2023 IEEE International Conference on Robotics and Automation (ICRA 2023)
May 29 - June 2, 2023. London, UK

Enforcing Constraints for Dynamic Obstacle Avoidance by Compliant Robots

Leonidas Koutras, Konstantinos Vlachos, George S. Kanakis, Fotios Dimeas,
Zoe Doulgeri and George A. Rovithakis

Abstract— **In this work a control scheme is proposed to enforce dynamic obstacle avoidance constraints to the full body of actively compliant robots. We argue that both compliance and accuracy are necessary to build safe collaborative robotic systems; obstacle avoidance is usually not enough, due to the reliance on perception systems which exhibit delays and errors. Our scheme is able to successfully avoid obstacles, while remaining compliant in the entirety of the executed task. Therefore, in case of unexpected collisions due to perception system errors, the robot remains safe for humans and its environment. Our approach is validated through experiments with simulated and real obstacles utilizing a 7-dof KUKA LBR iiwa robotic manipulator.**

I. INTRODUCTION

Modern robots tend to leave traditional structured environments of heavy industry and enter dynamic environments of human everyday-life [1], [2]. In such environments the robot is operating alongside humans in common tasks. Therefore it is essential for these robots to be safe for the humans in their workspace, as well as for themselves and their environment.

Safety is related to obstacle collision avoidance. Obstacle avoidance techniques have been studied for many years [3]–[6], with many new methods being published even in recent years [7]–[10]. They usually modify the trajectory of the robot on-line to avoid obstacles under the assumption of a robot that can accurately track the modified trajectory. Hence, in the majority of cases, a stiff robot is assumed. The effectiveness of these methods relies on the accuracy, processing speed and reliability of the robot's perception system. However, perception systems are prone to errors and delays. These errors can lead to detecting an obstacle further away from where it actually is, making a collision possible. Furthermore, actual robotic systems have torque and velocity limitations, which hinder successful obstacle avoidance. Hence, collisions may occur in practice. Therefore, it is important for the robot to be compliant in case of collisions.

Compliant robots can safely interact with their environment [11]. Active compliance control strategies such as impedance [12] and admittance [13] control realize closed loop robot dynamics in the form of a mass-spring-damper relation between the robot and an external contact making the robot safe to collisions and able to interact [14]. In these methods, unmodelled dynamics and task uncertainties affect tracking accuracy particularly in highly compliant or low stiffness robots. This issue has been studied in our previous work [15] for the case of Cartesian position control and a method has been proposed to achieve accurate tracking with compliant robots. It uses Dynamic Movement Primitives (DMP) [16]–[18] to generate a trajectory which in case of collisions pauses, waiting for the external contact force to disappear. In [15] the robot is under impedance control with low stiffness throughout the execution of the task and therefore, it remains safe during unexpected collisions. Nevertheless, low stiffness does not compromise the accuracy of the executed task even in the case of unmodelled dynamics and modelling errors.

There have been attempts in combining obstacle avoidance with compliance. In [19], an approach is proposed which utilizes proximity perception accompanied by variable impedance. This approach is heavily reliant on proximity sensing and could be dangerous in case of errors and failures. In [20], a variable stiffness approach is introduced which however utilizes specialized hardware. This approach adjusts the stiffness based on the obstacle perception, therefore it is also susceptible to perception failures. In [21], a Hierarchical Quadratic Programming approach is proposed, where the robot alternates between stiff and compliant behavior upon the collision detection, which is based on the error between the current and desired joint positions. All the above methods utilize transitions from stiff to compliant robot motion. They rely either on the perception system to initiate the transition, or on the error between the desired and actual behavior. The first case suffers from over-reliance on perception which often has errors and delays. In the second case, the robot is stiff at the moment of collision, therefore it is potentially dangerous when transitions do not occur instantly.

In this work we address the issue of avoiding the full body of actively compliant robots to collide with moving obstacles. This is achieved by utilizing a dynamic active constraint enforcement method [7] to augment the method of [15]. In contrast to other methods, in our approach the robot remains compliant during the execution of the entire task and therefore safe during unexpected collisions that could be attributed to perception failures. The controller of [15] is utilized to guarantee desired tracking accuracy with the low stiffness robot despite unmodelled dynamics and task uncertainties. The proposed approach is validated via experiments performed with virtual and real obstacles

Authors are with the Automation & Robotics Lab, Dept. of Electrical & Computer Engineering, Aristotle University of Thessaloniki, Greece {kleonidas, konstanvl, gkanakis, dimeasf, doulgeri, rovithak}@ece.auth.gr

The research leading to these results has received funding by the EU Horizon 2020 Research and Innovation Programme under grant agreement No 820767, project CoLLaboratE.

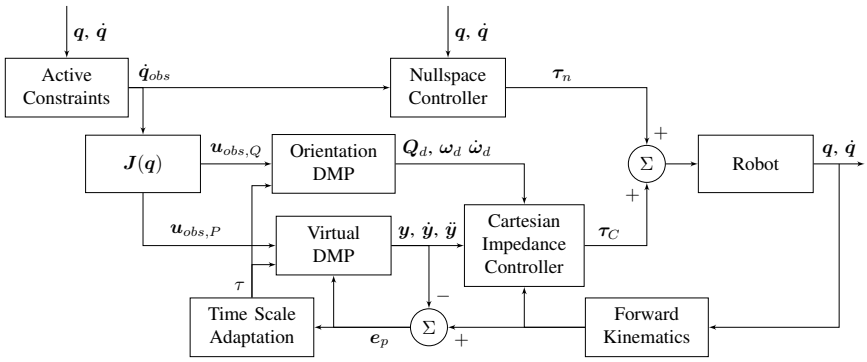


Fig. 1: Architecture of the proposed control scheme.

utilizing a 7-dof KUKA LBR iiwa robotic manipulator.

II. PROPOSED METHOD

The basic idea of the proposed approach is to avoid moving obstacles by modifying the operational space trajectory of a compliant robot as well as by inducing motion in the task nullspace to fully exploit the redundant degrees of freedom. The robot is compliant and thus safe in case of collision. A block diagram of the proposed solution is shown in Figure 1. In the heart of this approach is a low stiffness impedance controlled robot driven by a reference position trajectory $\boldsymbol{y}$, $\dot{\boldsymbol{y}}\ \ddot{\boldsymbol{y}} \in \mathbb{R}^3$ produced by a virtual DMP system achieving high tracking accuracy in the operational space as shown in our previous work [15]. A brief presentation of DMP is given in the Appendix. In this work, the virtual DMP proposed in [15] is augmented by the obstacle avoidance coupling term $\boldsymbol{v}_{obs,P}$. In particular, the virtual DMP transformation system is given by:

$$\begin{aligned} \tau \dot{\boldsymbol{z}}_P &= a_z(\beta_z(\boldsymbol{g}-\boldsymbol{y}) - \boldsymbol{z}_P - \boldsymbol{e}_p) + \boldsymbol{S}(\boldsymbol{g}-\boldsymbol{y_0})\boldsymbol{F}_P(x) \\ \tau \dot{\boldsymbol{y}} &= \boldsymbol{z}_P + \boldsymbol{v}_{obs,P} \end{aligned} \tag{1}$$

where $\boldsymbol{e}_p = \boldsymbol{p} - \boldsymbol{y} \in \mathbb{R}^3$ is the position tracking error with $\boldsymbol{p} \in \mathbb{R}^3$ being the end-effector position, $\boldsymbol{g}, \boldsymbol{y}_0 \in \mathbb{R}^3$ the goal and initial position of the movement, $\boldsymbol{z}_P \in \mathbb{R}^3$ the scaled velocity and $\boldsymbol{S}(\boldsymbol{g}-\boldsymbol{y}_0) \in \mathbb{R}^{3\times 3}$ a spatial scaling matrix. In the original DMP formulation, this matrix is given by $\boldsymbol{S}(\boldsymbol{g}-\boldsymbol{y}_0) = \mathbf{diag}(\boldsymbol{g}-\boldsymbol{y}_0)\,\mathbf{diag}(\boldsymbol{g}_{demo}-\boldsymbol{y}_{0,demo})^{-1}$, with $\mathbf{diag}(\cdot)$ denoting a diagonal matrix with elements the components of its argument vector and $\boldsymbol{g}_{demo}$, $\boldsymbol{y}_{0,demo}$ denoting the goal and initial position of the demonstrated trajectory. In this work we utilize the spatial scaling method proposed in [22], where $\boldsymbol{S}(\boldsymbol{g}-\boldsymbol{y}_0) = s_g \boldsymbol{R}_g$, with $s_g = \frac{\|\boldsymbol{g}-\boldsymbol{y}_0\|}{\|\boldsymbol{g}_{demo}-\boldsymbol{y}_{0,demo}\|}$ and $\boldsymbol{R}_g \in \mathcal{SO}(3)$ a rotation matrix that rotates $\boldsymbol{g}_{demo}-\boldsymbol{y}_{0,demo}$ to $\boldsymbol{g}-\boldsymbol{y}_0$ about an appropriately selected axis. The forcing term $\boldsymbol{F}_P(x)$ encodes the demonstrated trajectory given in terms of $\boldsymbol{y}_{demo}$, $\dot{\boldsymbol{y}}_{demo}$, $\ddot{\boldsymbol{y}}_{demo} \in \mathbb{R}^3$. For the orientation DMP, we utilize the formulation proposed in [23] augmented by the respective obstacle avoidance coupling term $\boldsymbol{v}_{obs,Q}$:

$$\begin{aligned} \tau \dot{\boldsymbol{z}}_Q &= -\alpha_z(\beta_z \boldsymbol{e}_Q + \boldsymbol{z}_Q) + \boldsymbol{S}(\boldsymbol{e}_{Q,0})\boldsymbol{F}_Q(x) \\ \tau \dot{\boldsymbol{e}}_Q &= \boldsymbol{z}_Q - \boldsymbol{v}_{obs,Q} \end{aligned} \tag{2}$$

where $\boldsymbol{e}_Q = 2\log(\boldsymbol{Q}_g * \overline{\boldsymbol{Q}}_d) \in \mathbb{R}^3$ is the orientation error with $\boldsymbol{Q}_d$, $\boldsymbol{Q}_g$ being the reference orientation provided to the low stiffness impedance controller and the goal orientation expressed in unit quaternions, $\overline{\boldsymbol{Q}}$ the quaternion conjugate, $\boldsymbol{z}_Q \in \mathbb{R}^3$ the scaled velocity, $\boldsymbol{S}(\boldsymbol{e}_{Q,0}) \in \mathbb{R}^{3\times 3}$ with $\boldsymbol{e}_{Q,0} = 2\log(\boldsymbol{Q}_g * \overline{\boldsymbol{Q}}_0)$ is the orientation spatial scaling matrix defined as in the position DMP and $\boldsymbol{F}_Q(x) \in \mathbb{R}^3$ the respective orientation forcing term. Notice that the minus sign in the last equation of (2) is owed to the utilization of the error $\boldsymbol{e}_Q$. Both systems (1) and (2) have a common phase variable governed by:

$$\tau \dot{x} = -\alpha_x x \tag{3}$$

with its initial value $x_0 = 1$ and α_x a positive gain. A sigmoid time scaling adaptation law as in [15] is utilized:

$$\tau = \tau_0 \left(1 + \exp\left(a_{sig}(\|\boldsymbol{e}_p\| - c_{sig})\right)\right) \tag{4}$$

with $\tau_0 > 0$ being the desired temporal scaling parameter and $a_{sig}, c_{sig} \in \mathbb{R}^+$ being positive parameters that regulate the slope and the center of the sigmoid function. Notice that in contrast to previously published works [18], which utilize the square norm of the tracking error $\boldsymbol{e}_p$, the adaptive temporal scaling parameter τ becomes equal to the desired τ_0 for high slope values a_{sig} and $||\boldsymbol{e}_p|| < c_{sig}$. This implies that there is a region of the tracking error in which the execution of the DMP does not slow down. For $||\boldsymbol{e}_p|| > c_{sig}$ the adaptive temporal scaling parameter τ abruptly increases stopping the evolution of the system (1), (2), (3). As a result the reference trajectory becomes a constant reference input. Thus, the whole control structure in this case is reduced to an impedance controlled robot driven by external forces, like a regulation case, with its compliance ensured by choosing low-impedance parameters (stiffness, damping gains).

The DMP system is coupled with the robot control system which realizes a target impedance in terms of inertia, damping and stiffness $\boldsymbol{M_d}, \boldsymbol{D_d}, \boldsymbol{K_d}$ with respect to the pose error $\begin{bmatrix} \boldsymbol{e}_p^T & \boldsymbol{e}_o^T \end{bmatrix}^T$.

$$\boldsymbol{M_d}\begin{bmatrix} \ddot{\boldsymbol{e}}_{\boldsymbol{p}} \\ \dot{\boldsymbol{\omega}}_{\boldsymbol{e}} \end{bmatrix} + \boldsymbol{D_d}\begin{bmatrix} \dot{\boldsymbol{e}}_{\boldsymbol{p}} \\ \boldsymbol{\omega}_{\boldsymbol{e}} \end{bmatrix} + \boldsymbol{K_d}\begin{bmatrix} \boldsymbol{e}_{\boldsymbol{p}} \\ \boldsymbol{e}_{\boldsymbol{o}} \end{bmatrix} = \boldsymbol{F}_e + \boldsymbol{F_{ext}} \tag{5}$$

where $\boldsymbol{e_o} = 2\log(\boldsymbol{Q_e}) \in \mathbb{R}^3$ is the orientation tracking error with $\boldsymbol{Q}_e = \boldsymbol{Q} * \overline{\boldsymbol{Q}}_d$ and $\boldsymbol{Q} \in \mathcal{S}^3$ being the orientation of the end-effector expressed as a unit quaternion. The angular velocity tracking error $\boldsymbol{\omega}_e = \boldsymbol{\omega} - \boldsymbol{\omega}_R \in \mathbb{R}^3$ with $\boldsymbol{\omega} \in \mathbb{R}^3$ being the angular velocity of the end-effector,

is defined with respect to $\boldsymbol{\omega}_R = \boldsymbol{R}_e\boldsymbol{\omega}_d \in \mathbb{R}^3$, which is the geometrically consistent reference angular velocity [24], [25], with $\boldsymbol{\omega}_d \in \mathbb{R}^3$ being the reference angular velocity and with $\boldsymbol{R}_e \in \mathcal{SO}(3)$ being the rotation matrix corresponding to $\boldsymbol{Q}_e$. The term $\boldsymbol{F}_e(t) \in \mathbb{R}^6$ is a disturbance input arising from unmodelled dynamics like joint friction, robot dynamic parametric as well as task uncertainties (e.g load) and external force measurement inaccuracies. The term $\boldsymbol{F}_{ext} \in \mathbb{R}^6$ denotes external contact forces. For the case of free motion ($\boldsymbol{F}_{ext} = \boldsymbol{0}$) and in the ideal case of a zero disturbance input $\boldsymbol{F}_e(t)$ the robot tracks perfectly the reference trajectory. However $\boldsymbol{F}_e(t)$ exists in practice. The utilization of disturbance observers [26] or data driven model predictive control setups [27] can decrease its magnitude but a residual disturbance term usually remains in practice due to estimation errors and the solution's computational cost is increased. The robot performance in terms of accurately tracking the reference trajectory depends on the stiffness and damping gains of the impedance controller. In this work, high stiffness and damping gains are utilized to reduce the effect of disturbances to the orientation tracking error. In contrast low stiffness and damping gains are selected for the position part of the impedance controller to ensure safety under unexpected collisions. Hence position tracking errors ($\boldsymbol{e}_p$) are expected. However as detailed in [15] by incorporating $\boldsymbol{e}_p$ in (1) the generated reference position trajectory compensates for such errors thus achieving position tracking accuracy of the desired trajectory.

Obstacle avoidance is achieved by imposing active constraints, which generate a joint velocity signal $\dot{\boldsymbol{q}}_{obs} \in \mathbb{R}^n$. This velocity is decomposed into two components; one that affects the motion in the operational space and is utilized in the DMP system as a coupling term in (1), (2) to modify the generated reference trajectory and one that affects motion in the task nullspace. The coupling term can be derived by:

$$\boldsymbol{v}_{obs} = \begin{bmatrix} \boldsymbol{v}_{obs,P} \\ \boldsymbol{v}_{obs,Q} \end{bmatrix} = \boldsymbol{J}(\boldsymbol{q})\dot{\boldsymbol{q}}_{obs} \tag{6}$$

where $\boldsymbol{J}(\boldsymbol{q})$ is the robot Jacobian.

The nullspace component is utilized in a decoupled nullspace controller inspired by [12], [28] which is applied directly at the torque level (Figure 1).The torque applied by the nullspace controller augments the impedance torque input that imposes the dynamics of (5) and is given by:

$$\boldsymbol{\tau}_n = -\boldsymbol{N}^T\boldsymbol{D}_N\boldsymbol{N}(\dot{\boldsymbol{q}} - \dot{\boldsymbol{q}}_{obs}) \tag{7}$$

where $\boldsymbol{q}, \dot{\boldsymbol{q}} \in \mathbb{R}^n$ are the joint positions and velocities, $\boldsymbol{D}_N \in \mathbb{R}^{r\times r}$ is a diagonal damping matrix and $\boldsymbol{N}(q) \in \mathbb{R}^{r\times n}$ is the dynamically consistent nullspace projection matrix, with n being the degrees of freedom of the robot and r being the redundant degrees of freedom. $\boldsymbol{N}(q)$ is given by:

$$\boldsymbol{N}(\boldsymbol{q}) = \left(\boldsymbol{Z}(\boldsymbol{q})\boldsymbol{M}(\boldsymbol{q})\boldsymbol{Z}^T(\boldsymbol{q})\right)^{-1}\boldsymbol{Z}(\boldsymbol{q})\boldsymbol{M}(\boldsymbol{q}) \tag{8}$$

where $\boldsymbol{M}(\boldsymbol{q}) \in \mathbb{R}^{n\times n}$ is the inertia matrix of the robot and $\boldsymbol{Z}(\boldsymbol{q}) \in \mathbb{R}^{r\times n}$ is the nullspace base matrix, given by:

$$\boldsymbol{Z}(\boldsymbol{q}) = \begin{bmatrix} -\boldsymbol{J}_r^T(\boldsymbol{q})\boldsymbol{J}_m^{-T}(\boldsymbol{q}) & \boldsymbol{I}_r \end{bmatrix} \tag{9}$$

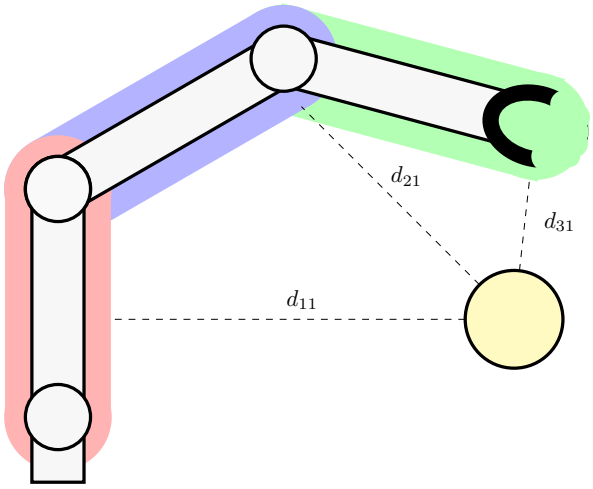


Fig. 2: Coverage of the body of a robotic manipulator by three capsules and distance of each capsule from a spherical obstacle.

where $\boldsymbol{I}_r$ is the identity matrix of dimension r and matrices $\boldsymbol{J}_m$ and $\boldsymbol{J}_r$ can be computed by writing the Jacobian of the manipulator as:

$$\boldsymbol{J}(\boldsymbol{q}) = \begin{bmatrix} \boldsymbol{J}_m(\boldsymbol{q}) & \boldsymbol{J}_r(\boldsymbol{q}) \end{bmatrix} \tag{10}$$

where $\boldsymbol{J}_m(\boldsymbol{q}) \in \mathbb{R}^{m\times m}$ is invertible away from singularities. Notice that the form of (10) may require the reordering of the columns of $\boldsymbol{J}(\boldsymbol{q})$ and that the nullspace matrix satisfies the equality $\boldsymbol{J}(\boldsymbol{q})\boldsymbol{Z}^T = \boldsymbol{0}$.

Since the nullspace component for obstacle avoidance does not interfere with the executed task it cannot induce any errors that would in any way affect the DMP temporal scaling parameter or halt the time evolution of the reference trajectory.

A. Generation of obstacle avoidance velocity signal

To generate the obstacle avoidance joint velocity $\dot{\boldsymbol{q}}_{obs}$ we utilize a generalized version of the method developed in [7] that ensures the enforcement of dynamic active constraints in the context of moving obstacles described by enclosing convex surfaces. Specifically, the controller is developed to guarantee that the end-effector of the robot avoids colliding with moving obstacles using a Cartesian velocity reference signal. To generalize this approach to the full body of the manipulator we utilize smooth enclosing surfaces, which completely cover the robot's body, as described in [29], where each link is modeled as a capsule. A capsule is consisted of two spheres with centers on the joints, and a cylinder whose axis connects the two spheres, covering the associating link. The radii of the spheres and the cylinder are equal. In Figure 2 an example of a manipulator covered by capsules is shown, as well the distance of each capsule from a spherical obstacle. Notice that the distance between a capsule and a point can be analytically computed.

Assume that the manipulator's body is modeled by M capsules and that N obstacles are in the workspace of the robot, modelled by convex surfaces. We denote by $d_{ij}(t)$ the

Euclidean distance between capsule i and obstacle j. We also define the harmonic mean $h(x_1, x_2, \ldots, x_n)$ as

$$h(x_1, x_2, \ldots, x_n) = \frac{n}{\sum_{i=1}^{n} \frac{1}{x_i}} \tag{11}$$

as well as the weighted harmonic mean $h_w(x_1, x_2, \ldots, x_n)$ as:

$$h_w(x_1, x_2, \ldots, x_n) = \frac{\sum_{i=1}^{n} w_i}{\sum_{i=1}^{n} \frac{w_i}{x_i}}. \tag{12}$$

The metric of constraint enforcement is:

$$m(t) = h(h_{w1}, h_{w2}, \ldots, h_{wN}) \tag{13}$$

where $h_{wi} = h_w(d_{i1}(t), d_{i2}(t), \ldots, d_{iM}(t))$.

The selection of the harmonic mean for combining distances is motivated by the fact that this metric emphasises more on small distances, making the robot more reactive to obstacles in its vicinity without however completely disregarding distant obstacles. In addition, the weighted harmonic mean is selected to prioritize the motion of each capsule depending on the executed task setting for example, a larger weight to the motion of the robot links closer to its end-effector than the links closer to its base. Notice that both the harmonic mean (11) and the weighted harmonic mean (12) are zero if one of their arguments is zero; hence, metric (13) becomes zero if and only if one of the distances becomes zero. Thus $m(t) > 0, \forall t$ is a sufficient condition for successful obstacle avoidance.

The obstacle avoidance joint velocity used in (6) and (7) is given by:

$$\dot{\boldsymbol{q}}_{obs} = -\kappa \frac{\epsilon}{1+\xi} \frac{1}{\|\dot{\boldsymbol{q}}_c\|^2} \dot{\boldsymbol{q}}_c \tag{14}$$

where κ is a positive gain, $\xi = \frac{m(t)-\rho}{\rho}$, $\epsilon := T(\xi) = \ln(1+\xi)$, ρ is a design parameter and $\dot{\boldsymbol{q}}_c = \left(\frac{\partial m(t)}{\partial \boldsymbol{q}}\right)^T$. This control input via the transformation of the error utilizing the diffeomorphism $T(\xi)$ guarantees that ϵ is bounded from below for all $t > 0$ and therefore $m(t) > 0$, ensuring that the robot does not collide with the obstacles.

Notice that self-collision avoidance can be incorporated in this method by augmenting the obstacle set with the link capsules. To ensure better performance, the distances between links that cannot collide due to joint limits should not be taken into consideration.

III. EXPERIMENTAL RESULTS

To validate the proposed approach, two sets of experiments are performed using a 7-dof KUKA LBR iiwa robot under gravity compensation of its own weight, driven at the torque level with $3ms$ control cycle utilizing an impedance control scheme without inertia shaping. The software runs on a real-time Linux PC, and the FRI library is used to command the robot. The different sub-modules are communicating using ROS. In the first set of experiments, we use simulated obstacles, moving in predefined trajectories in order to showcase specific parts of the proposed method. In the second set of experiments, we use a more complex setup from a real industrial assembly line regarding the placing of two PCBs on a TV chassis. In the latter set of experiments, humans are detected and modelled as dynamic obstacles by a perception system. The parameters used in both sets of experiments are given in Table I. Notice the low stiffness values in position for the robot controller, enabling its compliant behavior. To avoid obstacles by the whole robot body the robot is covered with six capsules, enclosing each link except the first one, which can only rotate about its own axis. The two capsules closer to the shoulder are given weight one while the rest of them weight two. This selection favours movement of robot links away from the shoulder, which is preferable in our experimental setup, as humans are closer to the robot's wrist and end-effector. Both experiments with simulated and real obstacles can be seen in the accompanying video.

Parameter	Description	Value
α_z, β_z	DMP gains	30, 7.5
N	Number of kernels	80
$k_{d,P}, d_{d,P}$	Stiffness and damping in position	200, 35
$k_{d,O}, d_{d,O}$	Stiffness and damping in orientation	100, 20
c_{sig}, α_{sig}	Sigmoid center and slope	0.07, 100
d_N	Nullspace gain	1
κ, ρ	Obstacle avoidance parameters	8, 0.38

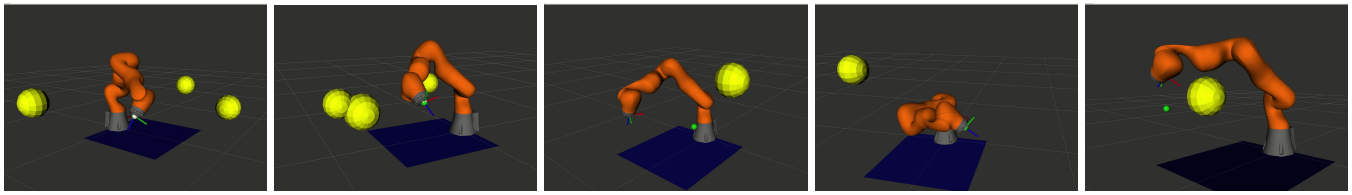
TABLE I: System parameters in experiments.

A. Simulated Obstacles

We performed three experiments with simulated obstacles. In all experiments, a planar obstacle is placed on the workbench where the robot stands, to ensure that the robot would not collide with it. Snapshots of all three experiments are shown in Figure 3. In all three experiments, the initial and goal pose of the robot are the same as the ones in the demonstrated data in order to facilitate comparison of the generated trajectories with the demonstrated one.

In the first scenario, three moving spherical obstacles are executing sinusoidal trajectories between two points in the workspace of the robot, while the robot is executing a desired trajectory, as shown in Figure 3a. In the first subplot of Figure 4 we can see the harmonic mean metric which stays above zero, achieving successful avoidance of all obstacles, despite the low gains of the impedance controller. Notice that after $t \approx 7s$ the metric is kept constant because all obstacles are then far away from the robot resulting in a zero obstacle avoidance signal. In Figure 5, the modified path due to the obstacle avoidance is shown as well as the demonstrated one. The dissimilarity between them is attributed to the trajectories of the obstacles, which are interfering with the path of the robot during the entire motion. The final configuration of the robot and the obstacles is depicted in Figure 3b.

For the second scenario, an obstacle is moving close to the elbow of the robot, as shown in Figure 3c. The second subplot of Figure 4 demonstrates successful avoidance by a movement mostly in the task nullspace. The demonstrated and modified path are shown in Figure 6. Since the obstacle is far away in the beginning of the robot motion we show results after $t = 8.8s$. In Figure 3d the final configuration is shown, where the task nullspace motion is evident.



(a) Scenario 1: $t = 0s$. **(b)** Scenario 1: $t = 8.3s$. **(c)** Scenario 2: $t = 0s$. **(d)** Scenario 2: $t = 16.8s$. **(e)** Scenario 3.

Fig. 3: Obstacle positions in all experiments.

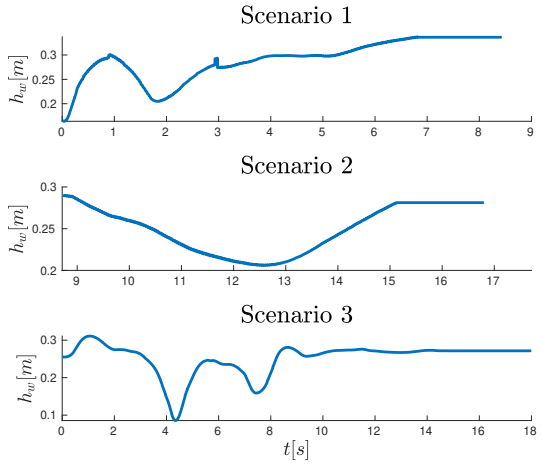


Fig. 4: Distance metrics for each experiment

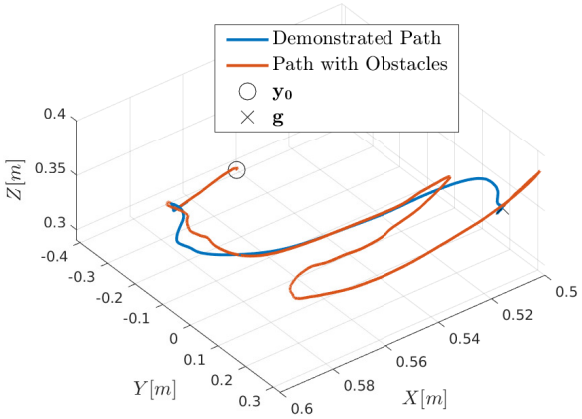


Fig. 6: Scenario 2: Demonstrated and modified path.

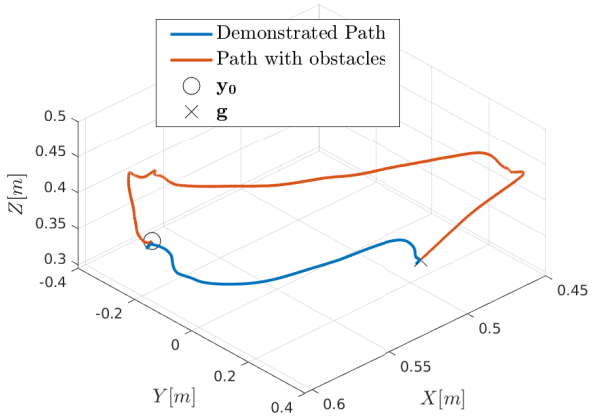


Fig. 5: Scenario 1: Demonstrated and modified path.

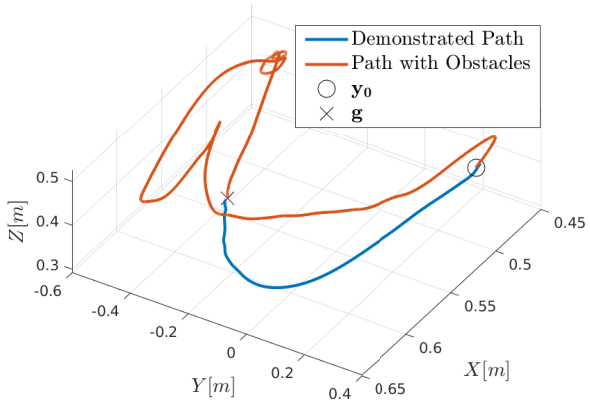


Fig. 7: Scenario 3: Demonstrated and modified path.

In the third scenario, we demonstrate the robot's compliance to collisions while the robot is avoiding a large stationary obstacle that is obstructing its nominal path, as shown in Figure 3e. A human enters the trajectory of the robot causing a collision. In practice, this can occur in case of the perception system fails to detect the human. After colliding, the human pushes the robot back before leaving the workspace of the robot, demonstrating its compliant nature. Figure 8, depicts the error $\boldsymbol{e}_p = \boldsymbol{p} - \boldsymbol{y}$ as well as the temporal scaling parameter τ, showing that indeed the system evolution stops during the unintentional contact. The path modification due to the human pushing the robot can be seen in the upper left of Figure 7 which shows the demonstrated and modified path. The large error seen in the first subplot of Figure 8 is also attributed to this pushing. In the third subplot of Figure 4 we can see that avoidance of the stationary obstacle is successful.

B. Real Obstacles

We have also tested the proposed approach in a complex lab setup, emulating an industrial environment. A sequence of DMP are trained to place two PCB boards on a static TV chassis. During execution, the TV is placed on a moving conveyor belt, and a human worker is responsible for screwing the boards after the robot has successfully placed them. Since the human and the robot operate in a shared

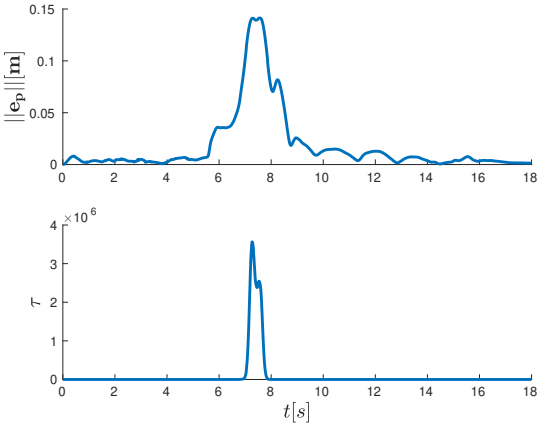
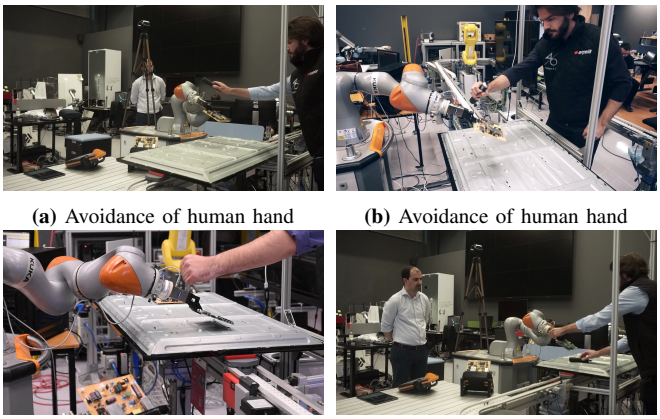


Fig. 8: Scenario 3: Error and temporal scaling parameter.



(a) Avoidance of human hand

(b) Avoidance of human hand

(c) Avoidance of human hand and TV chassis

(d) Compliance under contact with a human.

Fig. 9: Snapshots of experiments with real obstacles

workspace and in order to ensure safety, a perception system is used to locate and track the human skeleton. Spheres are appropriately placed to cover the worker's body, representing dynamic obstacles. The robot is under the proposed scheme, compliant throughout the task execution, therefore safe in case of collisions. To take into account the moving goal of the DMP, the approach developed in [30] is utilized, modifying (1) to get the following:

$$\begin{aligned} \tau \dot{\boldsymbol{z}}_P &= -\alpha_z(\beta_z \boldsymbol{e}_{P,G} + \boldsymbol{z}_P - \boldsymbol{e}_P) + \boldsymbol{S}(\boldsymbol{e}_{P,0})\boldsymbol{F}_P(x) \\ \tau \dot{\boldsymbol{e}}_{P,G} &= \boldsymbol{z}_P - \boldsymbol{v}_{obs,P} \end{aligned} \tag{15}$$

where $\boldsymbol{e}_{P,0} = \boldsymbol{g} - \boldsymbol{y}_0$ and the robot reference is given as $\boldsymbol{y} = \boldsymbol{g} - \boldsymbol{e}_{P,G}$.

As shown in the accompanying video the robot is able to avoid the human in all cases despite measurement noise and limitations on the accuracy of the perception system. Furthermore, due to its low stiffness it remains safe during the interaction with the human. Snapshots are shown in Figure 9. Figures 9a, 9b and 9c demonstrate the robot avoiding the human in different executions. To showcase that the robot indeed remains compliant in case of collisions, the obstacle surrounding the human's wrist is deactivated, while the other obstacles where still active. Figure 9d depicts this case.

IV. CONCLUSIONS

A control scheme is proposed and successfully implemented that enforces dynamic obstacle avoidance constraints to the full body of actively compliant robots. The control scheme modifies the operational space trajectory of the end-effector and induces additional motion in the nullspace of the robot to exploit the robot's redundancy in order to achieve accuracy, obstacle avoidance and safety to unexpected collisions. Experimental results with a 7-dof KUKA LBR iiwa robot show successful obstacle avoidance with multiple virtual moving obstacles as well as compliance during the entire task execution. The same results are successfully demonstrated in a complex industrial assembly line environment.

APPENDIX: DYNAMIC MOVEMENT PRIMITIVES

Dynamic Movement Primitives (DMP) [16]–[18] have been proposed to encode arbitrary trajectories in the Cartesian space by augmenting a linear dynamical system by a non linear forcing term:

$$\begin{aligned} \tau \dot{\boldsymbol{z}} &= \alpha_z(\beta_z(\boldsymbol{g} - \boldsymbol{y}) - \boldsymbol{z}) + \boldsymbol{S}(\boldsymbol{g} - \boldsymbol{y}_0)\boldsymbol{F}(x) \\ \tau \dot{\boldsymbol{y}} &= \boldsymbol{z} \end{aligned} \tag{16}$$

where α_z, β_z are positive gains, $\boldsymbol{y}, \boldsymbol{z} \in \mathbb{R}^3$ are the position and scaled velocity, $\tau > 0$ is a temporal scaling scalar, $\boldsymbol{g}, \boldsymbol{y}_0 \in \mathbb{R}^3$ are the goal and initial position and $\boldsymbol{S}(\boldsymbol{g} - \boldsymbol{y}_0) \in \mathbb{R}^{3\times 3}$ is a spatial scaling matrix. This set of equations is called the transformation system of the DMP.

The desired trajectory is encoded in the forcing term $\boldsymbol{F}(x)$:

$$\boldsymbol{F}(x) = \frac{\sum_{i=1}^{N} \boldsymbol{w}_i \Psi_i(x)}{\sum_{i=1}^{N} \Psi_i(x)} x \tag{17}$$

where $\Psi_i(x) = \exp\left(-h_i(c_i - x)^2\right)$ are N Gaussian kernel functions, with h_i, c_i denoting their inverse widths and centers. The weights $\boldsymbol{w}_i \in \mathbb{R}^3$ are learned using Locally Weighted Regression to encode the desired trajectory. To avoid explicit time dependency the canonical system of the phase variable x is utilized as a clocking mechanism governed by (3).

To encode orientation trajectories, the majority of the literature utilizes unit quaternions [31]. Early works in encoding orientation trajectories with DMP can be found in [32]. A more robust approach that has been proposed in [23] employs the transformation system given by:

$$\begin{aligned} \tau \dot{\boldsymbol{z}} &= -\alpha_z(\beta_z \boldsymbol{e}_Q + \boldsymbol{z}) + \boldsymbol{S}(\boldsymbol{e}_{Q,0})\boldsymbol{F}(x) \\ \tau \dot{\boldsymbol{e}}_Q &= \boldsymbol{z} \end{aligned} \tag{18}$$

where $\boldsymbol{e}_Q$ is the orientation error, $\boldsymbol{z}$ its velocity scaled by τ, $\boldsymbol{Q} = \begin{bmatrix}\eta & \boldsymbol{\epsilon}^T\end{bmatrix}^T = \exp\left(-\frac{1}{2}\boldsymbol{e}_Q\right) * \boldsymbol{Q}_g$ the orientation expressed as a unit quaternion, $\boldsymbol{e}_{Q,0}$ the initial orientation error and $\boldsymbol{Q}_0, \boldsymbol{Q}_g$ the initial and goal orientation. The forcing term is given by (17) and the canonical system is the same as the position DMP given by (3). More information on unit quaternion kinematics can be found in [23], [25], [32].

REFERENCES

[1] S. Bragança, E. Costa, I. Castellucci, and P. M. Arezes, *A Brief Overview of the Use of Collaborative Robots in Industry 4.0: Human Role and Safety*. Cham: Springer International Publishing, 2019, pp. 641–650. [Online]. Available: https://doi.org/10.1007/978-3-030-14730-3_68

[2] S. Kumar, C. Savur, and F. Sahin, "Survey of human–robot collaboration in industrial settings: Awareness, intelligence, and compliance," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 51, no. 1, pp. 280–297, 2021.

[3] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," *The International Journal of Robotics Research*, vol. 5, no. 1, pp. 90–98, 1986. [Online]. Available: https://doi.org/10.1177/027836498600500106

[4] E. Rimon and D. Koditschek, "Exact robot navigation using artificial potential functions," *IEEE Transactions on Robotics and Automation*, vol. 8, no. 5, pp. 501–518, 1992.

[5] J. Minguez, F. Lamiraux, and J.-P. Laumond, *Springer Handbook of Robotics*. Berlin, Heidelberg: Springer-Verlag, 2007.

[6] V. Kunchev, L. Jain, V. Ivancevic, and A. Finn, "Path planning and obstacle avoidance for autonomous mobile robots: A review," in *Knowledge-Based Intelligent Information and Engineering Systems*, B. Gabrys, R. J. Howlett, and L. C. Jain, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 537–544.

[7] G. S. Kanakis and G. A. Rovithakis, "Improving safety in human-robot collaboration via dynamic active constraints enforcement," in *2021 30th IEEE International Conference on Robot & Human Interactive Communication (RO-MAN)*, 2021, pp. 515–519.

[8] M. Ginesi, D. Meli, A. Roberti, N. Sansonetto, and P. Fiorini, "Dynamic movement primitives: Volumetric obstacle avoidance using dynamic potential functions," *Journal of Intelligent & Robotic Systems*, vol. 101, no. 4, pp. 816–830, 2021.

[9] M. A. Murtaza, S. Aguilera, M. Waqas, and S. Hutchinson, "Safety compliant control for robotic manipulator with task and input constraints," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 10 659–10 664, 2022.

[10] R. Weitschat and H. Aschemann, "Safe and efficient human–robot collaboration part ii: Optimal generalized human-in-the-loop real-time motion generation," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 3781–3788, 2018.

[11] A. Calanca, R. Muradore, and P. Fiorini, "A review of algorithms for compliant control of stiff and fixed-compliance robots," *IEEE/ASME Transactions on Mechatronics*, vol. 21, no. 2, pp. 613–624, 2016.

[12] C. Ott, *Cartesian Impedance Control of Redundant and Flexible-Joint Robots*, 1st ed. Springer Publishing Company, Incorporated, 2008.

[13] A. Q. Keemink, H. van der Kooij, and A. H. Stienen, "Admittance control for physical human–robot interaction," *The International Journal of Robotics Research*, vol. 37, no. 11, pp. 1421–1444, 2018. [Online]. Available: https://doi.org/10.1177/0278364918768950

[14] A. Gams, B. Nemec, A. J. Ijspeert, and A. Ude, "Coupling movement primitives: Interaction with the environment and bimanual tasks," *IEEE Transactions on Robotics*, vol. 30, no. 4, pp. 816–830, 2014.

[15] K. Vlachos and Z. Doulgeri, "A control scheme with a novel dmp-robot coupling achieving compliance and tracking accuracy under unknown task dynamics and model uncertainties," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 2310–2316, 2020.

[16] A. Ijspeert, J. Nakanishi, and S. Schaal, "Movement imitation with nonlinear dynamical systems in humanoid robots," in *Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No.02CH37292)*, vol. 2, 2002, pp. 1398–1403 vol.2.

[17] S. Schaal, P. Mohajerian, and A. Ijspeert, "Dynamics systems vs. optimal control — a unifying view," in *Computational Neuroscience: Theoretical Insights into Brain Function*, ser. Progress in Brain Research, P. Cisek, T. Drew, and J. F. Kalaska, Eds. Elsevier, 2007, vol. 165, pp. 425–445. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0079612306650279

[18] A. J. Ijspeert, J. Nakanishi, H. Hoffmann, P. Pastor, and S. Schaal, "Dynamical movement primitives: Learning attractor models for motor behaviors," *Neural Computation*, vol. 25, no. 2, pp. 328–373, Feb 2013.

[19] Y. Ding and U. Thomas, "Improving safety and accuracy of impedance controlled robot manipulators with proximity perception and proactive impact reactions," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 3816–3821.

[20] R. Schiavi, A. Bicchi, and F. Flacco, "Integration of active and passive compliance control for safe human-robot coexistence," in *2009 IEEE International Conference on Robotics and Automation*, 2009, pp. 259–264.

[21] F. Tassi, E. De Momi, and A. Ajoudani, "Augmented hierarchical quadratic programming for adaptive compliance robot control," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 3568–3574.

[22] L. Koutras and Z. Doulgeri, "A novel dmp formulation for global and frame independent spatial scaling in the task space," in *2020 29th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*, 2020, pp. 727–732.

[23] L. Koutras and Z. Doulgeri, "A correct formulation for the orientation dynamic movement primitives for robot control in the cartesian space," in *Proceedings of The 3rd Conference on Robot Learning*, 30 Oct –1 Nov 2019.

[24] F. Bullo and R. M. Murray, "Tracking for fully actuated mechanical systems: a geometric framework," *Automatica*, vol. 35, no. 1, pp. 17–34, 1999.

[25] L. Koutras and Z. Doulgeri, "Exponential stability of trajectory tracking control in the orientation space utilizing unit quaternions," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021, pp. 8151–8158.

[26] W.-H. Chen, D. Ballance, P. Gawthrop, and J. O'Reilly, "A nonlinear disturbance observer for robotic manipulators," *IEEE Transactions on Industrial Electronics*, vol. 47, no. 4, pp. 932–938, 2000.

[27] A. Carron, E. Arcari, M. Wermelinger, L. Hewing, M. Hutter, and M. N. Zeilinger, "Data-driven model predictive control for trajectory tracking with a robotic arm," *IEEE Robotics and Automation Letters*, vol. 4, no. 4, pp. 3758–3765, 2019.

[28] T. Kastritsi and Z. Doulgeri, "A passive admittance controller to enforce remote center of motion and tool spatial constraints with application in hands-on surgical procedures," *Robotics and Autonomous Systems*, vol. 152, p. 104073, 2022.

[29] S. Stavridis and Z. Doulgeri, "Bimanual assembly of two parts with relative motion generation and task related optimization," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 7131–7136.

[30] L. Koutras and Z. Doulgeri, "Dynamic movement primitives for moving goals with temporal scaling adaptation," in *2020 IEEE International Conference on Robotics and Automation*, 2020.

[31] S. Chiaverini and B. Siciliano, "The unit quaternion: A useful tool for inverse kinematics of robot manipulators," *Systems Analysis Modelling Simulation*, vol. 35, no. 1, pp. 45–60, Jan. 1999.

[32] A. Ude, B. Nemec, T. Petrić, and J. Morimoto, "Orientation in cartesian space dynamic movement primitives," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, May 2014, pp. 2997–3004.