Ioppolo, M. Giussani, Safe human-robot cooperation through sensor-less radio localization, in: 2014 12th IEEE International Conference on Industrial Informatics (INDIN), ISSN 1935-4576, 683–689, doi:10.1109/INDIN.2014.6945596, 2014.

[7] Y. Bai, H. Hu, Y. Li, C. Zhao, L. Luo, R. Wang, Research Methods for Human Activity Space Based on Vicon Motion Capture System, in: 2017 5th International Conference on Enterprise Systems (ES), 202–206, doi: 10.1109/ES.2017.40, 2017.

[8] P. Neto, N. Mendes, A. P. Moreira, Kalman filter-based yaw angle estimation by fusing inertial and magnetic sensing: a case study using low cost sensors, Sensor Review 35 (3) (2015) 244–250, doi:10.1108/SR-10-2014-0723, URL https://doi.org/10.1108/SR-10-2014-0723.

[9] D. Roetenberg, H. Luinge, P. Slycke, Xsens mvn: full 6dof human motion tracking using miniature inertial sensors. Xsens Motion Technologies BV, Tech. Rep., 2009.

[10] 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, J. L. Pons, Global Kalman filter approaches to estimate absolute angles of lower limb segments, BioMedical Engineering OnLine 16 (1) (2017) 58, ISSN 1475-925X, doi:10.1186/s12938-017-0346-7, URL https://doi.org/10.1186/s12938-017-0346-7.

[11] J. Ziegler, H. Kretzschmar, C. Stachniss, G. Grisetti, W. Burgard, Accurate human motion capture in large areas by combining IMU- and laser-based people tracking, in: 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems, ISSN 2153-0858, 86–91, doi:10.1109/IROS.2011.6094430, 2011.

[12] P. Bosscher, D. Hedman, Real-time collision avoidance algorithm for robotic manipulators, Industrial Robot: An International Journal 38 (2) (2011) 186–197.

[13] F. Ouyang, T. Zhang, Virtual Velocity Vector-Based Offline Collision-Free Path Planning of Industrial Robotic Manipulator, International Journal of Advanced Robotic Systems 12 (9) (2015) 129, doi:10.5772/60127.

[14] S. I. Choi, B. K. Kim, Obstacle avoidance control for redundant manipulators using collidability measure, Robotica 18 (02) (2000) 143–151.

[15] C. Fang, A. Rocchi, E. M. Hoffman, N. G. Tsagarakis, D. G. Caldwell, Efficient self-collision avoidance based on focus of interest for humanoid robots, in: Humanoid Robots (Humanoids), 2015 IEEE-RAS 15th International Conference on, 1060–1066, doi:10.1109/HUMANOIDS.2015.7363500, 2015.

[16] K. B. Kaldestad, S. Haddadin, R. Belder, G. Hovland, D. Anisi, et al., Collision avoidance with potential fields based on parallel processing of 3D-point cloud data on the GPU, in: Robotics and Automation (ICRA), 2014 IEEE International Conference on, IEEE, 3250–3257, 2014.

[17] R. V. Patel, F. Shadpey, Control of Redundant Robot Manipulators: Theory and Experiments, Springer Publishing Company, Incorporated, 1st edn., ISBN 3540250719, 9783540250715, 2009.

[18] A. Y. Afaghani, Y. Aiyama, On-line collision avoidance between two robot manipulators using collision map and simple Escaping method, in: Proceedings of the 2013 IEEE/SICE International Symposium on System Integration, SII 2013, Kobe, Japan, December 15-17, 2013, 105–110, doi:10.1109/SII.2013.6776637, 2013.

[19] B. Schmidt, L. Wang, Depth camera based collision avoidance via active robot control, Journal of Manufacturing Systems 33 (4) (2014) 711 – 718, ISSN 0278-6125, doi:https://doi.org/10.1016/j.jmsy.2014.04.004.

[20] F. Flacco, T. Kröger, A. De Luca, O. Khatib, A depth space approach to human-robot collision avoidance, in: Robotics and Automation (ICRA), 2012 IEEE International Conference on, IEEE, 338–345, 2012.

[21] J. Kim, W. Chung, Localization of a Mobile Robot Using a Laser Range Finder in a Glass-Walled Environment, IEEE Transactions on Industrial Electronics 63 (6) (2016) 3616–3627, ISSN 0278-0046, doi: 10.1109/TIE.2016.2523460.

[22] L. Spinello, M. Luber, K. O. Arras, Tracking people in 3D using a bottom-up top-down detector, in: 2011 IEEE International Conference on Robotics and Automation, ISSN 1050-4729, 1304–1310, doi: 10.1109/ICRA.2011.5980085, 2011.

[23] E. J. Jung, J. H. Lee, B. J. Yi, J. Park, S. Yuta, S. T. Noh, Development of a Laser-Range-Finder-Based Human Tracking and Control Algorithm for a Marathoner Service Robot, IEEE/ASME Transactions on Mechatronics 19 (6) (2014) 1963–1976, ISSN 1083-4435, doi: 10.1109/TMECH.2013.2294180.

[24] O. Khatib, Real-time obstacle avoidance for manipulators and mobile robots, The international journal of robotics research 5 (1) (1986) 90–98.

[25] F. Flacco, A. D. Luca, O. Khatib, Control of Redundant Robots Under Hard Joint Constraints: Saturation in the Null Space, IEEE Transactions on Robotics 31 (3) (2015) 637–654, ISSN 1552-3098, doi: 10.1109/TRO.2015.2418582.

[26] D. H. Park, H. Hoffmann, P. Pastor, S. Schaal, Movement reproduction and obstacle avoidance with dynamic movement primitives and potential fields, in: Humanoids 2008 - 8th IEEE-RAS International Conference on Humanoid Robots, ISSN 2164-0572, 91–98, doi:10.1109/ICHR.2008.4755937, 2008.

[27] S. Haddadin, H. Urbanek, S. Parusel, D. Burschka, J. Robmann, A. Albu-Schaffer, G. Hirzinger, Real-time reactive motion generation based on variable attractor dynamics and shaped velocities, in: 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems, ISSN 2153-0858, 3109–3116, doi:10.1109/IROS.2010.5650246, 2010.

[28] A. Mohammed, B. Schmidt, L. Wang, Active collision avoidance for human-robot collaboration driven by vision sensors, International Journal of Computer Integrated Manufacturing 30 (9) (2017) 970–980, doi:10.1080/0951192X.2016.1268269, URL https://doi.org/10.1080/0951192X.2016.1268269.

[29] M. Duguleana, F. G. Barbuceanu, A. Teirelbar, G. Mogan, Obstacle avoidance of redundant manipulators using neural networks based reinforcement learning, Robotics and Computer-Integrated Manufacturing 28 (2) (2012) 132 – 146, ISSN

0736-5845, doi:https://doi.org/10.1016/j.rcim.2011.07.004, URL http://www.sciencedirect.com/science/article/pii/S0736584511000962.

[30] G. S. Chyan, S. Ponnambalam, Obstacle avoidance control of redundant robots using variants of particle swarm optimization, Robotics and Computer-Integrated Manufacturing 28 (2) (2012) 147 – 153, ISSN 0736-5845, doi:https://doi.org/10.1016/j.rcim.2011.08.001, URL http://www.sciencedirect.com/science/article/pii/S0736584511000974.

[31] D. Han, H. Nie, J. Chen, M. Chen, Dynamic obstacle avoidance for manipulators using distance calculation and discrete detection, Robotics and Computer-Integrated Manufacturing 49 (2018) 98 – 104, ISSN 0736-5845, doi:https://doi.org/10.1016/j.rcim.2017.05.013, URL http://www.sciencedirect.com/science/article/pii/S0736584516303957.

[32] A. Y. Afaghani, Y. Aiyama, On-line collision detection of n-robot industrial manipulators using advanced collision map, in: Advanced Robotics (ICAR), 2015 International Conference on, IEEE, 422–427, 2015.

[33] J. Zhou, K. Nagase, S. Kimura, Y. Aiyama, Collision avoidance of two manipulators using RT-Middleware, in: System Integration (SII), 2011 IEEE/SICE International Symposium on, IEEE, 1031–1036, 2011.

[34] F. Seto, K. Kosuge, Y. Hirata, Self-collision Avoidance Motion Control for Human Robot Cooperation System using RoBE .

[35] M. Safeea, P. Neto, R. Bearee, Efficient Calculation of Minimum Distance Between Capsules and its use in Robotics, IEEE Access (2018) 1–1ISSN 2169-3536, doi:10.1109/ACCESS.2018.2889311.

[36] M. Safeea, N. Mendes, P. Neto, Minimum Distance Calculation for Safe Human Robot Interaction, Procedia Manufacturing 11 (2017) 99–106, doi:10.1016/j.promfg.2017.07.157.

[37] K. J. Åström, T. Hägglund, PID controllers: theory, design, and tuning, vol. 2, Instrument society of America Research Triangle Park, NC, 1995.

[38] 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) (2004) 16.

[39] Y. Nakamura, H. Hanafusa, Inverse Kinematic Solutions With Singularity Robustness for Robot Manipulator Control, Journal of Dynamic Systems, Measurement, and Control 108 (3) (1986) 163–171.

[40] M. Safeea, P. Neto, KUKA Sunrise Toolbox: Interfacing Collaborative Robots with MATLAB, IEEE Robotics Automation Magazine (2018) 1–1ISSN 1070-9932, doi:10.1109/MRA.2018.2877776.