

EVALUATION OF DUAL QUATERNION BASED CONTROLLERS ON ANTHROPOMORPHIC COMPLIANT MANIPULATORS

MARCOS S. PEREIRA*, LUIS F. C. FIGUEREDO*, MARIANA C. BERNARDES*

**Laboratório de Robótica e Automação (LARA)*

*Departamento de Engenharia Elétrica, Universidade de Brasília
Brasília - DF, Brasil*

Emails: marcos.si.pereira@gmail.com, figueredo@lara.unb.br, bernardes@unb.br

Abstract— This work addresses the evaluation of dual quaternion based controllers at the A2 Arm from Meka Robotics robotic manipulator — a anthropomorphic manipulator with seven degrees of freedom. The efficiency, compactness, and lack of singularity of the representation render the unit dual quaternion a suitable framework for simultaneously describing the attitude and the position of the end effector. Motivated by the advantages of this kinematic description and by the challenge of controlling a compliant robot with higher precision, a task-space linear-quadratic optimal tracking controller (LQR) is tested in order to find an optimal trajectory for the end effector, providing a tool to balance more conveniently the end effector error and its task-space velocity. The proposed LQR allows further control of the trade-off between lower errors and a higher control effort at task-space. The LQR is compared with a proportional controller with and without a feed-forward term and with a decoupled proportional controller.

Keywords— optimal control, compliant robots, control effort, linear quadratic regulator.

Resumo— O presente artigo analisa o uso de controladores baseados em quatérnios duais no braço manipulador antropomórfico A2 Arm da Meka Robotics. A eficiência, simplicidade e ausência de singularidades na representação fazem dos quatérnios duais unitários uma ferramenta adequada para descrever simultaneamente orientação e posição do efetuador do robô. Motivados pelas vantagens desse modelo de descrição cinemática e pelo desafio de controlar robôs complacentes com maior precisão, foi testado um controlador regulador quadrático linear (RQL) rastreador de trajetória projetado no espaço de tarefas com o objetivo de obter uma trajetória ótima para o efetuador e ter um modo de balancear o erro de trajetória e as velocidades do efetuador. O RQL proposto permite um equilíbrio entre velocidades mais altas e menor erro, mas com maior esforço de controle. Comparamos o RQL com um controlador proporcional com e sem termo de antecipação e com um controlador proporcional desacoplado.

Palavras-chave— controle ótimo, robôs complacentes, esforço de controle, regulador quadrático linear.

1 Introduction

Planning and control techniques for robot manipulators are closely influenced by the robot type and task particularities. Stiff robotic manipulators are able to accomplish tasks at relatively low velocities and accelerations with the use of kinematic control strategies. Nonetheless, compliant robots designed to work with humans have lower stiffness and require new control strategies. The present study evaluate the use of a optimization-based kinematic controller at task-space for the anthropomorphic Meka Robotics A2 Arm (Figure 1) compliant manipulator. The optimal controller is compared with decoupled proportional gain feedback controller, and with dual quaternion based proportional controller with and without a feed-forward term.

The specification of robot tasks and the actuation take place in different spaces, the first occurs in the task-space and the later in the joint-space. This discrepancy turns difficult to use joint space control techniques since joint space descriptions must be retrieved from the robot task specifications [Khatib, 1995]. In other words, joint-space control techniques demand an external inverse kinematics (IK) solution to find the joint variables in terms of the end effector position and orienta-

tion. Nonetheless, there may or may not be a solution for the IK and there may even be an infinite number of solutions. Since modern manipulator robots are often equipped with low level controllers at joint level, task-space control techniques can be applied directly at the end effector — making direct use of the available low level controllers instead of redesigning them.



Figure 1: Meka Robotics A2 Arm at LARA

Another important point to consider is a proper representation for the task-space. The homogeneous transformation matrix (HTM) describe both rotation and translation simultaneously and is commonly used in the mathematical development of robotics textbooks [W. Spong

et al., 2005, Siciliano et al., 2009, Dombre and Khalil, 2007]. The HTM is described in matrix form and usually requires a proper parametrization which makes the control design harder. In addition, HTM are computationally demanding and make use of 12 free parameters. The use of dual quaternions over HTM handles coupled and singularity problems with less computational cost, represent all rotations with lesser parameters and eliminates the problem of singularities that is inherent to the use of minimal rotation representations [Adorno, 2011]. In the light of those benefits, there has been an increasing interest in works related to the design of kinematic controllers in dual quaternion space. Those works comprise rigid motion stabilization and tracking [Wang and Yu, 2013], robust control [Figueredo et al., 2013], multiple rigid body coordination [Wang et al., 2012], multiple arm manipulation [Figueredo et al., 2014], and human-robot interaction [Adorno et al., 2014].

The aforementioned control strategies and mathematical tools were previously investigated within a Matlab simulation environment using the Kuka LBR iiwa 7 R800 industrial robot. However, experimentation with real robots is still needed. In this sense, this work evaluates control techniques on the A2 Arm. The motivation of using an anthropomorphic compliant arm lies in their increasing use in the context of collaborative robots, human-robot interaction (HRI) and the need of testing existing control techniques within these scenarios.

2 Preliminaries

To present the notation used in this work we recall dual quaternions and their basic algebra when representing rigid transformations and then do a brief review of existing work in the kinematic control of robotic manipulators.

2.1 Dual Quaternions

Dual quaternion are dual numbers in which the primary and dual parts are quaternions. A general quaternion is composed by a real element and the quaternionic units $\hat{i}$, $\hat{j}$ and $\hat{k}$ such that $\hat{i}^2 = \hat{j}^2 = \hat{k}^2 = \hat{i}\hat{j}\hat{k} = -1$ and is described by $h = h_1 + \hat{i}h_2 + \hat{j}h_3 + \hat{k}h_4$. Its conjugate is defined as $h^* \triangleq h_1 - h_2\hat{i} - h_3\hat{j} - h_4\hat{k}$. The norm of a quaternion is $\|h\| = \sqrt{hh^*}$.

The rotation of an angle θ around an axis $\mathbf{n} = n_x\hat{i} + n_y\hat{j} + n_z\hat{k}$ can be described by the unit quaternion $\mathbf{r} = \cos(\frac{\theta}{2}) + \mathbf{n}\sin(\frac{\theta}{2})$ where $\|\mathbf{r}\| = 1$. A translation $\mathbf{t} = t_x\hat{i} + t_y\hat{j} + t_z\hat{k}$ can be combined with a rotation $\mathbf{r}$ to represent a unit dual quaternion rigid body motion as $\underline{h} = \mathbf{r} + \frac{1}{2}\varepsilon\mathbf{tr}$ in which ε is nilpotent dual unit; i.e., $\varepsilon^2 = 0$, but $\varepsilon \neq 0$.

Furthermore, the unit dual quaternion conjugate is $\underline{h}^* \triangleq \mathbf{r}^* + \frac{1}{2}\varepsilon(\mathbf{tr})^*$.

A general dual quaternion is composed of eight elements $\underline{g} = g_1 + g_2\hat{i} + g_3\hat{j} + g_4\hat{k} + \varepsilon(g_5 + g_6\hat{i} + g_7\hat{j} + g_8\hat{k})$. The *vec* operator is used to map it into an eight-dimensional column vector; i.e., $\underline{g} \triangleq [g_1, g_2, g_3, g_4, g_5, g_6, g_7, g_8]^T$. We also define the matrix $\mathbf{C}_8 \triangleq \text{diag}(1, -1, -1, -1, 1, -1, -1, -1)$ such that

$$\text{vec}\underline{h}^* = \mathbf{C}_8\text{vec}\underline{h}. \quad (1)$$

Finally, given dual quaternions $\underline{g}_1, \underline{g}_2$, the Hamilton operators $\bar{\mathbf{H}}(\cdot), \bar{\mathbf{H}}^+(\cdot)$ are transformation matrices satisfying the following relation [Adorno, 2011]:

$$\text{vec}(\underline{g}_1\underline{g}_2) = \bar{\mathbf{H}}^+(\underline{g}_1)\text{vec}\underline{g}_2 = \bar{\mathbf{H}}(\underline{g}_2)\text{vec}\underline{g}_1. \quad (2)$$

The set of dual quaternions $\mathbb{H} \otimes \mathbb{D}$ forms a group under dual quaternion multiplication [Adorno, 2011]. Quaternions and dual quaternions are associative and distributive, but non-commutative. The Hamilton operators are used as a way to describe the multiplication operation of the dual quaternion algebra using matrix form which allows to commute the terms [Adorno, 2011].

2.2 Kinematic Modeling & Error Definition

The forward kinematics model (FKM) of a serial manipulator robot (that is, the mapping between the n -dimensional vector of joint positions $\theta \in \mathbb{R}^n$ and the end effector pose $\underline{h}$) can be obtained directly in dual quaternion space using algebraic manipulations [Adorno, 2011]. In addition, the differential FKM (i.e., the mapping between the joint velocities $\dot{\theta} \in \mathbb{R}^n$ and the generalized end effector velocity $\text{vec}\dot{\underline{h}} \in \mathbb{R}^8$) is given by

$$\text{vec}\dot{\underline{h}} = \mathbf{J}\dot{\theta}, \quad (3)$$

where $\mathbf{J} \in \mathbb{R}^{8 \times n}$ is the manipulator Jacobian (which is also found algebraically [Adorno, 2011] and depends on robot joints configuration).

Since the spatial distance in dual quaternion space is described by $\underline{h}_e = \underline{h}^*\underline{h}_d$, then the invariant error function is given by [Figueredo et al., 2013]

$$\underline{e} = 1 - \underline{h}^*\underline{h}_d \quad (4)$$

Considering a time-varying reference $\underline{h}_d = \underline{h}_d(t)$, then the derivative of (4) is given by

$$\dot{\underline{e}} = -\dot{\underline{h}}^*\underline{h}_d - \underline{h}^*\dot{\underline{h}}_d. \quad (5)$$

3 Dual Quaternion Kinematic Control

This section describes the characteristics of classic controllers and their application in the context of trajectory tracking for manipulators and also presents a recent dual quaternion based optimal controller derived in [Marinho et al., 2015].

3.1 Proportional Controller

Since the manipulator differential kinematics (3) is a simple linear mapping, it is common practice to design closed-loop controllers based on the pseudoinverse of the Jacobian $\mathbf{J}$. Proportional controllers can exponentially reduce the error between the current pose $\underline{\mathbf{h}}$ and a desired pose $\underline{\mathbf{h}}_d$. Applying a least square minimization to

$$\left\| \text{vec} \underline{\mathbf{e}} - \mathbf{J} \dot{\underline{\boldsymbol{\theta}}} \right\|^2$$

[Maciejewski and Klein, 1988], we obtain the following control law

$$\dot{\underline{\boldsymbol{\theta}}} = \mathbf{K} \mathbf{J}^\dagger \text{vec} \underline{\mathbf{e}}, \quad (6)$$

where $\mathbf{K}$ is a positive definite matrix and $\mathbf{J}^\dagger$ is the pseudoinverse of $\mathbf{J}$. Although this control law can be easily derived, we must remember that when representing rigid motions using the dual quaternion, the reverse motion operation in unit dual quaternion space is not given by a subtraction, but by the multiplication by the conjugate. Hence, the error is calculated in the invariant error form as in (4). By rewriting (6) to use (4) and defining

$$\mathbf{N} \triangleq \bar{\mathbf{H}}(\underline{\mathbf{h}}_d) \mathbf{C}_8 \mathbf{J}, \quad (7)$$

we obtain the result of [Figueredo et al., 2013] of the dual quaternion based proportional controller ($\mathbf{K}$ controller) given by

$$\begin{cases} \mathbf{K} = k\mathbf{I}, \text{ where } k \in \mathbb{R}^+ \\ \dot{\underline{\boldsymbol{\theta}}} = \mathbf{N}^\dagger \mathbf{K} \text{vec} \underline{\mathbf{e}} \end{cases} \quad (8)$$

The arbitrary gain matrix $\mathbf{K}$ in (8) controls the rate of convergence. It is shown in [W. Spong et al., 2005] and in [Figueredo et al., 2013] that (8) converges for $\mathbf{K} > \mathbf{0}$ and that it is stable.

3.2 Proportional Controller with Feed-forward Term

The proportional controller yields an easy to implement and efficient solution for setpoint control. Still, it neglects the influence of a moving reference. To better address the context of an end effector tracking a desired trajectory instead of a fixed point, we add a feed-forward term (K+FF) to the proportional controller (8). Applying the *vec* operator to (5), we obtain

$$\underline{\dot{\mathbf{e}}} = \text{vec}(-\dot{\underline{\mathbf{h}}}^* \underline{\mathbf{h}}_d) - \text{vec}(\underline{\mathbf{h}}^* \dot{\underline{\mathbf{h}}}_d)$$

Using (1)-(3) and (7) we obtain

$$\begin{aligned} \underline{\dot{\mathbf{e}}} &= -\bar{\mathbf{H}}(\underline{\mathbf{h}}_d) \text{vec} \dot{\underline{\mathbf{h}}}^* - \text{vec}(\underline{\mathbf{h}}^* \dot{\underline{\mathbf{h}}}_d) \\ &= -\bar{\mathbf{H}}(\underline{\mathbf{h}}_d) \mathbf{C}_8 \mathbf{J} \dot{\underline{\boldsymbol{\theta}}} - \text{vec}(\underline{\mathbf{h}}^* \dot{\underline{\mathbf{h}}}_d) \\ &= -\mathbf{N} \dot{\underline{\boldsymbol{\theta}}} - \text{vec}(\underline{\mathbf{h}}^* \dot{\underline{\mathbf{h}}}_d). \end{aligned}$$

To obtain an exponentially stable closed loop system, we define the following dual quaternion based proportional controller with the feed-forward term,

$$\begin{cases} \mathbf{K} = k\mathbf{I}, \text{ in which } k \in \mathbb{R}^+ \\ \dot{\underline{\boldsymbol{\theta}}} = \mathbf{N}^\dagger (\mathbf{K} \underline{\mathbf{e}} - \text{vec} \underline{\mathbf{h}}^* \dot{\underline{\mathbf{h}}}_d). \end{cases} \quad (9)$$

Hence, $\underline{\dot{\mathbf{e}}} = k \underline{\mathbf{e}}$ where k defines the convergence rate.

3.3 Linear Quadratic Optimal Control (LQR)

In [Marinho et al., 2015], the optimal state-feedback is derived and its computation is discussed. It is shown that the kinematic control with a time-varying reference can be described as a linear time-varying system with an additive perturbation term. Instead of considering $\dot{\underline{\boldsymbol{\theta}}}$ the input signal for the system, we can consider as input the end effector velocity $\mathbf{u}$ using the mapping $\mathbf{u} = -\mathbf{N} \dot{\underline{\boldsymbol{\theta}}}$. This allows the optimization to be done in task-space variables. Consequently, there are the advantages of not requiring an external IK solution and of making direct use of the available robot low level controller instead of redesigning them.

The LQR exploits future knowledge of the desired trajectory, therefore, it calculates the control signals offline for a given desired trajectory. With time varying $\underline{\mathbf{h}}$ and $\underline{\mathbf{h}}_d$, the error derivative is given by (5). Hence,

$$\begin{aligned} \underline{\mathbf{e}} &= \mathbf{1} - \underline{\mathbf{h}}^* \underline{\mathbf{h}}_d \\ \implies \underline{\mathbf{e}} \underline{\mathbf{h}}_d^* &= \underline{\mathbf{h}}_d^* - \underline{\mathbf{h}}^* \\ \implies \underline{\mathbf{h}}^* &= \underline{\mathbf{h}}_d^* - \underline{\mathbf{e}} \underline{\mathbf{h}}_d^*. \end{aligned} \quad (10)$$

Using the *vec* operator on both sides of (5) and applying (10) to (5) yields

$$\underline{\dot{\mathbf{e}}} = -\bar{\mathbf{H}}(\underline{\mathbf{h}}_d) \mathbf{C}_8 \text{vec} \dot{\underline{\mathbf{h}}} + \bar{\mathbf{H}}(\underline{\mathbf{h}}_d^* \dot{\underline{\mathbf{h}}}_d) \underline{\mathbf{e}} - \text{vec}(\underline{\mathbf{h}}_d^* \dot{\underline{\mathbf{h}}}). \quad (11)$$

By defining $\mathbf{A} \triangleq \bar{\mathbf{H}}(\underline{\mathbf{h}}_d^* \dot{\underline{\mathbf{h}}}_d)$ and $\mathbf{c} \triangleq -\text{vec}(\underline{\mathbf{h}}_d^* \dot{\underline{\mathbf{h}}})$, using (3) and (7), we find

$$\dot{\underline{\mathbf{e}}} = -\bar{\mathbf{H}}(\underline{\mathbf{h}}_d) \mathbf{C}_8 \mathbf{J} \dot{\underline{\boldsymbol{\theta}}} + \mathbf{A} \underline{\mathbf{e}} + \mathbf{c} \quad (12)$$

$$\dot{\underline{\mathbf{e}}} = \mathbf{A} \underline{\mathbf{e}} - \mathbf{N} \dot{\underline{\boldsymbol{\theta}}} + \mathbf{c}. \quad (13)$$

Hence, we aim at finding the optimal controller for the affine time-varying system

$$\dot{\underline{\mathbf{e}}}(t) = \mathbf{A}(t) \underline{\mathbf{e}}(t) + \mathbf{u}(t) + \mathbf{c}(t) \quad (14)$$

where $\mathbf{u}(t) = -\mathbf{N} \dot{\underline{\boldsymbol{\theta}}}(t)$.

Therefore, from the error point-of-view, it is possible to solve the tracking problem for a continuous trajectory using a finite horizon LQR applied to a disturbed system, as the error disturbance caused by the time-varying trajectory is given by $\mathbf{c}(t)$. Other modeled continuous disturbances can also be grouped into $\mathbf{c}(t)$ and used in the same solution.

Consider that the manipulator has to track the trajectory during $t \in [0, t_f]$. Then, we seek to minimize the following cost function

$$F = \frac{1}{2} \mathbf{e}(t_f)^T \mathbf{S} \mathbf{e}(t_f) + \frac{1}{2} \int_0^{t_f} (\mathbf{e}^T \mathbf{Q} \mathbf{e} + \mathbf{u}^T \mathbf{R} \mathbf{u}) dt, \quad (15)$$

given the matrices $\mathbf{S}, \mathbf{Q}(t) \geq 0$ and $\mathbf{R}(t) > 0$ with $\mathbf{S}, \mathbf{Q}, \mathbf{R} \in \mathbb{R}^{8 \times 8}$. The matrix $\mathbf{S}$ is the weight of the final error norm, the time-varying matrix $\mathbf{Q}$ weighs the error cost along the trajectory, and the time-varying matrix $\mathbf{R}$ weighs the control effort in terms of end effector velocity norm. As long as $\mathbf{N}$ is well conditioned, an increase in $\mathbf{R}$ will also cause an overall decrease in joint velocities. The optimization of (15) leads to an optimal feedback without excessive expenditure of control energy while keeping the error $\mathbf{e}(t)$ near zero [Locatelli, 2001].

The solution for optimal controller is given by

$$\begin{cases} \dot{\mathbf{P}} = -\mathbf{P}\mathbf{A} - \mathbf{A}^T \mathbf{P} + \mathbf{P}\mathbf{R}^{-1} \mathbf{P} - \mathbf{Q} \\ \dot{\boldsymbol{\xi}} = -\mathbf{A}^T \boldsymbol{\xi} + \mathbf{P}\mathbf{R}^{-1} \boldsymbol{\xi} - \mathbf{P}\mathbf{c}. \end{cases} \quad (16)$$

where $\mathbf{u}(t) = -\mathbf{R}^{-1}(\mathbf{P}\mathbf{e} + \boldsymbol{\xi})$.

We solve (16) by finding the boundary conditions using the final time, t_f , of our trajectory. We set $\boldsymbol{\xi}(t_f) = 0$ to find the first boundary condition which yields $\mathbf{P}(t_f) = \mathbf{S}$. Note that $\mathbf{A} \triangleq \bar{\mathbf{H}}(\mathbf{h}_d^* \mathbf{h}_d)$ for all t . Hence, we can numerically solve the differential Riccati equation $\mathbf{P}(t)$ backwards in time. As $\mathbf{c} \triangleq -\text{vec}(\mathbf{h}\mathbf{h})$ is also known for all t and, with the solution of $\mathbf{P}(t)$, we can find $\boldsymbol{\xi}(t)$ by also solving it backwards in time. Applying as joint velocities we obtain the dual quaternion LQR based controller [Marinho et al., 2015]

$$\dot{\boldsymbol{\theta}} = \mathbf{N}^\dagger \mathbf{R}^{-1}(\mathbf{P}\mathbf{e} + \boldsymbol{\xi}), \quad (17)$$

4 Experiments Evaluation & Discussion

In order to show the controllers behavior under different set of parameters, two different tasks on the A2 Arm manipulator were devised. We choose a circle and a square trajectory. The controllers were implemented using the DQ_robotics¹ in C++ and using ROS² to establish communication with the robot.

The robot needs to accomplish repetitive tasks while working alongside humans. Therefore, we check the ability of the robot to follow simple trajectories. Due to anthropomorphic arm joint limitations we considered initial robot configurations which allows the robot to execute the whole trajectory.

The circle chosen has 10 cm radius and is drawn on the XZ-plane while maintaining Y constant. The robot initial configuration is given

by $\boldsymbol{\theta}_0 = [0, \pi/3, 0, \pi/2, 0, \pi/3, 0, 0]$. The second trajectory is a square on the XZ-plane with 10 cm side with Y constant and initial configuration given by $\boldsymbol{\theta}_0 = [0, \pi/4, 0, \pi/2, 0, \pi/4, 0, 0]$. They are independent trajectories and will be run separately.

In order to simplify the LQR choice of parameters, we define $s, q, r \in \mathbb{R}$ such that

$$\mathbf{S} = s\mathbf{I}, \mathbf{Q} = q\mathbf{I}, \quad \mathbf{R} = r\mathbf{I},$$

where $\mathbf{I} \in \mathbb{R}$ is an identity matrix.

The choice of parameters is closely related with the task optimization goals. For instance, we suppose that the end-time error is not of higher importance than the error in the remainder of the trajectory by setting $s = 0$. The choice of q is done in order to keep the system stable during the whole trajectory and is closely related with r . Because the r parameter weighs the control effort in terms of end effector velocity norm, for a higher r , the joint velocities will be lower and we need to compensate with a higher q , that is a ‘‘stronger’’ error correction demanding faster correction, in order to obtain an acceptable minimization overall. The controllers described in (8) and (9) make use only of the proportional gain $k \in \mathbb{R}$ such that $\mathbf{K} = k\mathbf{I}$ and $\mathbf{I} \in \mathbb{R}^{8 \times 8}$. The choice of k was done by trial and error elevating the k value until there would be no more improvement in the integral norm of the error without an excessive increase in the joint control effort.

For the K controller and the K+FF controller we run each trajectory five times and get the median value for each attribute. The circular trajectory was also executed with a decoupled proportional controller which regards rotation and translation of the robot end effector separately.

In order to compare performance, we distinguish the LQR controllers by choosing a increasing weight for the control effort parameter, that is $r \in \{0.01, 0.1, 0.5, 1\}$. Table 1 shows the results for the circular and squared trajectory, where *Err.* is the integral norm of the error, $\dot{\boldsymbol{\theta}}$ is the integral norm of the control signal, *Eff.* is the integral norm of the control effort, and *L.Vel.* and *L.Acc.* are the end effector velocity and acceleration integral norms, respectively.

In all controllers, the choice of q allows a well-behaved initial motion. The effect in decreasing r as shown in Table 1 yields a higher control effort and a slight increase in r results in a smaller control effort. Meanwhile, the K and K+FF controllers have higher control signals and it cannot be directly adjusted by specific parameters. Both are bounded to the overall effect of the proportional gain k .

The behavior of the integral error norm when using the LQR controller can also be seen in Figure 2. It shows the importance of adapting q for each r considering the desired goal. When q is

¹<http://dqrobotics.sourceforge.net/>

²<http://www.ros.org/>

Table 1: Experimental results on the A2 Arm robot performing two different end-effector trajectories.

	Circular trajectory							Squared trajectory			
	LQR				K	K+FF	Dec. K	LQR		K	K+FF
Q	20	250	350	80	k=50	k=50	k=0.375	20	250	k=50	k=50
R	0.01	0.1	0.5	1				0.01	0.1		
<i>Err.</i>	4.77	4.45	6.78	16.77	4.47	4.45	52.47	2.47	2.23	2.63	2.63
$\dot{\theta}$	7.82	8.16	6.31	5.46	8.24	8.22	22.71	3.82	4.42	4.99	4.96
<i>Eff.</i>	0.12	0.12	0.07	0.09	0.12	0.12	0.22	0.06	0.07	0.09	0.09
<i>L.Vel.</i>	1.06	1.06	0.62	0.94	1.07	1.05	1.20	0.51	0.58	0.68	0.68
<i>L.Acc.</i>	23.03	22.89	19.86	21.31	23.12	26.21	27.56	10.90	13.74	18.95	19.22

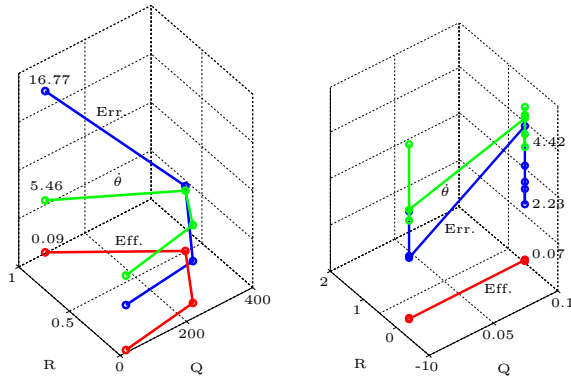


Figure 2: Q and R effects on error norm integral, control signals, and joints control effort during circular (left) and squared (right) trajectories.

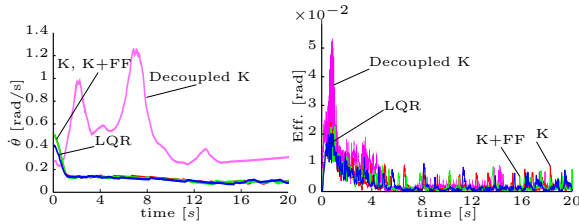


Figure 3: Circular trajectory: $r = 0.01$, $q = 20$

selected in accordance with r , it is possible to reduce control effort and get a lower error due to the q influence on error weight. The result of the squared trajectory seen in Table 1 gives an example of that.

Although the LQR did not have the lowest error norms, it did deliver in general the lowest end effector velocities and accelerations for a lower r (Table 1). The control effort is reduced with a higher r as expected when q is correctly chosen. This is graphically depicted in figures 3 and 4 for the circular and squared trajectory, respectively. The control signal $\dot{\theta}$ and the joints control effort norm at the beginning of the trajectory (near $t = 0$) are lower for the LQR. If r and q are reduced, the LQR controller yields a higher control signal and therefore higher joints control effort (Figures 3 and 5), but tend to lower the error as seen in Table 1 for the circular trajectory.

The Figure 3 shows that the LQR yields the lowest values and peaks as desired. In the light of using a real robot, the joint control effort yielded

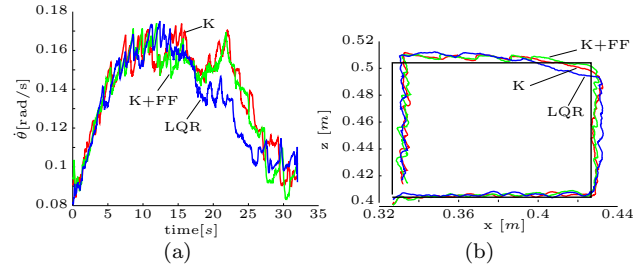


Figure 4: Squared trajectory results with $q = 20$ and $r = 0.01$. (a) Control signal norm, (b) Trajectory on XZ-plane.

will be much lower than the control output in order to prevent damage to the robot actuators. Joint velocities commanded by the controller are much greater than the angular velocities achieved by the robot joints (control effort) as seen in Figures 3 and 4 and in Table 1.

Lastly, the results of the circular trajectory contain the decoupled proportional controller. They show not only higher error, control effort and velocities as seen in Table 1, but can also lead the robot to singular configurations which may damage the actuators and joints.

5 Conclusion

In this paper, we evaluated trajectory control techniques in the task-space using dual quaternion based controllers and also a finite-time linear-quadratic optimal tracking controller (LQR). The optimization technique allows the designer to set optimal gains for a given trajectory in terms of control effort and error regulation. We provided experimental results using the A2 Arm from Meka Robotics 7 DOF compliant robot manipulator to illustrate the performance and effectiveness of the optimal criterion in comparison with a proportional gain controller and a proportional gain controller with a feed-forward term. Moreover, to illustrate the importance of a coupled — orientation and translation — representation and control criterion, we also compared the dual quaternion based controllers with a decoupled proportional controller.

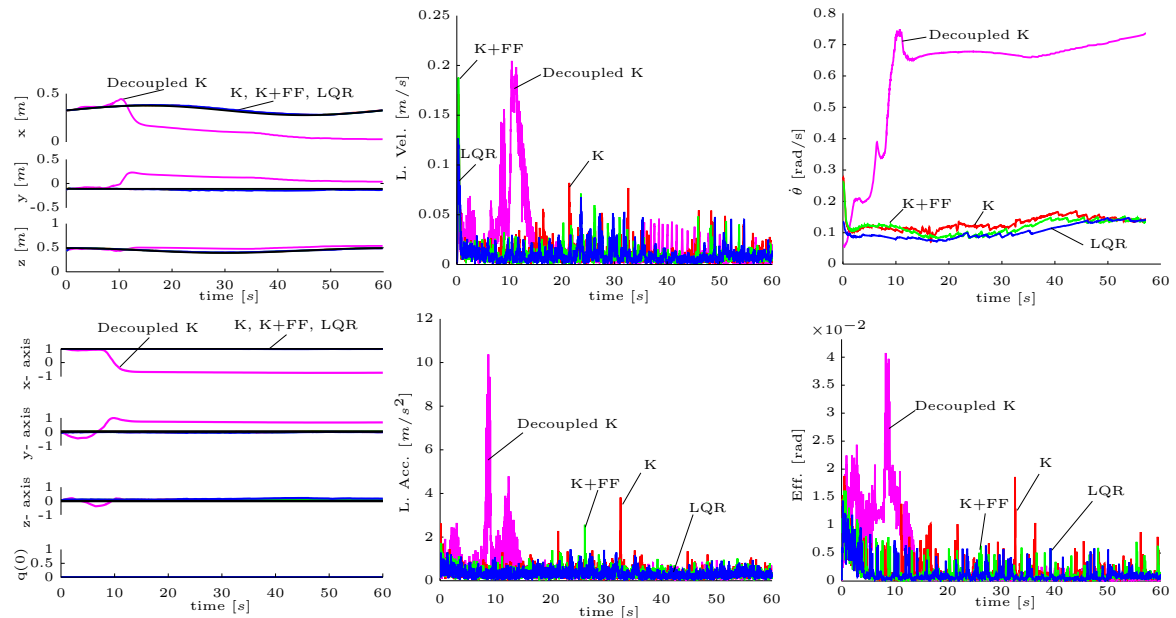


Figure 5: Circular trajectory with $r = 0.5$ and $q = 350$

References

- B. V. Adorno, A. P. L. Bó, and P. Fraisse. Kinematic modeling and control for human-robot co-operation considering different interaction roles. *Robotica*, 33(02):314–331, feb 2014.
- Bruno Vilhena Adorno. *Two-arm Manipulation: From Manipulators to Enhanced Human-Robot Collaboration*. PhD thesis, Université Montpellier 2, 2011.
- E. Dombre and W. Khalil. *Wiley: Robot Manipulators: Modeling, Performance Analysis and Control - Etienne Dombre, Wisama Khalil*. 2007.
- L.F.C. Figueredo, B.V. Adorno, J.Y. Ishihara, and G.A. Borges. Robust kinematic control of manipulator robots using dual quaternion representation. In *2013 IEEE International Conference on Robotics and Automation*, pages 1949–1955. IEEE, may 2013.
- L.F.C. Figueredo, B.V. Adorno, J.Y. Ishihara, and G.A. 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*, pages 1703–1709. IEEE, sep 2014.
- O. Khatib. Manipulator control at kinematic singularities: a dynamically consistent strategy. In *Proceedings IEEE/RSJ International Conference on Intelligent Robots and Systems. Human Robot Interaction and Cooperative Robots*, volume 3, pages 84–88. IEEE Comput. Soc. Press, 1995.
- Arturo Locatelli. *Optimal Control - An Introduction*, 2001.
- Anthony A. Maciejewski and Charles A. Klein. Numerical filtering for the operation of robotic manipulators through kinematically singular configurations. *Journal of Robotic Systems*, 5 (6):527–552, dec 1988.
- M. M. Marinho, L. F. C. Figueredo, and B. V. Adorno. A dual quaternion linear-quadratic optimal controller for trajectory tracking. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4047–4052. IEEE, sep 2015.
- B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo. *Robotics - Modelling, Planning and Control*. Springer, 2009.
- Mark W. Spong, Seth Hutchinson, and M. Vidyasagar. *Robot Modeling and Control - Mark W. Spong, Seth Hutchinson, M. Vidyasagar*. 2005.
- Xiangke Wang and Changbin Yu. Unit dual quaternion-based feedback linearization tracking problem for attitude and position dynamics. *Systems & Control Letters*, 62(3):225–233, mar 2013.
- Xiangke Wang, Changbin Yu, and Zhiyun Lin. A Dual Quaternion Solution to Attitude and Position Control for Rigid-Body Coordination. *IEEE Transactions on Robotics*, 28(5):1162–1170, oct 2012.