

Dual position control strategies using the cooperative dual task-space framework

Bruno Vilhena Adorno, Philippe Fraisse and Sébastien Druon

Abstract—We propose a set of control strategies for performing two arm manipulation with the goal of simplifying the task definition. In order to develop these strategies we propose a new representation, derived from the cooperative task-space, in the dual quaternion domain. The result is a compact and “singularity free” representation for two arm systems, named *cooperative dual task-space*. All the proposed control strategies share the same general scheme and are derived by using an analytical approach. Moreover, the mathematical treatment is given in a coherent and systematic fashion, and thus other strategies may be derived using the same argument. Experimental results show the effectiveness and usefulness of the cooperative dual task-space framework and the proposed control strategies.

I. INTRODUCTION

In the past thirty years a lot of research has been done in multi-manipulator systems [4]. This kind of cooperative system can be used to carry heavy payloads, to perform complex assembly tasks and in poorly structured environments, like outer space or undersea [4]. However, these advantages come with the drawback of leaving the system more complex. For instance, multiple manipulators cause internal stresses in the manipulated object, and a force control scheme has to be used in order to minimize these forces.

Thus, several approaches were proposed to tackle the problem of multi-arms manipulation. Khatib [10] presented a control scheme using N robots with the same number of DOF and rigidly connected to a common manipulated object. The control was made in the operational space using the concept of the augmented object, that is, the object submitted to the operational forces created by the N end-effectors acting at the operational point. In [15] it was proposed a physical model for internal forces and moments that appear in multi-arms manipulation. This model is based on a virtual linkage between the grasping points and was successfully applied in a system composed by two PUMA 560 manipulators using the augmented object model. Khatib et al. [9] used the concept of augmented object and virtual linkage model to implement a decentralized control between multiple mobile manipulator platforms. The system composed by the mobile platforms was able to perform tasks such erasing a whiteboard and cooperate in carrying a basket.

In [14] it was introduced the concept of symmetric control scheme for a tightly grasped object based on the relationships between forces and velocities in the object and the counterparts

in the “virtual sticks”, *i.e.*, vectors originating from the end-effector frame and ending at the origin of the reference frame of the object. Due to the principle of duality between force and velocity, a static analysis was used for finding the forces and then the velocities were found. Finally, the position of the virtual sticks could be determined by integrating the velocities, although extra care had to be taken in the orientation integration, since it was represented by Euler angles. The result was four meaningful variables: the absolute and relative positions; and absolute and relative orientations between the arms. Some authors have taken the inverse approach, defining directly a space composed by these four variables, named cooperative task-space [3], [5]. The main advantage of defining such a space is that the hypothesis of a firmly grasped object is relaxed, and then the control can be applied to flexible objects or for simply performing a coordinated movement between the arms. However, whenever a grasping occurs, external and internal forces appears and force control still has to be done for achieving good performance in the manipulation task.

Connolly and Pfeiffer [6] used normalized dual quaternion interpolation for generating a path between two points respecting the kinematics constraints of the resulted closed chain mechanism. Moreover, they performed force control by means of an external force control scheme [13] using an approximation for differential dual quaternions. Dooley and McCarthy [8] introduced the concept of operational image space. The operational space formulated for multiple manipulators by Khatib [10] was then represented in a subspace of the dual quaternion space, the image space. This representation was used to perform geometric analysis of the trajectories for cooperating robots.

In this paper, we propose a set of control strategies for performing two arm manipulation with the goal of simplifying the task definition. In order to develop these strategies we propose a new representation, derived from the cooperative task-space, in the dual quaternion domain. The result is a compact and “singularity free” representation for two arm systems, named cooperative dual task-space. The motivation for developing such representation arise in the context of the ASSIST project¹, whose goal is to build an assistant robot for interacting with quadriplegic people. As an example, the robot should be able of opening a bottle of water, filling a glass, and safely bringing it to the patient. Thus, a singularity free representation is not only desirable but also fundamental for achieving the high level of safety required by this kind

Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier (LIRMM), UMR 5506 - CC 477 161 rue Ada, 34392 Montpellier Cedex 5 - France. e-mail: {adorno, fraisse, druon}@lirmm.fr.

¹<http://www.lirmm.fr/assist>

of interaction, since singularities in the representation can potentially lead to inconsistencies in the task execution. One advantage of the proposed control strategies is that they share the same general scheme, and thus they can be easily changed in real time, accordingly to the requirements of the manipulation task. Moreover, some strategies permit the task execution without the need of specifying all the cooperative dual task-space variables. Furthermore, since we are using an analytical argument, and both forward kinematics model and the Jacobian are expressed directly in the dual quaternion space, it turns out that the mathematical treatment is given in a coherent and systematic fashion, and thus other strategies may be derived using the same procedure. For showing the usefulness and effectiveness of the proposed method, we performed experiments using the HOAP-3 robot for two usual tasks: grabbing a balloon and pouring water, both executed using the proposed strategies.

This paper is organized as follows: Section II briefly reviews rigid motions represented by dual quaternions, while Section III formalizes the cooperative dual task-space and in Section IV a set of control strategies is presented. Then, Section V shows the experimental results and discussions. Finally, Section VI presents the conclusions and the next steps in our research.

II. DUAL QUATERNIONS APPLIED TO RIGID MOTIONS

Dual quaternions have been proven to be a useful representation for describing rigid motions [2], [7], [17], [16], since they describe simultaneously both positions and orientations (hereafter regarded as dual positions) by using only eight parameters and are “singularity-free”. Furthermore, a sequence of rigid motions is represented by a sequence of dual quaternion multiplications. The dual quaternion

$$\underline{\mathbf{q}} = \mathbf{q} + \epsilon \frac{1}{2} \mathbf{t} \mathbf{q} \quad (1)$$

describing a rigid motion is a dual entity composed by a unit norm quaternion \mathbf{q} representing a rotation and an unbounded norm quaternion $\mathbf{q}' = \frac{1}{2} \mathbf{t} \mathbf{q}$, indirectly representing a translation.

The dual quaternion multiplication can be performed in two ways. The first one is by means of quaternions multiplications subjected to the Clifford algebra. In this algebra it is defined the operator ϵ with the properties $\epsilon \neq 0$ but $\epsilon^2 = 0$. Hence, given the dual quaternions $\underline{\mathbf{q}}_1 = \mathbf{q}_1 + \epsilon \mathbf{q}'_1$ and $\underline{\mathbf{q}}_2 = \mathbf{q}_2 + \epsilon \mathbf{q}'_2$, the multiplication $\underline{\mathbf{q}}_3 = \underline{\mathbf{q}}_1 \underline{\mathbf{q}}_2$ is

$$\underline{\mathbf{q}}_3 = \underline{\mathbf{q}}_1 \underline{\mathbf{q}}_2 = \mathbf{q}_1 \mathbf{q}_2 + \epsilon (\mathbf{q}_1 \mathbf{q}'_2 + \mathbf{q}'_1 \mathbf{q}_2). \quad (2)$$

Alternatively, the Hamilton operators used for quaternion multiplication in vector space can be extended to the dual domain [1], leading to

$$\bar{\underline{\mathbf{q}}}_3 = \overset{+}{\mathbf{H}}(\underline{\mathbf{q}}_1) \bar{\underline{\mathbf{q}}}_2 \quad (3)$$

$$= \bar{\mathbf{H}}(\underline{\mathbf{q}}_2) \bar{\underline{\mathbf{q}}}_1 \quad (4)$$

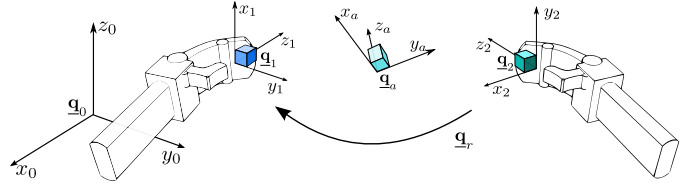


Figure 1: Cooperative dual task-space representation: $\underline{\mathbf{q}}_a$ and $\underline{\mathbf{q}}_r$ fully describe the manipulation task in terms of absolute and relative dual positions

We can note by (3) and (4) that the Hamilton operators provide a sort of “commutativity” for the mathematical operations describing a rigid motion, even if a sequence of rigid motions is not commutative. As we shall see in Section IV, we will exploit extensively this property in order to derive the control strategies for the cooperative manipulation.

III. COOPERATIVE DUAL TASK-SPACE

The equations used to describe the cooperative task-space in a vector space were first derived by Uchiyama and Dauchez [14] considering the hypothesis of a firmly grasped object. Then, in [5] this assumption was relaxed and the variables deduced by Uchiyama and Dauchez were used as a start point for defining the cooperative task-space. This space defines four physically meaningful variables in terms of absolute and relative position/orientation. Originally, Chiacchio et al. [5] used tridimensional vectors for representing the translations and rotation matrices for representing the orientations in the cooperative task-space. Then, Caccavale et al. [3] took one step further and used quaternions for representing the orientations. Here we go still further and use dual quaternions to represent both translations and rotations using just one operator.

Definition 1. The relative and absolute dual positions can be defined as

$$\underline{\mathbf{q}}_r = \underline{\mathbf{q}}_2^* \underline{\mathbf{q}}_1 \quad (5)$$

$$\underline{\mathbf{q}}_a = \underline{\mathbf{q}}_2 \underline{\mathbf{q}}_{\frac{r}{2}}, \quad (6)$$

where $\underline{\mathbf{q}}_2^*$ is the conjugate of $\underline{\mathbf{q}}_2$, $\underline{\mathbf{q}}_{\frac{r}{2}}$ is the transformation that corresponds to half of the angle ϕ about the axis \vec{u} of the quaternion $\underline{\mathbf{q}}_r$ and half of the translation between the two arms.

More specifically, $\underline{\mathbf{q}}_{\frac{r}{2}}$ is given by

$$\underline{\mathbf{q}}_{\frac{r}{2}} = \underline{\mathbf{q}}_{\frac{r}{2}} + \epsilon \frac{1}{4} \mathbf{t}_r \underline{\mathbf{q}}_{\frac{r}{2}} \quad (7)$$

where [3]

$$\underline{\mathbf{q}}_{\frac{r}{2}} = \cos \frac{\phi}{4} + \vec{u} \sin \frac{\phi}{4}. \quad (8)$$

Although in this paper we are not addressing the issue of the forces involved in the manipulation, it is important to remark that forces and moments can be represented directly in the

cooperative task-space [14]. Fig. 1 illustrates the cooperative dual task space variables, where $\underline{\mathbf{q}}_1$ and $\underline{\mathbf{q}}_2$ are the dual quaternions describing the rigid transformations of the frames attached to the first and second arms, respectively, with respect to a base frame represented by $\underline{\mathbf{q}}_0$, while $\underline{\mathbf{q}}_r$ and $\underline{\mathbf{q}}_a$ are the cooperative dual task space variables given by (5) and (6). There are two main advantages of using the dual quaternions to represent the two arms cooperation. Firstly, the task is defined using two equations instead of four [3]. Finally the ‘‘commutativity’’ achieved by using the Hamilton operators provides an easy way for defining different control strategies, as we will see in the next section.

IV. CONTROL STRATEGIES FOR DUAL POSITION CONTROL

Different tasks can be controlled differently. For instance, when driving a car both hands have a fixed geometrical relationship - thus we can imagine $\underline{\mathbf{q}}_r$ constant - but the absolute orientation $\underline{\mathbf{q}}_a$ must change in order to turn the wheel. Hence we can imagine a *full dual position control*, where we try to maintain $\underline{\mathbf{q}}_r$ and the absolute translation \mathbf{t}_a constant, and $\underline{\mathbf{q}}_a$ changes accordingly to the desired maneuver. On the other hand, let us suppose that we just want to drop a box handled by both arms. Raising the distance between them would suffice for accomplishing this task (if we consider that we are not grasping the box). Clearly, we can still drop the box by performing the control of every single variable of the cooperative dual task-space. Alternatively, we can act only on the relative translation between the arms or the relative dual position. However, controlling the relative distance requires only one degree of freedom, while the other task definitions can require up to twelve, as in the case of the *full dual position control* (six for the absolute variables plus six for the relative ones). Thus, in the case of *relative distance control*, the remaining degrees of freedom could be used to perform a secondary task, such as obstacle avoidance. However, a second task cannot be performed in the case of a robot that has less than twelve degrees of freedom and if the task is defined in terms of both absolute and relative dual positions. The conclusion is that, using the cooperative dual task-space formalism, we can accomplish the same task in different ways, but some of them are more convenient than others.

Motivated by this previous discussion we propose some control strategies using the cooperative task-space formalism. The idea is to provide some useful primitives that could be used by a higher level system and, as we shall see in the sequel, we exhaustively exploit the dual quaternion ‘‘commutativity’’ properties provided by the Hamilton operators in order to derive the relationship between these primitives and the joint variables. In this manner, we basically want to have a relationship between the primitive \vec{u}_d to be controlled and the joint variables of the two arms system $\vec{\theta}_R = \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix}$ given by

$$\dot{\vec{u}}_d = \mathbf{J}_{\text{task}} \dot{\vec{\theta}}_R, \quad (9)$$

where \mathbf{J}_{task} is the Jacobian associated to the task. Thus, our goal will be to find \mathbf{J}_{task} for each desired primitive and then

use it in any Jacobian based control method. Furthermore, we assume that the forward kinematic model is expressed in the dual quaternion space [12], and thus

$$\dot{\underline{\mathbf{q}}}_i = \mathbf{J}_{\underline{\mathbf{q}}_i} \dot{\theta}_i \quad (10)$$

where $i = 1, 2$ refers to the first and second arm and $\mathbf{J}_{\underline{\mathbf{q}}_i}$ is the analytical Jacobian.

1) *Relative dual position control*: The primitive to be controlled is the relative dual position, and thus $\vec{u}_d \triangleq \underline{\vec{\mathbf{q}}}_r$ and $\mathbf{J}_{\text{task}} \triangleq \mathbf{J}_{\underline{\mathbf{q}}_r}$. One example for this task definition is illustrated in Fig. 2a. $\mathbf{J}_{\underline{\mathbf{q}}_r}$ can be easily found taking the first derivative of (5) and using the Hamilton operators, that is

$$\dot{\underline{\mathbf{q}}}_r = \dot{\underline{\mathbf{q}}}_2^* \underline{\mathbf{q}}_1 + \underline{\mathbf{q}}_2^* \dot{\underline{\mathbf{q}}}_1, \quad (11)$$

$$= \bar{\mathbf{H}}(\underline{\mathbf{q}}_1) \dot{\underline{\mathbf{q}}}_2^* + \overset{+}{\mathbf{H}}(\underline{\mathbf{q}}_2^*) \dot{\underline{\mathbf{q}}}_1 \quad (12)$$

$$= \bar{\mathbf{H}}(\underline{\mathbf{q}}_1) \mathbf{J}_{\underline{\mathbf{q}}_2}^* \dot{\theta}_2 + \overset{+}{\mathbf{H}}(\underline{\mathbf{q}}_2^*) \mathbf{J}_{\underline{\mathbf{q}}_1} \dot{\theta}_1 \quad (13)$$

$$\dot{\vec{u}}_d \triangleq \dot{\underline{\vec{\mathbf{q}}}}_r = \underbrace{\begin{bmatrix} \overset{+}{\mathbf{H}}(\underline{\mathbf{q}}_2^*) \mathbf{J}_{\underline{\mathbf{q}}_1} & \bar{\mathbf{H}}(\underline{\mathbf{q}}_1) \mathbf{J}_{\underline{\mathbf{q}}_2}^* \end{bmatrix}}_{\mathbf{J}_{\underline{\mathbf{q}}_r}} \cdot \underbrace{\begin{bmatrix} \dot{\theta}_1 & \dot{\theta}_2 \end{bmatrix}}_{\dot{\vec{\theta}}_R}^T, \quad (14)$$

where

$$\mathbf{J}^* = \begin{cases} \text{Diag}(1, -1, -1, -1, 1, -1, -1, -1) \cdot \mathbf{J}, & \text{if } \mathbf{J}_{8 \times 8} \\ \text{Diag}(1, -1, -1, -1) \cdot \mathbf{J} & \text{if } \mathbf{J}_{4 \times 4} \end{cases} \quad (15)$$

that is, \mathbf{J}^* is an operator consisting of the dual quaternion Jacobian premultiplied by a diagonal matrix in order to relate the joint variables vector derivative with the conjugate dual quaternion derivative. More specifically, $\mathbf{J}_{\underline{\mathbf{q}}_i}^* = \mathbf{J}^* \mathbf{J}_{\underline{\mathbf{q}}_i}$, $i = 1, 2$. This same operator can be applied analogously to relate the joint variables vector derivative with the conjugate quaternion derivative.

2) *Relative Cartesian position control*: In this strategy, illustrated in Fig. 2b, $\vec{u}_d \triangleq \vec{\mathbf{t}}_r$ and $\mathbf{J}_{\text{task}} \triangleq \mathbf{J}_{\text{cartesian}}$. In order to find $\mathbf{J}_{\text{cartesian}}$, we isolate the translation quaternion from $\underline{\mathbf{q}}_r' = \frac{1}{2} \mathbf{t}_r \underline{\mathbf{q}}_r$ and take its first derivative

$$\mathbf{t}_r = 2 \underline{\mathbf{q}}_r' \underline{\mathbf{q}}_r^*, \quad (16)$$

$$\dot{\mathbf{t}}_r = 2 \dot{\underline{\mathbf{q}}}_r' \underline{\mathbf{q}}_r^* + 2 \underline{\mathbf{q}}_r' \dot{\underline{\mathbf{q}}}_r^*, \quad (17)$$

$$\dot{\vec{u}} \triangleq \dot{\vec{\mathbf{t}}}_r, \quad (18)$$

$$= \underbrace{\left(\overset{-}{2\mathbf{H}}(\underline{\mathbf{q}}_r') \mathbf{J}_{\underline{\mathbf{q}}_r'} + 2 \overset{+}{\mathbf{H}}(\underline{\mathbf{q}}_r') \mathbf{J}_{\underline{\mathbf{q}}_r}^* \right)}_{\mathbf{J}_{\text{cartesian}}} \dot{\vec{\theta}}_R, \quad (19)$$

where we decomposed the relative Jacobian such as $\mathbf{J}_{\underline{\mathbf{q}}} = \begin{bmatrix} \mathbf{J}_{\underline{\mathbf{q}}} \\ \mathbf{J}_{\underline{\mathbf{q}}'} \end{bmatrix}$.

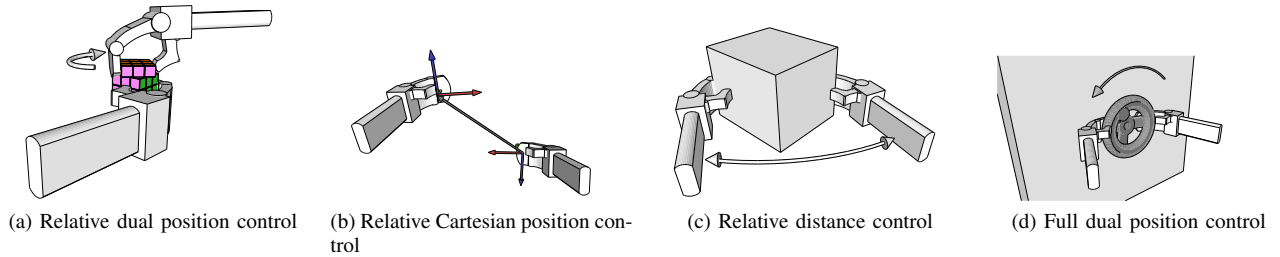


Figure 2: Different control strategies defined for different tasks

3) *Relative distance control*: In order to control the relative distance, as illustrated in Fig. 2c, firstly we write $u_d = f(\vec{t}_r)$, and then

$$\dot{u}_d = \frac{\partial u_d}{\partial \vec{t}_r} \cdot \frac{\partial \vec{t}_r}{\partial \vec{\theta}} \cdot \frac{d\vec{\theta}_R}{dt} \quad (20)$$

$$\stackrel{(19)}{=} \nabla u_d \mathbf{J}_{\text{cartesian}} \dot{\vec{\theta}}_R. \quad (21)$$

in which ∇u_d denotes the gradient of u_d .

The most obvious choice for u_d would be $u_d \triangleq \|\vec{t}_r\|$, but the gradient of this function is singular for $\vec{t}_r = \vec{0}$. Thus, we choose $u_d \triangleq \|\vec{t}_r\|^2$ and (21) becomes

$$\dot{u}_d = \underbrace{2\vec{t}_r^T \mathbf{J}_{\text{cartesian}}}_{\mathbf{J}_{\text{distance}}} \dot{\vec{\theta}}_R. \quad (22)$$

In a strict sense, (22) indicates that actually we control the square distance, but as the distance and its square are related by a bijective function, controlling the square of the relative distance is equivalent to control the relative distance.

4) *Full dual position control primitives*: In this strategy, illustrated in Fig. 2d, $\vec{u}_d \triangleq \begin{bmatrix} \vec{q}_a \\ \vec{q}_r \end{bmatrix}$ and $\mathbf{J}_{\text{task}} \triangleq \begin{bmatrix} \mathbf{J}_{\mathbf{q}_a} \\ \mathbf{J}_{\mathbf{q}_r} \end{bmatrix}$. The relative Jacobian $\mathbf{J}_{\mathbf{q}_r}$ is given by (14). In order to obtain $\mathbf{J}_{\mathbf{q}_a}$ we use an argument similar to the one used to derive $\mathbf{J}_{\mathbf{q}_r}$, that is, we take the first derivative of (6), then apply the Hamilton operators and finally use (10), leading to

$$\dot{\vec{q}}_a = \underbrace{\left(\bar{\mathbf{H}}(\mathbf{q}_{\frac{r}{2}}) \mathbf{J}_{\mathbf{q}_{\text{ext}}} + \overset{+}{\mathbf{H}}(\mathbf{q}_2) \mathbf{J}_{\mathbf{q}_{\frac{r}{2}}} \right)}_{\mathbf{J}_{\mathbf{q}_a}} \dot{\vec{\theta}}_R, \quad (23)$$

where $\mathbf{J}_{\mathbf{q}_{\text{ext}}} = \begin{bmatrix} \mathbf{0}_{8 \times n_1} & \mathbf{J}_{\mathbf{q}_2} \end{bmatrix}$ with n_1 being the number of joints of the first arm and $\dot{\vec{q}}_{\frac{r}{2}} = \mathbf{J}_{\mathbf{q}_{\frac{r}{2}}} \dot{\vec{\theta}}_R$. In order to find $\mathbf{J}_{\mathbf{q}_{\frac{r}{2}}}$, we take the first derivative of (7)

$$\dot{\vec{q}}_{\frac{r}{2}} = \left[\frac{1}{4} \left(\bar{\mathbf{H}}(\mathbf{q}_{\frac{r}{2}}) \dot{\vec{t}}_r + \overset{+}{\mathbf{H}}(\mathbf{t}_r) \dot{\vec{q}}_{\frac{r}{2}} \right) \right] \quad (24)$$

but we still have to calculate $\dot{\vec{q}}_{\frac{r}{2}}$. This can be done by using the quaternion propagation equation $\dot{\mathbf{q}} = \frac{1}{2} \boldsymbol{\omega} \mathbf{q}$ [11] for both $\dot{\mathbf{q}}_r$ and $\dot{\mathbf{q}}_{\frac{r}{2}}$ and after some quaternion algebra, we obtain

$$\dot{\vec{q}}_{\frac{r}{2}} = \frac{1}{2} \underbrace{\bar{\mathbf{H}}(\mathbf{q}_r^* \mathbf{q}_{\frac{r}{2}}) \mathbf{J}_{\mathbf{q}_r}}_{\mathbf{J}_{\mathbf{q}_{\frac{r}{2}}}} \dot{\vec{\theta}}_R. \quad (25)$$

Substituting (25) in (24) we have

$$\dot{\vec{q}}_{\frac{r}{2}} = \left[\frac{1}{4} \underbrace{\left(\bar{\mathbf{H}}(\mathbf{q}_{\frac{r}{2}}) \mathbf{J}_{\text{cartesian}} + \overset{+}{\mathbf{H}}(\mathbf{t}_r) \mathbf{J}_{\mathbf{q}_{\frac{r}{2}}} \right)}_{\mathbf{J}_{\mathbf{q}_{\frac{r}{2}}}} \right] \dot{\vec{\theta}}_R. \quad (26)$$

Finally, we can substitute (26) in (23) for obtaining the expanded expression for $\mathbf{J}_{\mathbf{q}_a}$.

A. Two arms control law

In order to perform the two arms control based on the previous derived primitives, the following classic control law can be applied

$$\dot{\vec{\theta}}_R = \mathbf{J}_{\text{task}}^+ \mathbf{K}_{\text{arm}} (\vec{u}_{\text{desired}} - \vec{u}_{\text{measured}}) \quad (27)$$

where \mathbf{K}_{arm} is a positive definite gain matrix and $\dot{\vec{\theta}}_R$ is integrated and passed directly as reference to the robot joints controller. In our case, we can calculate $\vec{u}_{\text{measured}}$ easily, since the forward kinematic model of the system is calculated directly in the dual quaternion space [12]. Thus, it is just a matter of calculating the forward kinematic model for both arms and then calculating $\vec{u}_{\text{measured}}$ using the relevant equations. For example, if we want to perform a *relative dual position control*, the controlled primitive is $\vec{u} = \vec{q}_r$, where \mathbf{q}_r is given by (5). On the other hand, if the goal is to control the relative Cartesian position $\vec{u} = \vec{t}_r$, we use (5) followed by (16), while the square of the relative distance can be obtained straightforwardly, that is, $u_d = \|\vec{t}_r\|^2$.

V. EXPERIMENTAL RESULTS AND DISCUSSIONS

In order to illustrate how our framework can be implemented in practice we have chosen two scenarios. In the first one, we have used the HOAP-3 robot for grabbing a balloon and in the second one the robot had to pour water in a glass,

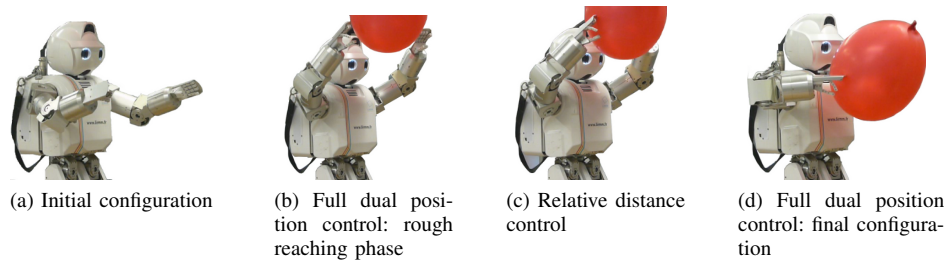


Figure 3: Grabbing a balloon

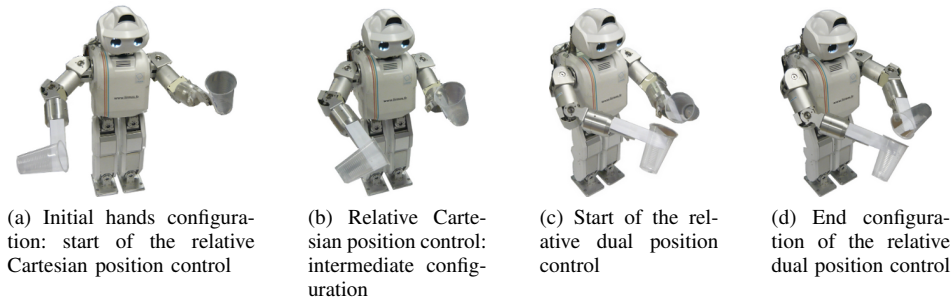


Figure 4: Pouring water

and we intend to show with these experiments how the control primitives can be chosen in order to accomplish the tasks. For both examples we have used the control law given by (27).

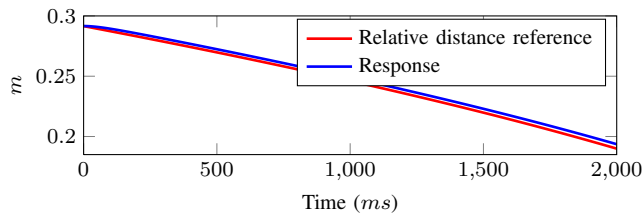
In the task of grabbing a balloon, given the initial configuration shown in Fig. 3a, we perform a *full dual position control*, since we use the current coordinate system of the balloon as the reference for the absolute frame and we also roughly specify the relative dual position in order to reach the object with a suitable pose. The final configuration after this rough reaching phase is illustrated in Fig. 3c. Then, a *relative distance control* is performed and the robot closes the arms and holds the balloon, as illustrated in Fig. 3c. Finally, in order to maintain the relative dual position and change the absolute dual position of the object, we perform again a *full dual position control*, as illustrated in Fig. 3d. The reference trajectories and the respective responses for this task are shown in Fig. 5. The full dual position plots refer to the final displacement subtask (the plots of the rough reaching phase are quite similar to this last one in terms of trajectory tracking). Since the unit quaternion is represented by $\mathbf{q} = \cos \frac{\theta}{2} + \sin \frac{\theta}{2} \vec{n}$, where \vec{n} is the rotation axis and θ is the angle of rotation about this axis, the orientation trajectories in Fig. 5 are represented by each component (n_x, n_y, n_z, θ) . We can note that the tracking errors in the *relative distance control* are quite negligible. In contrast, in the *full dual position control* the tracking errors of the absolute variables were larger due to the fact that the two arms system is underactuated for this task definition, since it requires twelve DOF and the robot has only four DOF per arm. Thus, as the control law given by (27) minimizes the error between the desired and measured variables, we should expect a worse performance, in terms of tracking errors, when executing the task using

an underactuated system. This is the exact behavior that we observe in Figs. 5b and 5c.

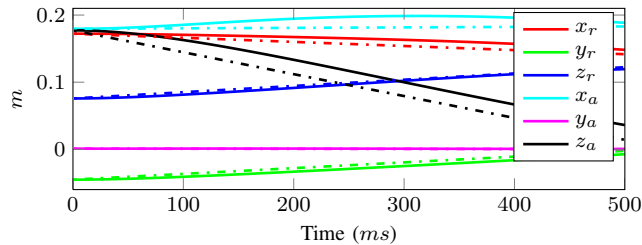
In the water pouring task the absolute position can be disregarded, since the only condition for an effective coordination is that it must occur inside the two arm workspace. Thus, if we consider the initial configuration indicated in Fig. 4a, we can first apply a *relative Cartesian position control* for the initial coordination between the hands, as we can see in Figs. 4b and 4c, and then we apply the *relative dual position control* strategy. In this way, as the distance between the hands remains constant, their relative orientation changes and we have the final configuration indicated in Fig. 4d. Thus, dividing the task into carefully chosen subtasks permits its execution even if the two arms system has less DOF than what would be required if all cooperative dual task-space variables were controlled. Moreover, as the system is now redundant with respect to the task, we can observe in Fig. 6 that the tracking errors are quite negligible.

VI. CONCLUSIONS AND FURTHER WORKS

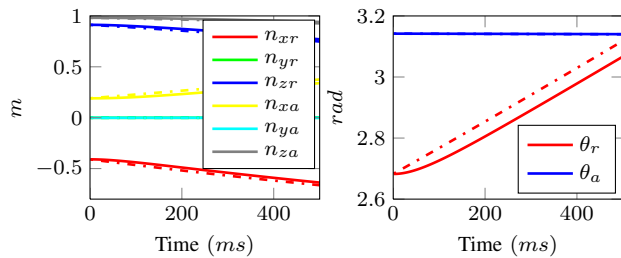
In this paper we have proposed a set of control strategies for performing two arm manipulation with the goal of simplifying the task definition. In order to develop these strategies we proposed a new representation, derived from the cooperative task-space, in the dual quaternion domain. The result is a compact and “singularity free” representation for two arm systems, named *cooperative dual task-space*. Our motivation in developing such representation originated in the context of the ASSIST project, where a high level of safety is required for the interaction between the robot and a quadriplegic person, and thus one should avoid inconsistent task executions due to singularities in the representation. All the proposed control strategies share the same general scheme and are derived by



(a) Relative distance control.



(b) Full dual position control: translation



(c) Full dual position control: orientation

Figure 5: Reference trajectories (dash-dotted lines) and responses (solid lines) for the task of grabbing a balloon

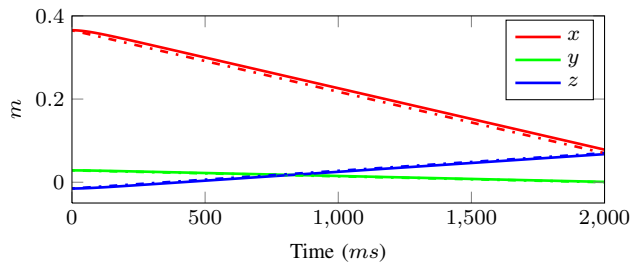


Figure 6: Reference trajectories (dash-dotted lines) and responses (solid lines) for the relative Cartesian position control in the task of pouring water

using an analytical approach. Since both forward kinematics model and the Jacobian are expressed directly in the dual quaternion space, all the mathematical treatment is given in a coherent and systematic fashion, and thus we hope that other strategies could be derived using the same argument. Finally, we have performed experiments on the HOAP-3 robot in order to validate the proposed techniques. These developments have shown to be promising in the geometric representation of the cooperation between two manipulators, as well in the associated kinematic control techniques. Currently, we are working towards the integration of a vision system into the task

execution, with the goal of replanning the task accordingly to the visual feedback. Moreover, further works have to be done to take into consideration the forces acting on the closed chain mechanism when an object is grasped by both arms. Hence, suitable control strategies for position/vision/force control in the cooperative dual task-space will be the next step in our research.

ACKNOWLEDGMENT

This work is supported by Agence Nationale de la Recherche (ANR) under grant ANR 07-ROBO-011 in the context of the ASSIST project.

REFERENCES

- [1] Bedia Akyar. Dual quaternions in spatial kinematics in an algebraic sense. *Turk. J. Math.*, 32(4):373–391, 2008.
- [2] O. Bottema and B. Roth. *Theoretical kinematics*, volume 24. North-Holland Publishing Company, 1979.
- [3] F. Caccavale, P. Chiacchio, and S. Chiaverini. Task-space regulation of cooperative manipulators. *Automatica*, 36(6):879–887, June 2000.
- [4] Fabrizio Caccavale and Masaru Uchiyama. *Handbook of Robotics*, chapter 29. Cooperative Manipulators, pages 701–718. Springer, 2008.
- [5] 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*, 118:691–697, 1996.
- [6] Thomas H. Connolly and Friedrich Pfeiffer. Cooperating manipulator control using dual quaternion coordinates. In *Proceedings of the 33rd IEEE Conference on Decision and Control, 1994*, volume 3, pages 2417–2418, Lake Buena Vista, FL, USA, December 1994.
- [7] J. R. Dooley and J. M. McCarthy. Spatial rigid body dynamics using dual quaternion components. In *Proceedings of the 1991 IEEE International Conference on Robotics and Automation*, pages 90–95, Sacramento, California, April 1991.
- [8] J.R. Dooley and J.M. McCarthy. On the geometric analysis of optimum trajectories for cooperating robots using dual quaternion coordinates. In *Proceedings of the IEEE International Conference on Robotics and Automation*, volume 1, pages 1031–1036, Atlanta, GA, USA, May 1993. Dept. of Mech. & Aersp. Eng., California Univ., Irvine, CA.
- [9] O. Khatib, K. Yokoi, K. Chang, D. Ruspini, R. Holmberg, and A. Casal. Decentralized cooperation between multiple manipulators. In *5th IEEE International Workshop on Robot and Human Communication*, pages 183–188, Tsukuba, Japan, 1996. Dept. of Comput. Sci., Stanford Univ., CA.
- [10] Oussama Khatib. Object manipulation in a multi-effector robot system. In *on The fourth international symposium*, pages 131–138, Cambridge, MA, USA, 1988. MIT Press.
- [11] F. Landis Markley. Attitude estimation or quaternion estimation? Technical report, Goddard Space Flight Center (NASA), 2003.
- [12] Hoang-Lan Pham, Véronique Perdereau, Bruno Vilhena Adorno, and Philippe Fraisse. Position and orientation control of robot manipulators using dual quaternion feedback. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2010.
- [13] J. De Schutter and H. van Brussel. Compliant robot motion II. A control approach based on external control loops. *Int. J. Rob. Res.*, 7(4):18–33, 1988.
- [14] M. Uchiyama and P. Dauchez. A symmetric hybrid position/force control scheme for the coordination of two robots. In *Proc. IEEE International Conference on Robotics and Automation*, pages 350–356 vol.1, 1988.
- [15] D. Williams and O. Khatib. The virtual linkage: a model for internal forces in multi-grasp manipulation. In *Proc. IEEE International Conference on Robotics and Automation*, pages 1025–1030 vol.1, 1993.
- [16] Yuanxin Wu, Xiaoping Hu, Dewen Hu, Tao Li, and Junxiang Lian. Strapdown inertial navigation system algorithms based on dual quaternions. *IEEE Transactions on Aerospace and Electronic Systems*, 41(1):110–132, 2005.
- [17] An Tzu Yang. *Application of quaternion algebra and dual numbers to the analysis of spatial mechanisms*. PhD thesis, Columbia University, 1963.