Coordinated Motion Generation and Object Placement: A Reactive Planning and Landing Approach

7 authors, including:

Jonathan Vorndamme
Technische Universität München
13 PUBLICATIONS   89 CITATIONS

SEE PROFILE

Zheng qu
Technische Universität München
3 PUBLICATIONS   48 CITATIONS

SEE PROFILE

Abdalla Swikir
Technical university of Munich
57 PUBLICATIONS   346 CITATIONS

SEE PROFILE

Coordinated Motion Generation and Object Placement: A Reactive Planning and Landing Approach

Riddhiman Laha[*,1], Jonathan Vorndamme[*,1], Luis F.C. Figueredo[1],
Zheng Qu[2], Abdalla Swikir[1], Christoph Jähne[2], and Sami Haddadin[1]

Abstract— Similar to human work, robotic tasks sometimes require two hands to be accomplished. This requires coordinated motion planning and control. While fulfilling the task in a coordinated manner is already a big challenge, the task at hand becomes even harder when obstacles are introduced in the environment that needs to be avoided. Furthermore in the case of dynamic environments, contacts cannot be avoided all the time, even with robust planning. In addition to geometric constraints, bimanual systems need to be able to detect and react to contacts during task execution. To this aim, we integrate a vector-field based planning scheme, that is able to avoid obstacles, with contact detection and reactive control methods based on contact wrench estimation such as admittance control. We also fuse the real contact forces into the planner directly together with the circular repulsive fields. The resulting planner-controller combination is capable of obstacle avoidance planning as well as reaction control in the case of unforeseen contacts that can also be used in situations where the manipulation needs to be guided by the environment such as landing control in only roughly known environments. We evaluate our approach on the torque-controlled Kobo bimanual set-up and also perform rigorous simulation studies.

I. INTRODUCTION

Cooperative manipulation planning raises several challenges in terms of motion coordination within the reduced cooperative workspace [1], [2]. Most planning strategies therefore focus on dealing with geometric constraints while satisfying the task-conditions stemming from the tightly coupled kinematics. In this work, instead, we are interested in scenarios where the cooperative robot manipulates objects not only under geometric constraints but also under desired and undesired external contact forces. Take, for instance, the task shown in Fig. 1 where the two-arm robot grasps a tray with a cup of water and generates coordinated joint motions to take the tray towards the other side of its reachable space – while addressing the geometric uncertainties of the environment (e.g., dimension of the tray) and dynamic obstacles (e.g., human in the scene). During the task, the human may accidentally contact the robot thereby generating undesired forces. On the other hand, during placement, contact forces are expected – yet uncertainties in task (e.g., wrong table height) may lead to larger contact forces which may destabilize the load or even damage the robot. In this paper,

Fig. 1. Dual-arm robot planning a path in the environment while reacting to unintended and intended contacts.

we propose an integrated planning and control framework that enables the cooperative system to manipulate objects under uncertain and dynamic scenarios like this, handling (un-)desired external forces, and design novel skills such as smooth landing and take-off of objects – crucial when handling delicate but large or bulky objects, for instance.

Enabling such cooperative motion capabilities is crucial to human-robot interaction applications where contacts are often inevitable. The solution for this, and many other planning tasks, requires robotic systems that are able to solve the following three key problems:

First, it is essential that our planning framework can produce coordinated motions and react within the proximity of obstacles – in a real-time fashion – while satisfying the cooperative task-constraints, e.g., holding the same relative distance, position or pose between end-effectors. In planning literature, most motion generation strategies rely on sampling-based methods [3], [4]. Cooperative manipulation, nonetheless, is inherently a constrained problem in a high dimensional space – and sampling becomes non-trivial. Existing solutions, e.g. [5]–[7], often cannot match the real-time requirements. Reactive methods, on the other hand, lead to systems that are often more responsive and suitable for compensating inaccuracies. Being intrinsically connected to control, reactive motion generation also breaks the standard sense-plan-act strategy leading to enhanced performance in dynamic and changing environments [8], [9]. The pioneering work in this domain [10] used artificial vector fields to design high and low potentials that led the robot towards the goal while avoiding the obstacles. Recently, approaches stimulated by electro-magnetic fields and Lorentz forces [11]–[14] claiming to be free of local-minima have come to the light, although global problems still persist. The circular fields (CF) approach [11], [12], from which we draw heavy inspiration, rotates the robot

away from the obstacles, which emanate an artificial magnetic-like field. Since this process does not affect the total energy, the ubiquitous local-minima problem is taken care of, at-least for simple scenarios. These vector field like approaches are usual routines in mobile robotics. However, in literature, very few papers discuss reactive vector-field-like approaches capabilities to scale up to robots with many degrees of freedom, e.g., humanoids and cooperative robots, which are manipulating along increasingly complex environments and task-space constraints, e.g., [15]–[17].

Indeed, real-time cooperative planning using reactive fields were firstly introduced in our previous work [18]. Therein, we explored cooperative primitives to reduce task dimensionality and enlarge the cooperative workspace. Notwithstanding, the planner neglected the environment uncertainties and dynamics, i.e., it was focused on a static fully observed scene. The method was also not well tested in real-world scenarios. In this work, we integrate the planner and geometric information dynamically in order to avoid future collisions while satisfying hard cooperative constraints.

Second, in a highly dynamic environment, it is not enough to only prevent undesired collisions. An efficient motion generation scheme must also account for and actively respond to unexpected events such as external forces. In this context, our real-time cooperative planning scheme is integrated with a set of tasks which include an admittance behaviour that ensures safety. The physical contact information must also be integrated into the planner in a unified manner. Hence, right after contact, the planner estimates the contact point and wrenches to build virtual fields which allow the robot to retract and circumvent the source of external forces. This provides additional flexibility to the cooperative system. Noteworthy, the integration of contact and wrench based perceptual feedback into a reactive planner was firstly presented in Haddadin et al. [14], [19]. Yet, to the best of the authors knowledge, there exists no result in the literature for cooperative manipulation despite the clear necessity of additional flexibility and task relaxations in such systems. One of the main challenges is the contact detection and wrench estimation for bimanual, humanoid and multi-DoF systems. As the straight forward adaptation of single arm strategies would tend to false positives in the contact detection due to internal strain of the bimanual system, we developed a method that is able to detect contacts while ignoring any internal strain in the system.

Third, in real scenarios, the robot needs to explore, rather than avoid, the contact forces along the placement surface. Most cooperative planning strategies focus solely on the transfer trajectory and assume objects can be simply dropped at the desired point. Whereas, in real scenarios, uncertainties in the geometry and texture of the contact surface may lead to failure, unexpected stresses in the load and end-effectors and can even damage arms unequipped with safety-stop criteria. In this paper, we address this problem by proposing a novel slide to placement strategy which is hereby termed as landing-control. The solution, inspired by airplane landing and take-off strategies and also by bees and bird behaviours, relies on a prelanding configuration from which the planner converges exponentially to the goal while enhancing admittance in the orientation of the load – similar to landing-gear shock-absorbing systems in airplanes. The compliance relates to the approaching strategy and at the time of contact the cooperative system is ready to adjust the load according to the final contact

surface. Conversely, the same concept can be used for taking-off which is less complex and involves smaller forces than landing.

Therefore, in this paper, we extend the state of the art and incorporate the contact wrench estimation into the cooperative planning scheme by

- improving the planner compliance with the environment through an inner cooperative admittance controller in attitude;
- exploring environment stabilization features with the admittance controller in order to let the environment guide the robot while the landing;
- integrating the sensed contact wrenches into the virtual force field in order to be able to reactively evade undesired contacts while enabling an increased motion velocity in uncertain environments.

II. COOPERATIVE DUAL TASK-SPACE

This section provides core concepts concerning the cooperative task-space and controllable primitives which are the backbone of the proposed planning and control approach. Among existing cooperative task-space representation [20]–[24], our approach heavily relies on dual quaternion algebra. The cooperative dual-task space[1] is more efficient than other models in terms of representation, computational complexity and, most important, its capability to extract geometric properties and primitives even in highly constrained contexts such as the cooperative spaces [1], [25]–[28].

Rigid body motion and transformations within dual quaternion algebra are represented by $\underline{\boldsymbol{x}} = \boldsymbol{r} + \frac{1}{2}\varepsilon\boldsymbol{p}\boldsymbol{r}$, where $\boldsymbol{r} = \cos(\phi/2) + \sin(\phi/2)\boldsymbol{n}$ represents a rotation with angle ϕ around the axis $\boldsymbol{n}$ [29], and $\boldsymbol{p}$ is a pure quaternion that represents the translation. The nilpotent Clifford unit ε is such that $\varepsilon \neq 0$ but $\varepsilon^2 = 0$, [30]. Under multiplication, the set of elements $\underline{\boldsymbol{x}}$ form the unit dual-quaternion group $\mathrm{Spin}(3) \ltimes \mathbb{R}^3$ with inverse element given by the conjugate $\underline{\boldsymbol{x}}^* = \boldsymbol{r}^* + \frac{1}{2}\varepsilon\boldsymbol{r}^*\boldsymbol{p}^*$. Dual quaternion elements $\underline{\boldsymbol{h}}$ can also be described by $\underline{\boldsymbol{h}} = \mathcal{P}(\underline{\boldsymbol{h}}) + \varepsilon\,\mathcal{D}(\underline{\boldsymbol{h}})$, where $\mathcal{P}(\underline{\boldsymbol{h}})$ and $\mathcal{D}(\underline{\boldsymbol{h}})$ are the primary and dual components.

A sequence of rigid body motions can be represented by a sequence of unit dual quaternion multiplications. For instance, the end-effector of a manipulator can be computed by the rigid-body transformations (multiplications) throughout the links. Similarly, a transformation between one hand to another, e.g., $\underline{\boldsymbol{x}}_2$ to $\underline{\boldsymbol{x}}_1$, can be described by $\underline{\boldsymbol{x}}_2\underline{\boldsymbol{x}}_r = \underline{\boldsymbol{x}}_1$ where $\underline{\boldsymbol{x}}_r$ depicts the relative transformation.

The *cooperative dual task-space (CDTS)* explores the relative ($\underline{\boldsymbol{x}}_r$) together with the absolute pose ($\underline{\boldsymbol{x}}_a$) transformation,[2] i.e.,

$$\underline{\boldsymbol{x}}_r = \underline{\boldsymbol{x}}_2^*\underline{\boldsymbol{x}}_1, \tag{1}$$

$$\underline{\boldsymbol{x}}_a = \underline{\boldsymbol{x}}_2\underline{\boldsymbol{x}}_{r/2}, \tag{2}$$

to describe different geometric primitives as shown in Fig. 2. In (1)-(2), $\underline{\boldsymbol{x}}_{r/2}$ depicts half of the transformation from left to right end-effectors, i.e., half of the angle ϕ_r around the axis $\boldsymbol{n}_r = \hat{i}n_x + \hat{j}n_y + \hat{k}n_z$ of the quaternion $\mathcal{P}(\underline{x}_r)$ and half of the translation between the two arms [25]. The associated cooperative kinematics are mapped from joint space actions

$$\underline{\dot{\boldsymbol{x}}}_r = \boldsymbol{J}_{\underline{\boldsymbol{x}}_r}\dot{\boldsymbol{q}}, \quad \text{and} \quad \underline{\dot{\boldsymbol{x}}}_a = \boldsymbol{J}_{\underline{\boldsymbol{x}}_a}\dot{\boldsymbol{q}},$$

---

[1]See [1], [25], for further details

[2]The absolute pose is located between end-effectors w.r.t. to a common coordinate system yet, without loss of generality, it can be shifted by means of a constant transformation.
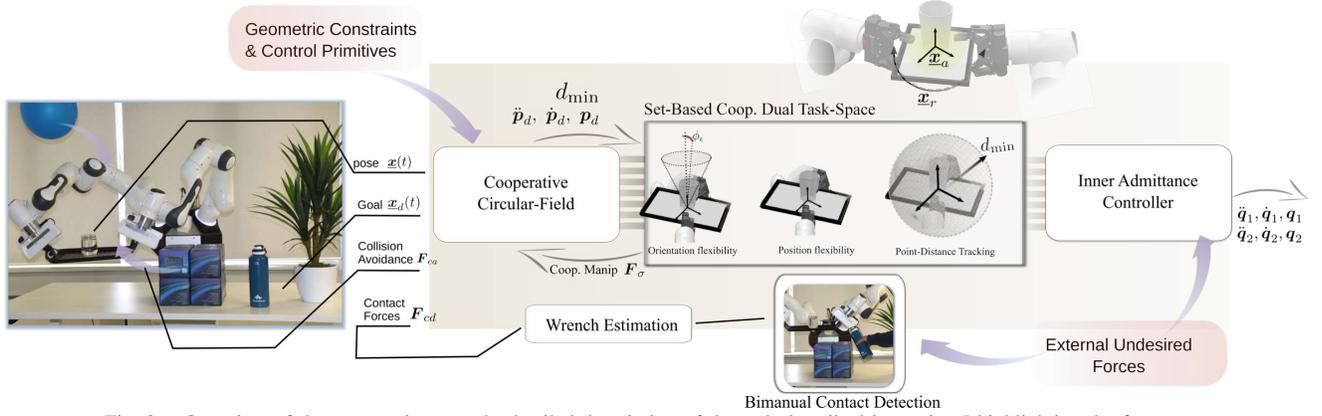
Fig. 2. Overview of the proposed approach: detailed description of the task described in section I highlighting the features.

through the relative and absolute dual quaternion Jacobians,

$$J_{\underline{\boldsymbol{x}}_r} = \left[\overset{+}{\boldsymbol{H}}(\underline{\boldsymbol{x}}_2^*)\boldsymbol{J}_{\underline{\boldsymbol{x}}_1} \quad \overset{-}{\boldsymbol{H}}(\underline{\boldsymbol{x}}_1)\boldsymbol{J}_{\underline{\boldsymbol{x}}_2}^* \right], \qquad (3)$$

$$J_{\underline{\boldsymbol{x}}_a} = \left[\overset{-}{\boldsymbol{H}}(\underline{\boldsymbol{x}}_{r/2})\boldsymbol{J}_{\underline{\boldsymbol{x}}_{2ext}} + \overset{+}{\boldsymbol{H}}(\underline{\boldsymbol{x}}_2)\boldsymbol{J}_{\underline{\boldsymbol{x}}_{r/2}} \right], \qquad (4)$$

where $\boldsymbol{q} = [\boldsymbol{q}_1^T \quad \boldsymbol{q}_2^T]^T \in \mathbb{R}^n$ is the augmented joint vector, and $\boldsymbol{J}_{\underline{\boldsymbol{x}}_i} = \partial f_i / \partial \boldsymbol{q}_i$ is the analytical Jacobian, which can be straightforwardly derived using dual quaternion algebra as in [28], [31]. The matrix $\boldsymbol{J}_{\underline{\boldsymbol{x}}_{2ext}}$ is defined by $[\boldsymbol{0} \ \boldsymbol{J}_{\underline{\boldsymbol{x}}_2}]$, and $\boldsymbol{J}_{\underline{\boldsymbol{x}}_{r/2}}$ is given by

$$J_{\underline{\boldsymbol{x}}_{r/2}} = \begin{bmatrix} \frac{1}{2}\overset{-}{\boldsymbol{H}}_4(\boldsymbol{r}^*{}_{r/2})\boldsymbol{J}_{\mathcal{P}(\underline{\boldsymbol{x}}_r)} \\ \frac{1}{4}\left(\overset{-}{\boldsymbol{H}}_4(\boldsymbol{r}_{r/2})\boldsymbol{J}_{\boldsymbol{p}_r} + \overset{+}{\boldsymbol{H}}_4(\boldsymbol{p}_r)\boldsymbol{J}_{\mathcal{P}(\underline{\boldsymbol{x}}_{r/2})}\right) \end{bmatrix}$$

where $\boldsymbol{J}_{\mathcal{P}(\underline{\boldsymbol{x}}_{r/2})}$ refers to the first four rows of the relative dual quaternion Jacobian $\boldsymbol{J}_{\underline{\boldsymbol{x}}_r}$. The matrices $\overset{+}{\boldsymbol{H}}$ and $\overset{-}{\boldsymbol{H}}$ are Hamilton operators that can be used to commute terms when performing dual quaternions multiplications.[3] Considering the mapping of the dual quaternion set $\mathcal{H}$ into $\mathbb{R}^8$, that is, vec : $\mathcal{H} \to \mathbb{R}^8$ and the dual quaternion $\underline{\boldsymbol{z}} = \underline{\boldsymbol{x}}\underline{\boldsymbol{y}}$, the Hamilton operators, $\overset{+}{\boldsymbol{H}}$ and $\overset{-}{\boldsymbol{H}}$, satisfy vec $\underline{\boldsymbol{z}} = \overset{+}{\boldsymbol{H}}(\underline{\boldsymbol{x}})$ vec $\underline{\boldsymbol{y}} = \overset{-}{\boldsymbol{H}}(\underline{\boldsymbol{y}})$ vec $\underline{\boldsymbol{x}}$. Similarly, we can use $\overset{+}{\boldsymbol{H}}_4$ and $\overset{-}{\boldsymbol{H}}_4$ for the mapping $\mathcal{S}^3 \to \mathbb{R}^4$, for further details, see [28], [31]. Notice that the cooperative dual-task space formulation is more general than a leader-follower framework, which often kinematically couples follower arms within the nullspace of the leader motion [32], [33]. The CDTS allows both asymmetrical (such as leader-follower) and symmetrical coordinated motion between arms. In our case, coordination is defined by cooperative and individual arm tasks (e.g., joint limit avoidance). A leader-follower framework can be obtained from CDTS, as does other frameworks, see, e.g., [34].

III. COOPERATIVE SET-BASED TASK-PRIORITY (CoSTP) CONTROLLER

This section explains our control framework. Stemming from relative and absolute variables defined in (1)-(2) and the corresponding Jacobians (3)-(4), we directly extract several cooperative geometric primitives and their local joint to task-space mapping. For brevity, we will focus on most common geometric primitives[4], as shown in Table I, for the relative or absolute variables $\chi = \{r, a\}$.

[3]Similar to $SE(3)$, unit dual quaternion multiplication is not commutative.
[4]For general geometric features that can be extracted from dual quaternions, we refer to [1], [35], [36].

A. Geometric cooperative task primitives

In this work, we explore the relative/absolute position between arms, the attitude transformation between them, the square norm distance, and the deviation along the z-axis[5] from a static line $\boldsymbol{l}$ and the current line $\boldsymbol{l}_z$, which is crucial to control the tilt angle of an object, e.g. a glass, for instance. The $\boldsymbol{l}_z = \boldsymbol{r}\hat{k}\boldsymbol{r}^*$ is a Plücker line resulting from the transformation of the z-axis (given by $\hat{k}$) from the frame defined by the rotation $\boldsymbol{r}$ to the coordinate system frame. Notice the line constraint Jacobian $\boldsymbol{J}_{r_z} = \left[\overset{-}{\boldsymbol{H}}_4(\hat{k}\boldsymbol{r}^*)\boldsymbol{J}_r + \overset{+}{\boldsymbol{H}}_4(\boldsymbol{r}\hat{k})\boldsymbol{C}_4\boldsymbol{J}_r \right]$ is defined as in [18], [36], with $\boldsymbol{C}_4 = \text{diag}(1, -1, -1, -1)$, and $\boldsymbol{J}_r = \boldsymbol{J}_{\mathcal{P}(\underline{\boldsymbol{x}}_\chi)}$ being the rotation Jacobian – four upper rows of the analytical Jacobian $\boldsymbol{J}_{\underline{\boldsymbol{x}}_\chi}$.

In addition to the geometric primitives, an efficient bimanual manipulation task requires control of variables concerning the joint configuration, e.g., joint-limit avoidance ($\underline{q_i} \leq q_i \leq \overline{q_i}$) and the minimum singular value associated with the absolute and relative cooperative kinematics. The former, for instance, can be simply described by $\tilde{q}_{\text{lim}} = \frac{1}{2}(\boldsymbol{q}_c - \boldsymbol{q})^T(\boldsymbol{q}_c - \boldsymbol{q})$, where $\boldsymbol{q}_{c(i)} = \frac{(\underline{q_i} + \overline{q_i})}{2}$, with kinematics guided by $\boldsymbol{J}_q = (\boldsymbol{q}_c - \boldsymbol{q})$.

It is important to highlight that despite the desired pose or planned trajectories, the dynamics and behaviour of the system is guided by the geometric primitives from Table I together with (3)-(4). In most cooperative tasks, for instance, the relative pose is fixed (e.g., tightly grasping a shared object) and actions, such as controlling the absolute (shared load) position $\boldsymbol{p}_a$, are done in the nullspace of $\boldsymbol{J}_{\underline{\boldsymbol{x}}_r}$ from (3). The increased number of constraints required to successfully track a trajectory while satifying such constraints may pose serious issues for practical implementation. With this in mind, we have previously designed switching strategies based on the controllable primitives in Table I that follows the definition below [1], [18].

Definition 1. *For a given set $\mathfrak{S} \subseteq Spin(3) \ltimes \mathbb{R}^3$, the following proper subsets can be drawn from geometric structures of interest with regard to this set,*

$$\mathfrak{R}_d(\mathfrak{S}) = \left\{ d \in \mathbb{R} \mid d = \left\| \mathcal{T}(\underline{\boldsymbol{x}}_g) \right\|, \ \underline{\boldsymbol{x}}_g \in \mathfrak{S} \right\},$$

$$\mathfrak{R}_p(\mathfrak{S}) = \left\{ \boldsymbol{p} \in \mathbb{H}_0 \mid \boldsymbol{p} = \mathcal{T}(\underline{\boldsymbol{x}}_g), \ \underline{\boldsymbol{x}}_g \in \mathfrak{S} \right\},$$

$$\mathfrak{R}_o(\mathfrak{S}) = \left\{ \boldsymbol{r} \in Spin(3) \mid \boldsymbol{r} = \mathcal{P}(\underline{\boldsymbol{x}}_g), \ \underline{\boldsymbol{x}}_g \in \mathfrak{S} \right\},$$

$$\mathfrak{R}_{\phi_\iota}(\mathfrak{S}) = \left\{ \phi_\iota \in \mathbb{R} \mid \phi_\iota = \cos^{-1}(\langle \boldsymbol{l}_z, \boldsymbol{l} \rangle), \ \boldsymbol{l}_z = \boldsymbol{r}\hat{k}\boldsymbol{r}^* \right\},$$

[5]Without loss of generality, the deviation could take place along any unitary axis.

TABLE I

Task Primitive	$\boldsymbol{u}$	Contr. Sets	DOFs	Task Jacobian
Rel/Abs position ($\boldsymbol{p} \in \mathfrak{R}_p$)	$\boldsymbol{p}_\chi = 2\,\mathcal{D}\left(\underline{\boldsymbol{x}}_\chi\right) \mathcal{P}\left(\underline{\boldsymbol{x}}_\chi\right)^*$	$\underline{\boldsymbol{p}} \preceq \boldsymbol{p} \preceq \overline{\boldsymbol{p}}$	3	$\boldsymbol{J}_{\boldsymbol{p}_\chi} = \left[\overset{+}{\boldsymbol{H}}(\underline{\boldsymbol{x}}_2^*)\boldsymbol{J}_{\underline{\boldsymbol{x}}_1} \quad \overline{\boldsymbol{H}}(\underline{\boldsymbol{x}}_1)\boldsymbol{J}_{\underline{\boldsymbol{x}}_2}^* \right]$
Rel/Abs orientation ($\boldsymbol{r} \in \mathfrak{R}_o$)	$\boldsymbol{r}_\chi = \mathcal{P}\left(\underline{\boldsymbol{x}}_\chi\right)$	$\underline{\boldsymbol{r}_\ell} \preceq \log \boldsymbol{r} \preceq \overline{\boldsymbol{r}_\ell}$	3	$\boldsymbol{J}_{\boldsymbol{r}_\chi} = \boldsymbol{J}_{\mathcal{P}(\underline{\boldsymbol{x}}_\chi)}$
Rel/Abs distance ($d \in \mathfrak{R}_d$)	$d \triangleq \|\boldsymbol{p}_\chi\|^2$	$\underline{d} \leq d \leq \overline{d}$	1	$\boldsymbol{J}_{d_\chi} = 2(\mathrm{vec}_4^T\,\boldsymbol{p}_\chi)\boldsymbol{J}_{\boldsymbol{p}_\chi}$
Rel/Abs tilt ($\phi_\iota \in \mathfrak{R}_{\phi_\iota}$)	$\tilde{\mathbf{l}} = (\boldsymbol{l} - \boldsymbol{l}_z)^T(\boldsymbol{l} - \boldsymbol{l}_z)$	$\underline{\phi_\iota} \leq \phi_\iota \leq \overline{\phi_\iota}$	1	$\boldsymbol{J}_{l_{zerr}} = -2(\boldsymbol{l} - \boldsymbol{l}_z)^T \boldsymbol{J}_{r_z}$

where $\boldsymbol{\mathcal{T}}\left(\underline{\boldsymbol{x}}_g\right) \triangleq 2\mathcal{D}\left(\underline{\boldsymbol{x}}_g\right)\mathcal{P}\left(\underline{\boldsymbol{x}}_g\right)^*$ *is the translation described as in Table I, and* $\mathbb{H}_0$ *is the set of all pure quaternions, isomorphic to* $\mathbb{R}^3$. *Briefly,* ϕ_ι *describes the opening angle of a solid cone defined by the rotation of the body z-axis to the coordinate frame, i.e.,* $\boldsymbol{l}_z$ *around a desired line* $\boldsymbol{l}$, *as shown in Fig. 2.*

From the primitive sets, Definition 1, we can properly define the controllable subsets as shown in Table I. To this aim, let us now consider the spaces $\mathfrak{R}_\chi \subset \mathrm{Spin}(3) \ltimes \mathbb{R}^3$ which correspond to the space of controllable relative poses, $\chi = r$, and absolute poses, $\chi = a$. Hence, each task primitive in Table I should have a counterpart w.r.t. the absolute and relative frames. For instance, consider the absolute distant primitive set, which is defined by the $\mathfrak{R}_d = \{d_a \in \mathfrak{R}_d(\mathfrak{R}_a) \mid \underline{d}_a \leq d_a \leq \overline{d}_a\}$, where $\underline{d}_a$ is the absolute distance bounded by $\underline{d}_a$ and $\overline{d}_a$. Finally, notice that each geometric primitive requires different DoFs from the CDTS (e.g., 1 for the relative/absolute distance, 3 for the absolute position, etc) which can be further explored to enhance flexibility and ensure desired reactiveness in cooperative planning scenarios. More specifically, considering a symmetric tightly coupled manipulation task, one can define the desired trajectory according to the 3D translation – e.g., taking the desired absolute trajectory $\boldsymbol{p}_a(t)$ to be given by the sum of forces in (14) – or according to the distance to the desired trajectory, i.e., building a funnel around the result from (14) which requires only 1 DoF instead of 3. We refer readers to [18] for further information about how to explore geometric primitives within the motion generation context.

B. Switching strategy

The differential equation which captures each task in the CDTS is given by $\dot{\boldsymbol{u}} = \boldsymbol{J}_u \dot{\boldsymbol{q}}$. Each task has its corresponding error for attaining a desired equilibrium represented by $\boldsymbol{e} = \boldsymbol{u} - \hat{\boldsymbol{u}}$. The error dynamics are governed by $\dot{\boldsymbol{e}} = \boldsymbol{J}_u \dot{\boldsymbol{q}}$. Now, differentiating the above we obtain $\ddot{\boldsymbol{e}} = \dot{\boldsymbol{J}}_u \dot{\boldsymbol{q}} + \boldsymbol{J}_u \ddot{\boldsymbol{q}}$, which is the resulting acceleration. Our CDTS definition helps us to exploit the redundant degrees of freedom, thereby paving the way for defining lower priority tasks in the nullspace of the primary task $\boldsymbol{u}$. Therefore, the joint velocities are given by,

$$\dot{\boldsymbol{q}} = \boldsymbol{J}_u^+ \dot{\boldsymbol{e}} + \boldsymbol{P}_u \dot{\boldsymbol{q}}_{\mathrm{lp}},$$

where $\dot{\boldsymbol{e}}$ is the desired error convergence, and $\boldsymbol{P}_u = (\boldsymbol{I} - \boldsymbol{J}_u^+ \boldsymbol{J}_u)$ is the nullspace projector that projects the desired velocities $\dot{\boldsymbol{q}}_{\mathrm{lp}} \in \mathbb{R}^n$ of the lower priority tasks in the nullspace of $\boldsymbol{J}_u$, [37]. For a given order of η tasks, the joint velocities would be computed as

$$\dot{\boldsymbol{q}} = \boldsymbol{J}_{u_1}^+ \kappa_1 \dot{\boldsymbol{e}}_1 + \boldsymbol{P}_{u_1} \kappa_2 \dot{\boldsymbol{e}}_2 + \boldsymbol{P}_{u_{1:2}} \kappa_3 \dot{\boldsymbol{e}}_3 + \cdots + \boldsymbol{P}_{u_{1:\eta-1}} \kappa_\eta \dot{\boldsymbol{e}}_\eta,$$

κ_i is the positive gain which is defined accordingly to [38] such to ensure asymptotic convergence of the resulting tasks. The matrix $\boldsymbol{P}_{u_{1:i}}$ now denotes the nullspace projector for an augmented Jacobian $\boldsymbol{J}_{u_{1:i}} = [\boldsymbol{J}_{u_1} \cdots \boldsymbol{J}_{u_i}]$. Our switching strategy is centred on the idea of activation and deactivation of

tasks, each of which evolve individually and are in accordance with a set-based geometric region. For instance, for a given task i, its first order dynamics is therefore expressed by the task Jacobian mapping $\boldsymbol{J}_{u_i}$ and the controlled joint velocity for the whole closed-loop system, such that

$$\dot{\boldsymbol{e}}_i = \boldsymbol{J}_{u_i} \dot{\boldsymbol{q}} = \boldsymbol{J}_{u_i} \left(\boldsymbol{J}_{u_1}^+ \kappa_1 \dot{\boldsymbol{e}}_1 + \cdots + \boldsymbol{P}_{u_{1:\eta-1}} \kappa_\eta \dot{\boldsymbol{e}}_\eta \right).$$

The scheme is designed to explore bounded geometric regions as described in table I. As we are using specific control gains, the asymptotic stability of the active tasks can be proved leveraging the fact that all possible N modes are asymptotically stable. Undesirable residual torques, which usually reduce robot performance can be handled using a hysteresis based fading. For a more detailed discussion on set-based switching strategy we refer readers to [39], [18].

C. Bimanual collision detection and wrench estimation

Classical approaches to detect contacts in bimanual tasks based on external torques such as [40] will fail in a bimanual manipulation setting, as the internal strain of the system even with low errors in the relative control will easily exceed the torque thresholds and lead to false detections continously. We therefore developed the following approach for bimanual contact detection utilizing the estimation of the external wrenches at the end-effectors of both arms. The contact wrenches $\boldsymbol{\mathcal{F}}_{\mathrm{ext,l}} = \left(\boldsymbol{f}_{\mathrm{ext,l}}^{\mathrm{T}} \quad \boldsymbol{m}_{\mathrm{ext,l}}^{\mathrm{T}}\right)^{\mathrm{T}}$ and $\boldsymbol{\mathcal{F}}_{\mathrm{ext,r}} = \left(\boldsymbol{f}_{\mathrm{ext,r}}^{\mathrm{T}} \quad \boldsymbol{m}_{\mathrm{ext,r}}^{\mathrm{T}}\right)^{\mathrm{T}}$ for left and the right arm respectively, are calculated internally in the robot controller based on the external torque estimation. We use them to calculate the estimated external wrench $\boldsymbol{\mathcal{F}}_{\mathrm{ext}} = \left(\boldsymbol{f}_{\mathrm{ext}}^{\mathrm{T}} \quad \boldsymbol{m}_{\mathrm{ext}}^{\mathrm{T}}\right)^{\mathrm{T}}$ on the intermediate frame between the two arms (absolute pose) as

$$\boldsymbol{\mathcal{F}}_{\mathrm{ext}} := \begin{pmatrix} \boldsymbol{f}_{\mathrm{ext,l}} + \boldsymbol{f}_{\mathrm{ext,r}} \\ \boldsymbol{m}_{\mathrm{ext,l}} + \boldsymbol{m}_{\mathrm{ext,r}} - \frac{1}{2}\boldsymbol{v}_{\mathrm{lr}} \times \boldsymbol{f}_{\mathrm{ext,l}} + \frac{1}{2}\boldsymbol{v}_{\mathrm{lr}} \times \boldsymbol{f}_{\mathrm{ext,r}} \end{pmatrix}. \quad (5)$$

where $\boldsymbol{v}_{\mathrm{lr}}$ is the vector from the left to the right arm end-effector. We test $\boldsymbol{\mathcal{F}}_{\mathrm{ext}}$ against a fixed threshold $\boldsymbol{\mathcal{F}}_{\mathrm{ext,th}}$ element-wise for contact detection. The advantage of using $\boldsymbol{\mathcal{F}}_{\mathrm{ext}}$ for contact detection is the elimination of false detections due to internal strain, as any internal strain will show up in both arms wrenches but with a different sign and therefore not influence $\boldsymbol{\mathcal{F}}_{\mathrm{ext}}$. This information is further exploited in the control and as an input to the planner to enable contact reactions. The planner can straightforwardly integrate contact forces that are sensed into the planning scheme and the measured torque can be used for rotational admittance control in the controller to be able to adapt to unknown surfaces. In order to not react to modeling errors or sensor noise, the following dead zone function is applied to the elements of $\boldsymbol{f}_{\mathrm{ext}}$ in the experiments:

$$f_{\mathrm{ext},i}^* := \begin{cases} f_{\mathrm{ext},i} + f_o, & f_{\mathrm{ext},i} < -f_o \\ 0, & -f_o <= f_{\mathrm{ext},i} <= f_o \\ f_{\mathrm{ext},i} - f_o, & f_{\mathrm{ext},i} > f_o \end{cases} \quad (6)$$

where $i \in \{x, y, z\}$ and f_o is the half size of the dead zone.

D. Landing-and-take-off-controller

When the robot is close to the landing surface of roughly known position and orientation, it will establish contact with that surface. At detecting a contact, the control for the rotation of the intermediate frame will change from the tilt controller to admittance control according to the control law

$$\dot{\boldsymbol{q}}_{\text{rot}} := \boldsymbol{J}_{\text{g,rot}}^{+} \int_0^t \boldsymbol{m}_{\text{ext}}^*(\tau) - k_\omega \boldsymbol{\omega}(\tau) d\tau \qquad (7)$$

where $\boldsymbol{J}_{\text{g,rot}}^{+}$, $\boldsymbol{\omega}$ and k_ω denote a robust pseudo inverse of the rotational geometric Jacobian, the rotational velocity of the intermediate frame and the damping gain of the admittance control, respectively. Similar to the force, the dead zone function from (6) is also applied to $\boldsymbol{m}_{\text{ext}}$. This approach allows the robot to adapt to unknown surface inclinations and establish a steady contact before releasing the object.

IV. REACTIVE COOPERATIVE PLANNING STRATEGY FOR TASK SPACE TRAJECTORY GENERATION

In this section, we will explain how we generate the task-space trajectory $\boldsymbol{x}_d$ for the absolute position tracking task. In the first part of this section, we will present the standard algorithm based on circular fields that allows such reactive planning behaviour in a broader context. This concept follows closely the works of Haddadin et al. in [14], [19]. Thereafter, we introduce the artificial repulsive forces stemming from the task-flexibility approach based on cooperative set-based task-priority, from sec. II, and from the bimanual contact wrench estimation from sec. III-C. Finally, we introduce the concept of the landing-and-take-off-controller based on holistic integration of the cooperative tools introduced in the previous section.

A. Reactive circular field

The proposed strategy builds from the classic idea of integrating an attractor field – that, in our case, guides the desired cooperative task-variable towards its goal – and a repulsive field that reacts in the proximity to obstacles. From a point-mass robot example, we would have the following dynamics

$$m\ddot{\boldsymbol{x}}_d = \underbrace{-k_a(\boldsymbol{x}_d - \boldsymbol{x}_g)}_{\mathbf{F}_a} \underbrace{-k_d\dot{\boldsymbol{x}}_d}_{\mathbf{F}_d} + \mathbf{F}_c \qquad (8)$$

where $\boldsymbol{x}_d$ denotes the desired position, $\boldsymbol{x}_g$ the goal, k_a is some positive constant representing the intensity of the force field, m denotes the mass of the robot, and k_d is a positive constant for damping out oscillations. These terms shape the virtual attractor dynamics that pushes the system to the target goal. This is directly affected by obstacle avoidance force and others for desirable behavior. It is important to note that these forces only affect the vector field responsible for task-space behavior of the robot (i.e. the absolute position of the intermediate frame). Robot joint motion is handled entirely by the switching controller defined in Section II.

The circular field is a repulsive force generator $\mathbf{F}_c$ which takes inspiration on electromagnetism laws, [41]. The robot's end-effector is pushed away from the object like a charged particle in a megnetic field. We designed a proximity ball (with radius $\bar{r}$) wherein the repulsive actions take place – otherwise, the external force is ignored to reduce the resulting computation complexity. If the robot is located inside the proximity ball, i.e. it is close enough to the obstacle, then it will react with a circular force force given by

$$\mathbf{F_c} = (\boldsymbol{v} \times \mathbf{B})K(d), \qquad (9)$$

where $\boldsymbol{v}$ is the particle velocity, $\mathbf{B}$ is the desired magnetic field, and $K(d)$ is a scaling factor depending on the distance. In general we seek $K \propto \frac{1}{d^\gamma}$ for some $\gamma \in \mathbb{N}$, so the force is magnified in close proximity of the obstacle. In this work we used $\gamma = 1$. The desired magnetic field $\mathbf{B}$ needs to be such that the agent is pushed away from the obstacle. This is valid if the field is perpendicular to both the agent velocity and to the normal joining the agent and its goal to the obstacle. This perpendicular vector equals to $\boldsymbol{l} = \boldsymbol{l}_{OA} - \boldsymbol{l}_{GA} \frac{\langle \boldsymbol{l}_{OA}, \boldsymbol{l}_{GA} \rangle}{\|\boldsymbol{l}_{GA}\|^2}$, where $\boldsymbol{l}_{XY}$ is the vector to a point X from a point Y; and A, O and G denote the positions of agent, obstacle and goal, respectively. In case they are co-linear $\boldsymbol{l} = \boldsymbol{0}$, and $\boldsymbol{l}$ is set arbitrarily. Finally, we have $\mathbf{B} = \boldsymbol{v} \times \frac{\boldsymbol{l}}{\|\boldsymbol{l}\|}$ which yields

$$\mathbf{F}_c = \frac{k_c}{d^2} \left(\boldsymbol{v} \times \left(\boldsymbol{v} \times \frac{\boldsymbol{l}}{\|\boldsymbol{l}\|} \right) \right), \qquad (10)$$

with k_c being a constant parameter. Additionally, a repulsive force scaled by constant parameter k_r can also be used to get additional traction in trap scenarios

$$\mathbf{F}_r = \frac{k_r}{d(\boldsymbol{x}_g, \boldsymbol{o})^2} \frac{\delta d}{\delta \boldsymbol{x}_g}, \qquad (11)$$

where $d(\boldsymbol{x}_g, \boldsymbol{o})$ is the distance of the point to the obstacle $\boldsymbol{o}$.

B. Cooperative dual-task space forces

When generating motions in the cooperative task-space, the planner needs to explicitly consider the geometric and force-feedback constraints stemming from previous sections. In our previous work [18], we introduced the concept of *cooperative manipulability guidance force* ($\boldsymbol{F}_\sigma$) to address geometric constraints from the cooperative system geometry. Based on the singular value decomposition of the absolute position Jacobian, described in Table I, we designed forces in positive and negative directions of the output singular vector $\boldsymbol{u}_{\min}$ associated with the min. sing. value $\sigma_{\min}$, that is,

$$\boldsymbol{F}_\sigma = \lambda \boldsymbol{u}_{\min}, \qquad (12)$$

$$\lambda = \begin{cases} (1 - \frac{\sigma_{\min}}{\epsilon_\sigma})k_\sigma, & \text{if } \sigma_{\min} < \epsilon_\sigma; \\ 0, & \text{otherwise,} \end{cases} \qquad (13)$$

where λ is a gain as defined in [42]. If above a specific threshold λ is zero and linearly growing up to a maximum value (k_σ), otherwise. This also allows to scale the maximum velocities stemming from the virtual force – which is only introduced in this paper. The resulting guidance force leads the robot towards directions that require less effort (in terms of kinematic velocity) from the bimanual system.

In this work, we also introduce the concept of repulsive contact wrenches $\mathbf{F}_{\text{cw}}$. These are virtual forces generated when the robot is in undesired contact with obstacles. When such a contact is happening, the estimated wrench is fed to the circular field planner. The planner is then able to react to this contact force, thus being pushed away from the obstacle. The updated force field with all geometric-, manipulability- and contact and wrench-based perceptual feedback is given by

$$m\ddot{\boldsymbol{x}}_d = \mathbf{F}_a + \mathbf{F}_d + \mathbf{F}_c + \mathbf{F}_r + \boldsymbol{F}_\sigma + \mathbf{F}_{\text{cw}} \qquad (14)$$

$$\mathbf{F}_{\text{cw}} = k_{\text{cw}} \boldsymbol{f}_{\text{ext}}, \qquad (15)$$

where $\mathbf{F}_a$ is the attractive force towards the goal, $\mathbf{F}_d$ is the damping force, and $\boldsymbol{f}_{\text{ext}}$ is the contact force at the shared load estimated from (5) and $k_{\text{cw}} \geq 0$ is it's proportional gain. After (14) has been computed, the output velocity $\dot{\boldsymbol{x}}_d$ is limited to a defined maximum value. All these forces influence the

TABLE II

Scenario	Planner	Success Rate	Exec. Time
Static Scene	Decoupled CF	66	11055 ms
	CF-CoSTP	82	7854 ms
Dynamic Obstacles	Decoupled CF	44	10155 ms
	CF-CoSTP	90	6128.5 ms

geometric primitive(s) which are controlled based on the respective error dynamics resulting in appropriate joint inputs from the cooperative set-based task priority strategies in Section III.

V. SIMULATIONS AND EXPERIMENTS

This section explores quantitative and experimental aspects concerning the performance and evaluation of our proposed framework. Our reactive planner is successful in handling geometric constraints with respect to its responsive collision avoidance capabilities – while ensuring cooperative task constraints are satisfied. For experiments, we use the two arm system KoBo that features two slightly modified Franka Emika Panda arms. The controller is programmed (C++) in the ROS control framework and runs at a frequency of 1 kHz while the planner is running at 100 Hz in a separate process.

A. Quantitative analysis

To this aim, we devised two different scenarios, both implemented with two Franka Emika Panda robots,[6]. The first task is to bring a tray from a top shelf to a bottom one in a static yet highly constrained scene due to the size of the obstacles and the load. A second scene included two dynamic obstacles that were allowed to move in a plane along the environment. Both scenarios were executed sufficient times with different initial conditions. The obtained results were compared against a decoupled circular field with a cooperative control tracker (constrained both in relative and absolute poses). The results, summarized in Table II, clearly indicate the necessity of an integrated approach as proposed. Furthermore the exploration of the relaxation techniques stemming from the proposed strategy enlarges the cooperative manipulability workspace and allows an increase in the success rate. Most of the tasks within the decoupled approach failed due to hard cooperative constraints.

B. Collision avoidance in constrained environment

In this experiment, the robot does a placement task after handover from a human. The main objective here is to carry the tray with a glass of water while avoiding obstacles along the way (the scene is similar to Fig. 2) and reaching the goal. The control tasks that we consider here are maintaining the relative pose, tilt angle, the absolute position and joint limit avoidance with gains κ_{rp}=0.001, κ_{ee}=−0.05, κ_{ap}=50.0, and κ_{jl}=0.0005 respectively. The tasks are prioritized in the given order and not switching was required. Finally, the parameters for the task-space trajectory generation are $m = 0.5$ [kg], $k_a = 0.8$, $k_d = 0.0$, $k_c = 0.3$, $k_r = 0.08$, $k_\sigma = 0$, $k_{\mathrm{cw}} = 0.2$, $k_\omega = 0.5$, $f_o = 6$ for the force and $f_o = 1.5$ for the torque.

In addition, we also performed tasks where the human would intentionally generate forces along the trajectory. The results from one of the trials is shown in Fig. 4. For more details we refer readers to our video attachment.

[6]Simulations were executed with CoppeliaSim & DQ Robotics [43].
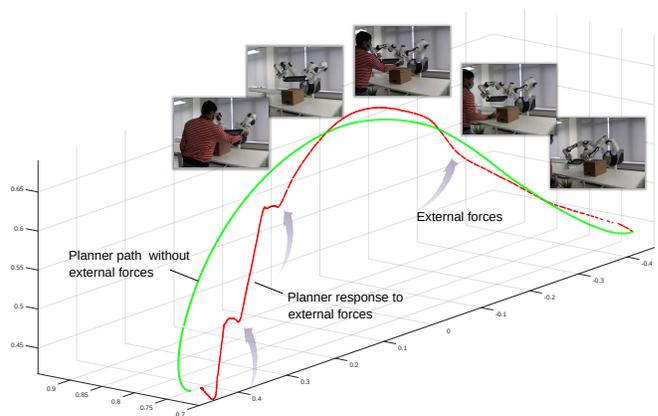
Fig. 3. Cooperative collision evasion strategy: the plot in green depicts the trajectory the planner generated without external forces whilst the red curve shows the response along time. Notice that both scenarios are different and therefore a different path is not only expected but desired.
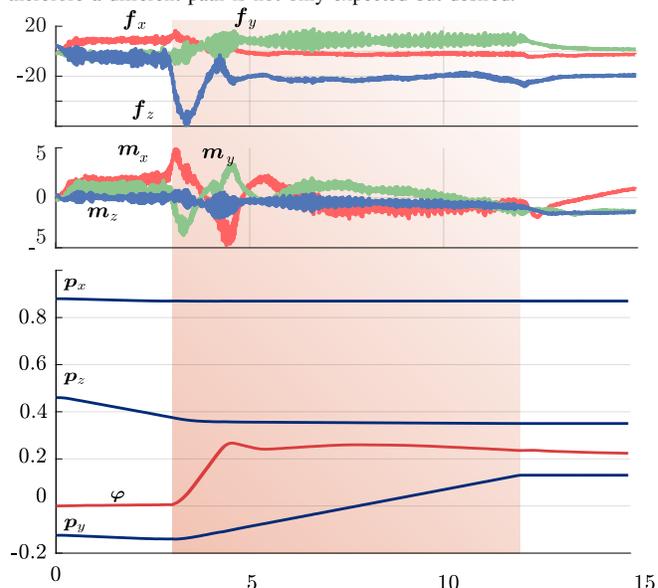
Fig. 4. Trajectory for a landing ($\boldsymbol{p}$) with a sliding motion along the y axis, and normal angle (φ) along z-axis, and external forces ($\boldsymbol{f}$) and torques ($\boldsymbol{m}$) according to (5). The landing takes place in the marked area of the plot – starting with the contact detection and ending when the final goal is reached by sliding the object over the table.

C. Landing task

In this experiment, the robot is holding a box and is commanded to go down until it detects a contact with the table surface. In contrast to the previous experiment, the rotational admittance task with gain κ_{ad} is added. The planning parameters are $m = 0.5$ [kg], $k_a = 0.5$, $k_d = 0.0$, $k_c = 0.3$, $k_r = 0.08$, $k_\sigma = 0$, $k_{\mathrm{cw}} = 0.2$, $k_\omega = 0.5$, $f_o = 6$ for the force and $f_o = 1.5$ for the torque. Task gains are the same from the previous subsection. When the contact is detected, the controller switches the tilt controller for the rotational admittance controller, at third priority. Afterwards it lands by sliding on the table surface along the y-axis. As the planner starts with an angle of 15 degrees towards the table, it needs to adapt its orientation during the sliding motion.

Fig. 4 shows the motion behavior during the task. After about 3 seconds, the contact is detected and the landing control is in command. It can be seen from the angle plot, that the planner clearly aligns to the table. As it reacts to the torque it senses, the overall torque stays in a small range between -5 and 5 Nm although the force is at an equilibrium

of -20 N after the adaptation, as can be seen in Fig. 4. Also the initially high forces of -60 N in z direction decrease significantly as soon as the planner adapts to the surface after about 1.5 seconds. Therefore, the landing strategy can be evaluated as successful.

VI. Conclusion

In this paper, we present a strategy for safe reactive constrained bimanual planning for scenarios where the robot has to interact (intentional or unintentional) with the environment. This is achieved by a reactive cooperative planning policy along with bimanual collision detection in addition to a landing-and-take-off controller. In future work, we plan to integrate online adaptation schemes for grasping and predictive multiple agents for a global planning method.

References

[1] L. Figueredo, B. V. Adorno, J. Y. Ishihara, and G. Borges, "Switching strategy for flexible task execution using the cooperative dual task-space framework," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2014, pp. 1703–1709.

[2] F. Caccavale and M. Uchiyama, *Handbook of Robotics - ch. 29. Cooperative Manipulators*. Springer, 2008.

[3] S. M. LaValle, *Planning Algorithms*. Cambridge University Press, 2006.

[4] J. J. Kuffner and S. M. LaValle, "Rrt-connect: An efficient approach to single-query path planning," in *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings*, vol. 2, 2000, pp. 995–1001 vol.2.

[5] M. Bonilla, L. Pallottino, and A. Bicchi, "Noninteracting constrained motion planning and control for robot manipulators," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2017, pp. 4038–4043.

[6] D. Berenson, S. S. Srinivasa, D. Ferguson, and J. J. Kuffner, "Manipulation planning on constraint manifolds," in *2009 IEEE International Conference on Robotics and Automation*, 2009, pp. 625–632.

[7] F. Burget, M. Bennewitz, and W. Burgard, "Bi2rrt*: An efficient sampling-based path planning framework for task-constrained mobile manipulation," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2016, pp. 3714–3721.

[8] D. Kragic, J. Gustafson, H. Karaoguz, P. Jensfelt, and R. Krug, "Interactive, collaborative robots: Challenges and opportunities," in *IJCAI International Joint Conference on Artificial Intelligence*, 2018.

[9] D. Kappler, F. Meier, J. Issac, J. Mainprice, C. G. Cifuentes, M. Wuthrich, V. Berenz, S. Schaal, N. Ratliff, and J. Bohg, "Real-time perception meets reactive motion generation," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 1864–1871, 2018.

[10] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," in *Proceedings. 1985 IEEE International Conference on Robotics and Automation*, vol. 2. IEEE, 1985, pp. 500–505.

[11] L. Singh, H. Stephanou, and J. Wen, "Real-time robot motion control with circulatory fields," in *Proceedings of IEEE International Conference on Robotics and Automation*, 1996, pp. 2737–2742.

[12] L. Singh, J. Wen, and H. Stephanou, "Motion planning and dynamic control of a linked manipulator using modified magnetic fields," in *Proceedings of the IEEE International Conference on Control Applications*. IEEE, 1997, pp. 9–15.

[13] A. Ataka, H.-K. Lam, and K. Althoefer, "Reactive magnetic-field-inspired navigation method for robots in unknown convex 3d environments," *IEEE Robotics and Automation Letters*, 2018.

[14] S. Haddadin, R. Belder, and A. Albu-Schäffer, "Dynamic motion planning for robots in partially unknown environments," *IFAC Proceedings Volumes*, vol. 44, no. 1, pp. 6842–6850, 2011.

[15] O. Brock and O. Khatib, "Elastic strips: A framework for motion generation in human environments," *The International Journal of Robotics Research*, vol. 21, no. 12, pp. 1031–1052, 2002.

[16] Y. Yang, V. Ivan, Z. Li, M. Fallon, and S. Vijayakumar, "idrm: Humanoid motion planning with realtime end-pose selection in complex environments," in *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, 11 2016, pp. 271–278.

[17] R. Laha, A. Rao, L. Figueredo, Q. Chang, S. Haddadin, and N. Chakraborty, "Point-to-point path planning based on user guidance and screw linear interpolation," in *Proceedings of the ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference (IDETC/CIE)*, August 2021.

[18] R. Laha, L. F. Figueredo, J. Vrabel, A. Swikir, and S. Haddadin, "Reactive Cooperative Manipulation based on Set Primitives and Circular Fields," in *IEEE International Conference on Robotics and Automation*, Xi'an, China, May 2021.

[19] S. Haddadin, H. Urbanek, S. Parusel, D. Burschka, J. Roßmann, A. Albu-Schäffer, and 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*, 2010, pp. 3109–3116.

[20] O. Khatib, "Object manipulation in a multi-effector robot system," in *Proceedings of the 4th International Symposium on Robotics Research*. Cambridge, MA, USA: MIT Press, 1988, pp. 137–144.

[21] M. Uchiyama and P. Dauchez, "A symmetric hybrid position/force control scheme for the coordination of two robots," in *Proceedings. 1988 IEEE International Conference on Robotics and Automation*, Apr 1988, pp. 350–356 vol.1.

[22] T. Connolly and F. Pfeiffer, "Cooperating manipulator control using dual quaternion coordinates," in *IEEE Conference on Decision and Control*, Lake Buena Vista, 1994, pp. 2417–2418.

[23] P. Chiacchio, S. Chiaverini, and B. Siciliano, "Direct and inverse kinematics for coordinated motion tasks of a two-manipulator system," *J. Dyn. Sys., Meas., Control*, no. 4, pp. 691–697, 1996.

[24] F. Caccavale, P. Chiacchio, and S. Chiaverini, "Task-space regulation of cooperative manipulators," *Automatica*, vol. 36, no. 6, pp. 879–887, 2000.

[25] B. V. Adorno, P. Fraisse, and S. Druon, "Dual position control strategies using the cooperative dual task-space framework," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Taipei, Oct. 2010, pp. 3955–3960.

[26] J. Funda and R. Paul, "A computational analysis of screw transformations in robotics," *IEEE Transactions on Robotics and Automation*, vol. 6, no. 3, pp. 348–356, 1990.

[27] E. Özgür and Y. Mezouar, "Kinematic modeling and control of a robot arm using unit dual quaternions," *Robotics and Autonomous Systems*, vol. 77, pp. 66 – 73, 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0921889015301184

[28] L. F. C. Figueredo, "Kinematic control based on dual quaternion algebra and its application to robot manipulators," Ph.D. dissertation, University of Brasilia, Brazil, 2016.

[29] J. Kuipers, *Quaternions and Rotation Sequences: A Primer with Applications to Orbits, Aerospace, and Virtual Reality*. Princeton University Press, 1999.

[30] J. M. Selig, *Geometric Fundamentals of Robotics*, 2nd ed. Springer-Verlag New York Inc., 2005.

[31] B. V. Adorno, "Robot Kinematic Modeling and Control Based on Dual Quaternion Algebra – Part I: Fundamentals," 2017.

[32] L. Yan, Y. Yang, W. Xu, and S. Vijayakumar, "Dual-arm coordinated motion planning and compliance control for capturing moving objects with large momentum," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 7137–7144.

[33] R. S. Jamisola Jr and R. G. Roberts, "A more compact expression of relative jacobian based on individual manipulator jacobians," *Robotics and Autonomous Systems*, vol. 63, pp. 158–164, 2015.

[34] H. A. Park and C. G. Lee, "Dual-arm coordinated-motion task specification and performance evaluation," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2016.

[35] M. M. Marinho, B. V. Adorno, K. Harada, and M. Mitsuishi, "Dynamic Active Constraints for Surgical Robots Using Vector-Field Inequalities," *IEEE Transactions on Robotics*, vol. 35, no. 5, pp. 1166–1185, 2019.

[36] J. J. Quiroz-Omaña and B. V. Adorno, "Whole-body control with (self) collision avoidance using vector field inequalities," *IEEE Robotics and Automation Letters*, vol. 4, no. 4, pp. 4048–4053, 2019.

[37] A. Liegeois, "Automatic supervisory control of the configuration and behavior of multibody mechanisms," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 7, no. 12, pp. 868–871, 1977.

[38] G. Antonelli, "Stability analysis for prioritized closed-loop inverse kinematic algorithms for redundant robotic systems," *IEEE Transactions on Robotics*, vol. 25, no. 5, pp. 985–994, 2009.

[39] S. Moe, G. Antonelli, A. R. Teel, K. Y. Pettersen, and J. Schrimpf, "Set-based tasks within the singularity-robust multiple task-priority inverse kinematics framework: General formulation, stability analysis, and experimental results," *Frontiers Robotics AI*, 2016.

[40] A. De Luca, A. Albu-Schäffer, S. Haddadin, and G. Hirzinger, "Collision detection and safe reaction with the DLR-III lightweight manipulator arm," in *IROS*, 2006.

[41] H. C. Verma, *Concepts of Physics, Part 2*. Bharti Bhavan, 2011.

[42] S. Chiaverini, "Singularity-robust task-priority redundancy resolution for real-time kinematic control of robot manipulators," *IEEE Transactions on Robotics and Automation*, pp. 398–410, Jun. 1997.

[43] B. V. Adorno and M. Marques Marinho, "Dq robotics: A library for robot modeling and control," *IEEE Robotics Automation Magazine*, pp. 0–0, 2020.