Mechatronics 55 (2018) 171–179

Contents lists available at ScienceDirect

Mechatronics

journal homepage: www.elsevier.com/locate/mechatronics

Hierarchical tracking task control in redundant manipulators with compliance control in the null-space☆

Abbas Karami[a], Hamid Sadeghian[b,*], Mehid Keshmiri[a], Giuseppe Oriolo[c]

[a] Department of Mechanical Engineering, Isfahan University of Technology, Isfahan 84156-83111, Iran
[b] Engineering Department, University of Isfahan, Isfahan 8174673441, Iran
[c] Dipartimento di Ingegneria Informatica, Automatica e Gestionale, Sapienza Università di Roma, Rome, Italy

ARTICLE INFO

Keywords:
Redundant manipulators
Prioritized control
Multi-task tracking
Physical interaction
Disturbance observer
Null-space compliance

ABSTRACT

In this paper, a new approach for dealing with multiple tracking tasks during physical interaction is proposed. By using this method, multiple tasks are accomplished based on the assigned priority in addition to a compliant behavior in the null-space of the main tasks. This issue is critical when robots are employed for complex manipulation in unknown environments and in the presence of human. During the manipulation in the dynamic environments, different objects may collide with the robot body and disturb its manipulation. In these cases, the robot is expected to continue execution of the tasks, accurately. Meanwhile, the robot should be compliant to ensure the safety during the interaction. A nonlinear controller-observer is proposed for tracking the desired trajectory based on a preallocated hierarchy. The suggested controller-observer estimates the external torques applied to the robot body without using joint torque measurements and compensates its projection on the task spaces. Asymptotic stability of the task space errors, the null-space velocity and the external interaction estimation error during accomplishing multiple tracking tasks are shown analytically. Finally, the algorithm performance is shown through experiments on a 7-DOF KUKA LWR robot arm.

1. Introduction

Using robots near human for complicated manipulations usually needs simultaneous execution of multiple operational tasks. Minimally Invasive Surgery by surgical robots [1] and domestic services by humanoid robots [2] are two examples of the considered scenarios. Different methods can be used for handling multiple robotic tasks according to an allocated priorities (e.g. [3–8]). The accomplishment of the tasks using these methods are different. Most of the common approaches for priority allocation employ null-space projectors for projecting lower priority tasks to the null-space of the higher priority tasks. These methods promise to fulfill the desired hierarchy [6,9–11].

Currently, the prevailing formulation for managing multiple tasks at velocity and acceleration level is the one proposed in [3]. This method is used for handling prioritized tasks in robots within various control approaches [9,10,12,13]. Alternative solutions for inverse kinematic problem of redundant robots are suggested in [4,5].

Robots can be controlled in their task space by several approaches (*e.g.* [14,15]). However, Operational Space Formulation (OSF) is the most common one. Following OSF method, decoupled dynamics for each task can be realized. Decoupling task dynamics is useful for handling different tasks [10,16] and inspecting the system stability [2,17,18].

Stability of the robot system depends on the null-space behaviour [19,20]. Nakanishi et al. in [21] has compared multiple controllers which are used for accomplishing tasks using OSF. It is reported in [21] that the null-space stability is an unresolved issue for all these methods. In [6], a new hierarchical formulation is proposed which can be used for investigating stability in the null-space along with multiple position regulation tasks. Here, a new intuitive Jacobian is employed for the case of multiple tasks execution. This Jacobian is computed by exploiting the null-space base matrix. Decoupled dynamics in each level can be realized by using this Jacobian. This characteristics allows to handle various tasks separately according to the allocated priorities. In comparison with the method proposed in [6], current research uses more clear and intuitive null-space projection with less computation burden. Moreover, it is not limited to the regulation tasks and can be employed for tracking tasks in any arbitrary level as well.

An example of possible physical interaction scenarios is depicted in Fig. 1. Safe human-robot coexistence and cooperation can be ensured by proper strategies for avoiding and/or handling physical interactions. In this regard, different methods can be found in previous works such as

☆ This paper was recommended for publication by Associate Editor Prof Cesare Fantuzzi.

* Corresponding author.
E-mail address: h.sadeghian@eng.ui.ac.ir (H. Sadeghian).

https://doi.org/10.1016/j.mechatronics.2018.09.005
Received 5 March 2018; Received in revised form 8 September 2018; Accepted 18 September 2018
Available online 1 October 2018

Fig. 1. Possible accidental human-robot interaction scenario [17].

[17,22,23]. Robot redundancy can be exploited effectively for this goal. Exteroceptive sensors are used for obstacle avoidance in [9,24]. In both of these works, redundant degrees of freedom are used to find the safest joint space trajectory for the main task while an obstacle approaches to the robot body.

Since collision avoidance can not be guaranteed because of the visual limitations, sensitive skin [25–27] and observers [28] are employed for collision detection. Several reaction strategies based on observer data are introduced in [29,30]. In the previous works by the authors of this paper [17], an impedance behavior is imposed in the null-space of the main (regulation) task to ensure safety and accuracy, simultaneously. Current research extends the previous work to multiple tracking tasks by rigorous stability analysis to ensure efficiency in most of the possible cases. More issues about physical human robot interaction and applications can be found in [31].

Recently, in [2,17,32] tracking problem in redundant manipulators is considered. The proposed controllers accomplish tracking of the task without considering any external physical interaction. On the other hand, regulation of the task space position around a desired point during physical interaction with robot body were studied in our previous work [17]. The position tracking problem is a more practical and useful manipulation task whose controller design and stability analysis is far complicated. This problem is investigated here within a multi priority scheme by considering physical interactions and model inaccuracy. To the best of our knowledge, handling external interaction with a desired compliance in the null-space during trajectory tracking is not possible through previous prioritization approaches.

Therefore, each of the previous methods miss one or more of the following issues: considering accidental physical interaction and model inaccuracy, guaranteeing stability and desired compliance in the null-space, ensuring zero task space error, and accomplishing multiple prioritized tasks including tracking. As the main state of the art, by exploiting the priority allocation method besides the proposed controller-observer for tracking tasks and null-space compliance, the whole system stability is proven in the presence of external physical interactions.

The main contributions of the current study are as follows:

- Clear representation of the robot dynamics in multiple levels including null-space with minimal states using a new Jacobian definition is given.
- Multiple tracking tasks are attainable in any arbitrary hierarchy besides desired compliance in the null-space.
- Proper controller-observer is suggested by considering accidental physical interactions and model inaccuracy.
- Stability analysis is performed by using nested conditional stability analysis for time varying systems in multi subsets along with invariance like theorem.

The rest of the paper is organized as follows: in Section 2 some fundamental issues about controlling the redundant robots are reviewed. The task allocation method as well as controller design are addressed in Sections 3 and 4, respectively. The system stability analysis is discussed in Section 5. Experimental results on 7-DOF KUKA LWR4 robot are reported in Section 6. Finally, the paper is concluded in Section 7.

2. Preliminaries

The kinematics relationship between task space and joint space of a robot is given as

$$\dot{\boldsymbol{x}} = \boldsymbol{J}(\boldsymbol{q})\dot{\boldsymbol{q}}, \tag{1}$$

where $\dot{\boldsymbol{x}} \in I\!R^m$ and $\dot{\boldsymbol{q}} \in I\!R^n$ represent the task space and joint space velocity vectors, respectively. The Jacobian matrix is denoted by $\boldsymbol{J}(\boldsymbol{q}) \in I\!R^{m\times n}$ and for redundant manipulators $m<n$. For a given $\dot{\boldsymbol{x}}$, (1) has infinite solutions for joint space command acceleration $\dot{\mathbf{q}}_{\mathbf{c}}$ in a redundant manipulator which can be obtained by

$$\dot{\boldsymbol{q}}_c = \boldsymbol{J}^{\dagger}(\boldsymbol{q})\dot{\boldsymbol{x}} + \boldsymbol{P}(\boldsymbol{q})\boldsymbol{\zeta}. \tag{2}$$

$\boldsymbol{J}\dagger(\boldsymbol{q})$ is the pseudo-inverse of the Jacobian matrix given by $\boldsymbol{J}^{\dagger} = \boldsymbol{J}^T(\boldsymbol{J}\boldsymbol{J}^T)^{-1}$, $\boldsymbol{P}(\boldsymbol{q}) \in I\!R^{n\times n}$ denotes a projection matrix to the null-space of $\boldsymbol{J}(\boldsymbol{q})$ and $\boldsymbol{\zeta} \in I\!R^n$ represents the commanded joint space velocity in the null-space of the main task.[1] A common definition for the projection matrix is $\boldsymbol{P} = \boldsymbol{I} - \boldsymbol{J}^{\dagger}\boldsymbol{J}$. Using (2), we obtain a joint space velocity which satisfies (1) and is close to $\boldsymbol{\zeta}$ in the sense of least square.

Eq. (2)can be extended to l tasks as

$$\dot{\boldsymbol{q}}_c = \sum_{i=1}^{l} \dot{\boldsymbol{q}}_{i,c},$$

$$\dot{\boldsymbol{q}}_{i,c} = \begin{cases} \bar{\boldsymbol{J}}_1^{\dagger}\dot{\boldsymbol{x}}_{1,c} & i = 1 \\ \bar{\boldsymbol{J}}_i^{\dagger}(\dot{\boldsymbol{x}}_{i,c} - \boldsymbol{J}_i\dot{\boldsymbol{q}}_{i-1}) & i = 2, \ \ldots, l \end{cases}, \tag{3}$$

where,

$$\bar{\boldsymbol{J}}_i = \boldsymbol{J}_i\boldsymbol{P}_{i-1}. \tag{4}$$

$\bar{\boldsymbol{J}}_i \in I\!R^{m_i\times n}$ and $\dot{\boldsymbol{x}}_i \in I\!R^{m_i}$ represent the i-th task Jacobian matrix and velocity vector where m_i is the task dimension [3]. The null-space projection matrices in the case of multiple tasks can be computed as [13],

$$\boldsymbol{P}_i = \prod_{j=1}^{i}(\boldsymbol{I} - \bar{\boldsymbol{J}}_j^{\dagger}\bar{\boldsymbol{J}}_j), \qquad \boldsymbol{P}_0 = \boldsymbol{I}. \tag{5}$$

2.1. Multi priority control at acceleration level

The dynamic equation of a robot with n-link is written as

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{g}(\boldsymbol{q}) = \boldsymbol{\tau} - \boldsymbol{\tau}_{ext}, \tag{6}$$

where $\boldsymbol{M}(\boldsymbol{q}) \in I\!R^{n\times n}$ is the inertia matrix, $\boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}}) \in I\!R^{n\times n}$ is the matrix include both Coriolis and centrifugal effects and $\boldsymbol{g}(\boldsymbol{q}) \in I\!R^n$ and $\boldsymbol{\tau}_{ext} \in I\!R^n$ are gravity and external torques vector, respectively.

[1] Dependencies on $\boldsymbol{q}$ and $\dot{\boldsymbol{q}}$ are usually shown at the first parameter introduction and omitted elsewhere for the sake of clarity.

The resolution of the joint acceleration for a given task in a redundant robot is usually more accurate but elaborating in comparison with joint velocity synthesis. Due to explicit incorporation of acceleration information, this formulation improves the tracking ability. Moreover force and impedance control algorithms can be easily applied within that. Hence, it is possible to track the desired trajectory accurately and handle physical interactions with the robot body compliantly, simultaneously (see for example [33]).

Based on the relation (1), the solution for second order inverse kinematics is

$$\ddot{\boldsymbol{q}}_c = \boldsymbol{J}^{\dagger}(\ddot{\boldsymbol{x}} - \dot{\boldsymbol{J}}\dot{\boldsymbol{q}}) + \boldsymbol{P}\boldsymbol{\zeta}, \tag{7}$$

and in the case of multi task system, $\ddot{\boldsymbol{q}}_c$ is obtained as

$$\ddot{\boldsymbol{q}}_c = \sum_{i=1}^{l} \ddot{\boldsymbol{q}}_{i,c},$$

$$\ddot{\boldsymbol{q}}_{i,c} = \bar{\boldsymbol{J}}_i^{\dagger}(\ddot{\boldsymbol{x}}_{i,c} - \dot{\boldsymbol{J}}_i\dot{\boldsymbol{q}} - \boldsymbol{a}_i),$$

$$\boldsymbol{a}_i = \begin{cases} \boldsymbol{0} & i = 1 \\ \boldsymbol{J}_i \sum_{j=1}^{i-1} \ddot{\boldsymbol{q}}_{j,c} & i = 2, \dots, l \end{cases}, \tag{8}$$

where $\ddot{\boldsymbol{x}}_c$ is the command acceleration in the ith task space. Joint command acceleration $\ddot{\boldsymbol{q}}_c$ can be used to obtain driving torques by

$$\boldsymbol{\tau} = \boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}}_c + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{g}(\boldsymbol{q}) + \boldsymbol{\tau}_{est}, \tag{9}$$

where the estimation of the external torque is shown by $\boldsymbol{\tau}_{est}$. The closed-loop behavior of the system is then obtained as

$$\ddot{\boldsymbol{q}} = \ddot{\boldsymbol{q}}_c. \tag{10}$$

Multiplying both sides of (10) with Jacobian matrix, the ith subtask closed-loop equation is realized as [13],

$$\bar{\boldsymbol{J}}_i\bar{\boldsymbol{J}}_i^{\dagger}[\ddot{\boldsymbol{x}}_{i,c} - \ddot{\boldsymbol{x}}_i] = 0. \tag{11}$$

2.2. Stability of multi tasks system

When the manipulator is in non-singular configuration, using task space command acceleration as

$$\ddot{\boldsymbol{x}}_c = \ddot{\boldsymbol{x}}_d + \boldsymbol{k}_{i,d}\dot{\tilde{\boldsymbol{x}}} + \boldsymbol{k}_{i,p}\tilde{\boldsymbol{x}}, \tag{12}$$

in (11), with $\boldsymbol{k}_{i,d} \in I\!R^{m_i \times m_i}$ and $\boldsymbol{k}_{i,p} \in I\!R^{m_i \times m_i}$ as positive definite matrices, implies asymptotic tracking for the ith task. To ensure null-space stability, the command acceleration is suggested as

$$\ddot{\boldsymbol{q}}_c = \boldsymbol{J}^{\dagger}(\ddot{\boldsymbol{x}}_c - \dot{\boldsymbol{J}}\dot{\boldsymbol{q}}) + \boldsymbol{P}(\ddot{\boldsymbol{q}}_{c_N} - \dot{\boldsymbol{P}}\dot{\boldsymbol{q}}), \tag{13}$$

where

$$\ddot{\boldsymbol{q}}_{c_N} = \ddot{\boldsymbol{q}}_d + \boldsymbol{K}(\dot{\boldsymbol{q}}_d - \dot{\boldsymbol{q}}). \tag{14}$$

The desired velocity in the null-space is $\dot{\boldsymbol{q}}_d = \boldsymbol{P}\boldsymbol{\xi}$ [13] and the closed-loop null-space dynamics is obtained as

$$\boldsymbol{P}(\dot{\boldsymbol{e}}_N + \boldsymbol{K}\boldsymbol{e}_N) = \boldsymbol{0}, \tag{15}$$

where $\boldsymbol{e}_N = \boldsymbol{P}(\boldsymbol{\xi} - \dot{\boldsymbol{q}})$, $\boldsymbol{\xi}$ is the desired velocity in the null-space and $\boldsymbol{K} \in I\!R^{n \times n}$ denotes the control gain. $\boldsymbol{P} \in I\!R^{n \times n}$ is a null-space projection matrix and is not full rank (more details about the rank of the projection matrices can be found in the second chapter of [34]). Because of the rank deficiency, it is not possible to guarantee that the tracking is accomplished perfectly in the null-space. Meanwhile, it is possible to prove the stability of the null-space dynamics represented by (15) through (*c.f.*, [13,19])

$$V = \frac{1}{2}\boldsymbol{e}_N^T\boldsymbol{e}_N. \tag{16}$$

Considering of external torques are necessary for controlling the robots in dynamic and unpredictable environments and is not involved in the methods mentioned in the current section. The stability of the system with physical interaction is just studied for regulation tasks in [6].

3. Problem formulation using prioritized scheme

In this section, a novel approach for assigning the task priority is introduced and exploited for computing prioritized dynamics of the manipulator. The main issues that are considered in the new task allocation method are as follows; lower priority tasks should not disturb any higher priority tasks, tracking tasks should be attainable in any hierarchical level and null-space behavior should be described with minimal coordinates. Hence, one can obtain a complete and clear representation of the robot dynamics in the coordinates correlated to the prioritized tasks.

This hierarchy scheme can be achieved by choosing the projected Jacobian of the tasks as

$$\bar{\boldsymbol{J}}_i = \begin{cases} \boldsymbol{J}_1, & i = 1 \\ \boldsymbol{J}_i\boldsymbol{Z}_{i-1}^T(\boldsymbol{Z}_{i-1}\boldsymbol{M}\boldsymbol{Z}_{i-1}^T)^{-1}\boldsymbol{Z}_{i-1}\boldsymbol{M}, & i = 2, \dots, l-1 \\ (\boldsymbol{Z}_{l-1}\boldsymbol{M}\boldsymbol{Z}_{l-1}^T)^{-1}\boldsymbol{Z}_{l-1}\boldsymbol{M}, & i = l \end{cases} \tag{17}$$

where $l-1$ tasks are considered for the manipulator and the lth space is the remaining null-space. $\boldsymbol{Z}_{i-1}(\boldsymbol{q})$ denotes the null-space base matrix of an augmented Jacobian matrix defined as

$$\boldsymbol{J}_{aug,i-1} = \begin{bmatrix} \bar{\boldsymbol{J}}_1 \\ \bar{\boldsymbol{J}}_2 \\ \vdots \\ \bar{\boldsymbol{J}}_{i-1} \end{bmatrix}, \tag{18}$$

for $(i-1)$th level. Note that $\boldsymbol{Z}_{i-1}$ should fulfil

$$\boldsymbol{J}_{aug,i-1}\boldsymbol{Z}_{i-1}^T = \boldsymbol{0}, \quad i = 1, \dots, l. \tag{19}$$

Representing the matrices in (19) by appropriate partitioned forms $\boldsymbol{J}_{aug,i-1} = [\boldsymbol{J}_{w,i-1}\ \boldsymbol{J}_{v,i-1}]$ and $\boldsymbol{Z}_{i-1} = [\boldsymbol{Z}_{w,i-1}\ \boldsymbol{I}]$ with invertible $\boldsymbol{J}_{w,i-1}$, the full rank null-space base matrix is obtained as

$$\boldsymbol{Z}_{i-1} = [-\boldsymbol{J}_{v,i-1}^T\boldsymbol{J}_{w,i-1}^{-T}\ \boldsymbol{I}]. \tag{20}$$

The method is discussed in details in [35]. Singular value decomposition can be used alternatively for computing the null-space base matrix [36].

Introducing $\boldsymbol{\nu} = \boldsymbol{Z}_{l-1}\dot{\boldsymbol{q}}$, where $\boldsymbol{\nu} \in I\!R^{n-\sum_{i=1}^{l-1} m_i}$ is the null-space velocity vector, the task space velocity vector can be written as $\dot{\boldsymbol{x}} = (\dot{\boldsymbol{x}}_1, \ \dots, \dot{\boldsymbol{x}}_{l-1}, \boldsymbol{\nu})$. Dynamic consistency [14] is promised by employing Jacobian pseudo-inverse matrix as

$$\bar{\boldsymbol{J}}_i^{\#} = \begin{cases} \boldsymbol{M}^{-1}\bar{\boldsymbol{J}}_i^T(\bar{\boldsymbol{J}}_i\boldsymbol{M}^{-1}\bar{\boldsymbol{J}}_i^T)^{-1} & i = 1, \dots, l-1 \\ \boldsymbol{Z}_{l-1}^T & i = l. \end{cases} \tag{21}$$

Using (17–19), one can obtain

$$\begin{cases} \boldsymbol{J}_1\boldsymbol{Z}_1^T = 0, & i = 1 \\ \boldsymbol{J}_j\boldsymbol{Z}_{j-1}^T(\boldsymbol{Z}_{j-1}\boldsymbol{M}\boldsymbol{Z}_{j-1}^T)^{-1}\boldsymbol{Z}_{j-1}\boldsymbol{M}\boldsymbol{Z}_i^T = \boldsymbol{0}, & i = 2, \dots, l \end{cases}, \tag{22}$$

for $j \le i$ which ensures

$$\boldsymbol{J}_{aug_{i-1}}\bar{\boldsymbol{J}}_i^{\#} = \boldsymbol{0}, \tag{23}$$

and

$$\bar{\boldsymbol{J}}_j\boldsymbol{M}^{-1}\bar{\boldsymbol{J}}_i^T = \boldsymbol{0} \quad \text{for any } i \text{ and } j. \tag{24}$$

Herein, relation (23) guarantees the order of priority and (24) promises diagonal task space inertial matrix (\Lambda) and dynamic consistency as [6]

$$\Lambda = diag(\Lambda_1, \Lambda_2, \dots, \Lambda_l) \tag{25}$$

On the other hand, by considering (23), the recursive formula for command acceleration is obtained as

$$\ddot{\boldsymbol{q}}_c = \sum_{i=1}^{l} \ddot{\boldsymbol{q}}_{i,c},$$

$$\ddot{\boldsymbol{q}}_{i,c} = \bar{\boldsymbol{J}}_i^{\#}(\ddot{\boldsymbol{x}}_{i,c} - \dot{\boldsymbol{J}}_i\dot{\boldsymbol{q}}). \tag{26}$$

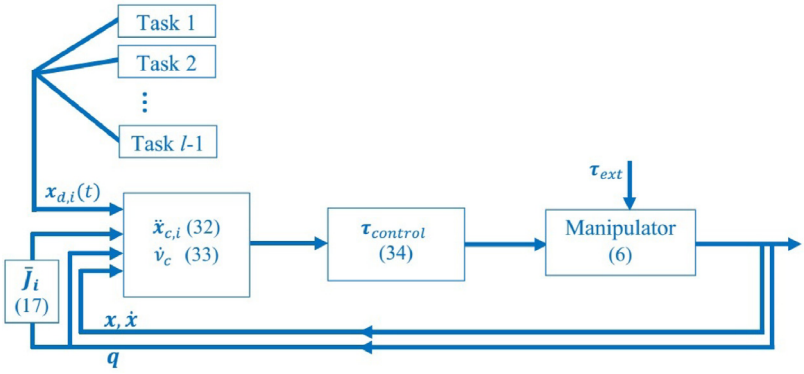

Fig. 2. The system block diagram.

Using (17) besides the OSF method, the task space dynamics in all levels are realized as

$$\Lambda_i \ddot{x}_i + \mu_i + h_i = \bar{J}_i^{\#T} \tau_{control} - \bar{J}_i^{\#T} \tau_{ext}, \tag{27}$$

where

$$\begin{aligned} &\Lambda_i = (\bar{J}_i M^{-1} \bar{J}_i^T)^{-1}, \\ &\mu_i = \bar{J}_i^{\#T} C(q, \dot{q})\dot{q} - \Lambda_i \dot{\bar{J}}_i \dot{q}, \\ &h_i = \bar{J}_i^{\#T} g(q), \end{aligned} \tag{28}$$

for $1 \le i \le l$.

When τ_{est} is not available through sensor or observer, by employing (26) in (9) and substituting in (27) the task space dynamics in the non-singular configuration simplifies as

$$\ddot{x}_i = \ddot{x}_{i,c} - \bar{J}_i M^{-1} \tau_{ext}. \tag{29}$$

It can be seen that the external physical interactions affect the robot manipulation in all the hierarchal levels. It is possible to reduce the interaction influence on the manipulation tasks using external interaction observer.

Our main goals in the next sections are to design proper controller-observer for accurate position trajectory tracking in any arbitrary priority level besides compliance and stable behavior in the null-space.

4. Controller design

In this section, a controller-observer for trajectory tracking during physical interactions is proposed. A momentum based observer originally proposed in [28] is exploited to obtain an accurate estimation of the external torques. The main idea for momentum based observer is to construct the residual vector r using robot momentum $p = M\dot{q}$ as

$$r(t) = K_I [p(t) - \int_0^t (\tau + C^T(q, \dot{q})\dot{q} - g(q) + r(\sigma)) d\sigma], \tag{30}$$

where $p(0) = 0$, $r(0) = 0$ and $K_I \in IR^{n \times n}$ is positive definite matrix. Thus the residual vector dynamics is given by

$$\dot{r} = -K_I r - K_I \tau_{ext}. \tag{31}$$

According to (31), r is a filtered version of τ_{ext} and one can realize $\tau_{ext} \approx -\mathbf{r}$. Hence, $-\mathbf{r}$ can be used as estimated external torque, i.e., in (9).

The controller action aims at multiple simultaneous trajectory tracking on "$l-1$" levels besides a compliance behaviour on the null-space of the main tasks. The following proposition holds for various constant and time varying trajectories $x_{i,d}(t) \in IR^{m_i}$ and its proof is given later in Section 5.

Proposition 1. *Denoting the ith task desired trajectory as $x_{i,d}(t)$, the command acceleration for this task is given by*

$$\ddot{x}_{i,c} = \ddot{x}_{i,d} + K_{i,d}\dot{\tilde{x}}_i + K_{i,p}\tilde{x}_i - \Lambda_i^{-1} \bar{J}_i^{\#T} r, \quad 1 \le i < l, \tag{32}$$

together with the null-space command acceleration

$$\dot{v}_c = \dot{v}_d + \Lambda_l^{-1}((\mu_l + K_{l,d})\tilde{v} - Z_l K_{l,p}\tilde{q}), \tag{33}$$

guarantees that $\tilde{x}_i$, $\dot{\tilde{x}}_i$, $\tilde{v}$, $\tilde{q}$ and external interaction estimation error $\tilde{r} = r + \tau_{ext}$ converges to zero, asymptotically. In recent relations, $\tilde{x}_i = x_{i,d} - x_i$, $\tilde{v} = v_d - v$, $\tilde{q} = q_{l,d} - q$ and $q_{l,d}$ is a properly chosen desired configuration in the joint space which will be discussed later. $K_{i,d} \in IR^{m_i \times m_i}$, $K_{i,p} \in IR^{m_i \times m_i}$, $K_{l,d} \in IR^{m_l \times m_l}$ and $K_{l,p} \in IR^{n \times n}$ are symmetric positive definite matrices.

It is noteworthy that since the velocity vector v is in general non-integrable [37], for the null-space command acceleration the projected joint space error ($Z_l K_{l,p}\tilde{q}$) is used instead of the null-space position error [38].

The command torque is then computed as

$$\begin{aligned} \tau_{control} = &\sum_{i=1}^{l-1} \bar{J}_i^T (\Lambda_i(\ddot{x}_{i,c} - \dot{\bar{J}}_i \dot{q})) + \bar{J}_l^T (\Lambda_l(\dot{v}_c - \dot{\bar{J}}_l \dot{q}))+ \\ &C(q, \dot{q}) + g(q), \end{aligned} \tag{34}$$

and the corresponding system block diagram is shown in Fig. 2. By observing (21) and (28), it is possible to show that the control torque computed through (9) together with (26) coincides with (34).

5. Stability analysis

This section includes the details of the stability analysis of the system using the controller-observer proposed in Section 4. The stability proof of the overall control algorithm is based on the concept of conditional stability and invariance-like theorem. For the purpose of this paper, both of the theorems are reviewed in the following.

Theorem 1 ([39]:). *Suppose that $\dot{z} = f(t, z)$ is locally Lipschitz in $z \in IR^n$, uniform in $t \ge 0$, and $f(t, 0) = 0$. Meanwhile, $\alpha \in \mathcal{K}$ and $\mathcal{K}$ is the set of real-valued functions that are continuous, zero at zero and strictly increasing. If there exists a function $V(t, z) \in C^1$ in a neighborhood Ω of the origin such that*

- *$V(z) \ge 0$ for all $z \in \Omega$ and $V(0) = 0$.*
- *$\dot{V}(t, z) \le -\alpha V(t, z)$ for all $z \in \Omega$ and all $t \ge 0$;*
- *on the largest positively invariant set $\mathcal{A}$ contained in $\{z \in \Omega | V(z) = 0\}$ the equilibrium $z = 0$ is uniformly asymptotically stable,*

then, the origin is asymptotically stable.

Theorem 2 ([40]:). *Let $D \subset IR^n$ be a domain containing $z = 0$ and suppose that $f(t, z)$ is piecewise continuous in t and locally Lipschitz in z for all $t \ge 0$ and $z \in D$. Let $z = 0$ be an equilibrium point for $\dot{z} = f(z, t)$ at $t = 0$ and V: $[0, \infty] \times D \to IR$ be a continuously differentiable function such that*

- $W_1(z) \le V(t, z) \le W_2(z)$, (35)
- $\dot{V}(t, z) = \frac{\partial V}{\partial z} f(t, z) + \frac{\partial V}{\partial z} \le 0$, (36)

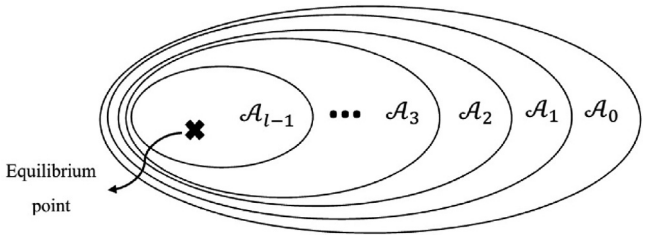

Fig. 3. Nested sets used for the stability proof in the state space. $\mathcal{A}_0$ is the complete and unrestricted state space $\boldsymbol{z}$. By employing the conditional stability theorem for the ith step, a new subset $\mathcal{A}_i$ is created where the next step of the stability analysis is performed inside this subset.

- $V(t+\delta, \phi(t+\delta; t, \boldsymbol{z})) - V(t, \boldsymbol{z}) \leq -\lambda V(t, \boldsymbol{z}), \quad 0 \leq \lambda \leq 1,$ (37)

$\forall t \geq 0, \forall \boldsymbol{z} \in D$, *for some* $\delta > 0$. $W_1(\boldsymbol{z})$ *and* $W_2(\boldsymbol{z})$ *are continuous positive definite functions on* D *and* $\phi(\tau; t, \boldsymbol{z})$ *denotes the solution of the system that starts at* $(t, \boldsymbol{z})$. *The origin is then uniformly asymptotically stable.*

Proof of Proposition 1.. By using the command accelerations (32) and (33), the task space and the null-space closed-loop dynamics are obtained as

$$\ddot{\tilde{\boldsymbol{x}}}_i + \boldsymbol{K}_{i,d}\dot{\tilde{\boldsymbol{x}}}_i + \boldsymbol{K}_{i,p}\tilde{\boldsymbol{x}}_i = \boldsymbol{\Lambda}_i^{-1}\bar{\boldsymbol{J}}_i^{\#T}\tilde{\boldsymbol{r}} \quad i = 1, \ldots, l-1, \tag{38}$$

and

$$\boldsymbol{\Lambda}_l\dot{\tilde{\boldsymbol{\nu}}} + (\boldsymbol{\mu}_l + \boldsymbol{K}_{l,d})\tilde{\boldsymbol{\nu}} + \boldsymbol{Z}_l\boldsymbol{K}_{l,p}\tilde{\boldsymbol{q}} = \boldsymbol{Z}_l\boldsymbol{\tau}_{ext}. \tag{39}$$

The residual dynamics (31) can also be written as

$$\dot{\tilde{\boldsymbol{r}}} + \boldsymbol{k}_I\tilde{\boldsymbol{r}} = \dot{\boldsymbol{\tau}}_{ext}. \tag{40}$$

Consequently, the state vector of the system is $\boldsymbol{z} = (\tilde{\boldsymbol{x}}_1, \dot{\tilde{\boldsymbol{x}}}_1, \ldots, \tilde{\boldsymbol{x}}_{l-1}, \dot{\tilde{\boldsymbol{x}}}_{l-1}, \tilde{\boldsymbol{r}}, \tilde{\boldsymbol{\nu}}, \tilde{\boldsymbol{q}})$.

In the following, the system stability is studied in multiple steps according to the *Theorem* 1. At the first step, the observer dynamics is considered for the stability analysis. Let the unknown external torque to be constant (or slowly time-varying). The *Lyapunov* like function candidate

$$V_1(\boldsymbol{z}) = \frac{1}{2}\tilde{\boldsymbol{r}}^T\boldsymbol{K}_I{}^{-1}\tilde{\boldsymbol{r}}, \tag{41}$$

is then positive semi-definite in the whole state space. The time derivative of (41) along the system trajectory is

$$\dot{V}_1(\boldsymbol{z}) = -\tilde{\boldsymbol{r}}^T\tilde{\boldsymbol{r}}. \tag{42}$$

By choosing proper $\boldsymbol{K}_I$, we realize $-\tilde{\boldsymbol{r}}^T\tilde{\boldsymbol{r}} \leq -\alpha\frac{1}{2}\tilde{\boldsymbol{r}}^T\boldsymbol{K}_I{}^{-1}\tilde{\boldsymbol{r}}$. Therefore, the first two conditions in *Theorem* 1 are fulfilled and asymptotic stability of the subset $\mathcal{A}_1 = \{\boldsymbol{z} \in \mathcal{A}_0 | V_1(\boldsymbol{z}) = 0\}$ must be shown which includes multiple position tracking task space states. To this end, *Theorem* 1 will be applied iteratively for each of these tasks inside a nested subsets as shown in Fig. 3. By employing *Lyapunov equation*[2] for (38), the function

$$\begin{aligned} V_i(\boldsymbol{z}) = &\frac{1}{2}\tilde{\boldsymbol{x}}_i^T(\boldsymbol{K}_{i,d}\boldsymbol{K}_{i,p}{}^{-1} + \boldsymbol{K}_{i,d}{}^{-1}\boldsymbol{K}_{i,p} + \boldsymbol{K}_{i,d}{}^{-1})\tilde{\boldsymbol{x}}_i + \\ &\frac{1}{2}\dot{\tilde{\boldsymbol{x}}}_i^T(\boldsymbol{I} + \boldsymbol{K}_{i,p}{}^{-1})\boldsymbol{K}_{i,d}{}^{-1}\dot{\tilde{\boldsymbol{x}}}_i + \dot{\tilde{\boldsymbol{x}}}_i^T\boldsymbol{K}_{i,p}{}^{-1}\tilde{\boldsymbol{x}}_i. \end{aligned} \tag{43}$$

is suggested for the ith task where $1 \leq i \leq l-1$. The time derivative of (43) along the system trajectory is then

$$\dot{V}_i(\boldsymbol{z}) = -\dot{\tilde{\boldsymbol{x}}}_i^T\dot{\tilde{\boldsymbol{x}}}_i - \tilde{\boldsymbol{x}}_i^T\tilde{\boldsymbol{x}}_i. \tag{44}$$

By selection of appropriate controller gains, $\dot{V}_i(\boldsymbol{z}) \leq -\alpha V_i(\boldsymbol{z})$ can be fulfilled. In each iteration of employing *Theorem* 1, the current subset is restricted to $V_i(\boldsymbol{z})$ and the new subset is realized as $\mathcal{A}_{i+1} = \{\boldsymbol{z} \in \mathcal{A}_i | V_i(\boldsymbol{z}) = 0\}$. The last subset is obtained as $\mathcal{A}_l = \{\tilde{\boldsymbol{r}} = \boldsymbol{0}, \dot{\tilde{\boldsymbol{x}}}_1 = \boldsymbol{0}, \tilde{\boldsymbol{x}}_1 = \boldsymbol{0}, \ldots, \dot{\tilde{\boldsymbol{x}}}_{l-1} = \boldsymbol{0}, \tilde{\boldsymbol{x}}_{l-1} = \boldsymbol{0}, \tilde{\boldsymbol{\nu}}, \tilde{\boldsymbol{q}}^*\}$ where $\boldsymbol{q}^*$ is defined based on

$$\boldsymbol{K}_{r,p}\tilde{\boldsymbol{q}} - \boldsymbol{\tau}_{ext} = \boldsymbol{K}_{r,p}\tilde{\boldsymbol{q}}^*, \tag{45}$$

and $\boldsymbol{Z}_l$ is full rank (see more in [41]). In order to show the stability of the system, the asymptotic stability in $\mathcal{A}_l$ must be shown. The *Lyapunov* function candidate for the last step is proposed as

$$V_l(\boldsymbol{z}) = \frac{1}{2}\tilde{\boldsymbol{\nu}}^T\boldsymbol{\Lambda}_l\tilde{\boldsymbol{\nu}} + \frac{1}{2}\tilde{\boldsymbol{q}}^{*T}\boldsymbol{K}_{l,p}\tilde{\boldsymbol{q}}^*, \tag{46}$$

where its time derivative along the system trajectory is

$$\begin{aligned} \dot{V}_l(\boldsymbol{z}) = &-\tilde{\boldsymbol{\nu}}^T(\boldsymbol{K}_{l,d})\tilde{\boldsymbol{\nu}} - \tilde{\boldsymbol{\nu}}^T\boldsymbol{Z}_{l-1}\boldsymbol{K}_{l,p}\tilde{\boldsymbol{q}} + \tilde{\boldsymbol{\nu}}^T\boldsymbol{Z}_{l-1}\boldsymbol{\tau}_{ext} \\ &+ \dot{\tilde{\boldsymbol{q}}}^{*T}\boldsymbol{K}_{l,p}\tilde{\boldsymbol{q}}^*. \end{aligned} \tag{47}$$

Meanwhile, the robot inverse kinematic is

$$\dot{\boldsymbol{q}} = \bar{\boldsymbol{J}}^{\#}_{aug_{l-1}}\dot{\boldsymbol{x}} + \boldsymbol{Z}_{l-1}^T\boldsymbol{\nu}, \tag{48}$$

and within subset $\mathcal{A}_l$, all the previous task errors and their derivatives are zero, and thus $\bar{\boldsymbol{J}}^{\#}_{aug_{l-1}}\dot{\boldsymbol{x}} = \bar{\boldsymbol{J}}^{\#}_{aug_{l-1}}\dot{\boldsymbol{x}}_d$ holds. Considering (45) and the constant (or slowly time varying) $\boldsymbol{\tau}_{ext}$, one can obtain

$$\dot{\boldsymbol{q}}^* = \dot{\boldsymbol{q}}_{l,d}. \tag{49}$$

From (19) and (22) we see that

$$\bar{\boldsymbol{J}}_l\boldsymbol{J}^{\#}_{aug,l-1} = \boldsymbol{0}. \tag{50}$$

By choosing $\boldsymbol{q}_{l,d}(t) = \int_0^t \bar{\boldsymbol{J}}^{\#}_{aug_{l-1}}(\boldsymbol{q})\dot{\boldsymbol{x}}_d dt$, one can obtain

$$\begin{aligned} \tilde{\boldsymbol{\nu}} &= \boldsymbol{\nu}_d - \boldsymbol{\nu} \\ &= \bar{\boldsymbol{J}}_l\dot{\boldsymbol{q}}_{l,d} - \boldsymbol{\nu} \\ &= \bar{\boldsymbol{J}}_l\boldsymbol{J}^{\#}_{aug,l-1}\dot{\boldsymbol{x}}_d - \boldsymbol{\nu} \\ &= -\boldsymbol{\nu}, \end{aligned} \tag{51}$$

and

$$\begin{aligned} \dot{\tilde{\boldsymbol{q}}}^* &= \dot{\boldsymbol{q}}^* - \dot{\boldsymbol{q}} \\ &= \dot{\boldsymbol{q}}_{l,d} - \dot{\boldsymbol{q}} \end{aligned}$$

[2] Let the system $\dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x}$ and suppose that $\boldsymbol{A}$ is a *Hurwitz matrix*. If a symmetric positive definite matrix $\boldsymbol{P}$ can be found which satisfies $\boldsymbol{P}\boldsymbol{A} + \boldsymbol{A}^T\boldsymbol{P} = -\boldsymbol{I}$, then the time derivative of the positive definite function $V = \boldsymbol{x}^T\boldsymbol{P}\boldsymbol{x}$ along the system trajectory is negative definite [40]

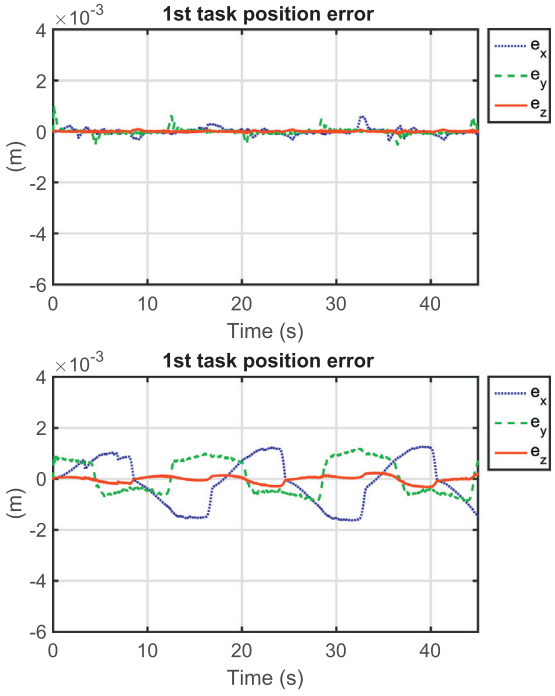

Fig. 4. Two levels hierarchy tracking performance without external interaction: Using (32) and (33) (Top) and using (12)–(14) (Bottom).

Fig. 5. Circular path tracking: Snapshots of the experimental setup during manipulation.

$$= \boldsymbol{J}^{\#}_{aug,l-1}\dot{\boldsymbol{x}}_d - \left(\overline{\boldsymbol{J}}^{\#}_{aug_{l-1}}\dot{\boldsymbol{x}}_d + \boldsymbol{Z}^T_{l-1}\boldsymbol{v}\right)$$
$$= -\boldsymbol{Z}^T_{l-1}\boldsymbol{v}. \tag{52}$$

Consequently, (47) is simplified as

$$\dot{V}_l(\boldsymbol{z}) = -\tilde{\boldsymbol{v}}^T(\boldsymbol{K}_{l,d})\tilde{\boldsymbol{v}}, \tag{53}$$

which implies that $\tilde{\boldsymbol{v}}$ and $\tilde{\boldsymbol{q}}^*$ are bounded. Furthermore, setting $\delta = \lambda \neq 0$ in *Theorem* 2, (37) can be rewritten as

$$\dot{V}(t, \boldsymbol{y}) \leq -V(t, \boldsymbol{y}). \tag{54}$$

By properly choosing $\boldsymbol{K}_{l,p}$ and $\boldsymbol{K}_{l,d}$, (54) can be fulfilled for V_l and $\dot{V}_l$ as

$$-\tilde{\boldsymbol{v}}^T(\boldsymbol{K}_{l,d})\tilde{\boldsymbol{v}} \leq -(\frac{1}{2}\tilde{\boldsymbol{v}}^T\boldsymbol{\Lambda}_l\tilde{\boldsymbol{v}} + \frac{1}{2}\tilde{\boldsymbol{q}}^{*T}\boldsymbol{K}_{l,p}\tilde{\boldsymbol{q}}^*). \tag{55}$$

Eventually, according to *Theorem* 2 the subset $\mathcal{A}_l$ is uniformly asymptotically stable. Hereafter, using *Theorem* 1 asymptotic stability can be shown successively for $\mathcal{A}_i$ from $i = l-1$ to $i = 1$. Therefore the stationary point $\boldsymbol{z} = \{\tilde{\boldsymbol{r}} = \boldsymbol{0}, \dot{\tilde{\boldsymbol{x}}}_1 = \boldsymbol{0}, \tilde{\boldsymbol{x}}_1 = \boldsymbol{0}, \ \ldots, \dot{\tilde{\boldsymbol{x}}}_{l-1} = \boldsymbol{0}, \tilde{\boldsymbol{x}}_{l-1} = \boldsymbol{0}, \tilde{\boldsymbol{v}} = \boldsymbol{0}, \tilde{\boldsymbol{q}}^* = 0\}$ is asymptotically stable. □

6. Experimental evaluation

The proposed approach is verified experimentally on a 7-DOF KUKA LWR IV robot arm. Each experimental case comprises multiple priority levels where position controls are considered at higher priority levels and compliance control is realized at lower priority level. The robot joint torque is commanded using the Fast Research Interface (FRI) libraries with 2 ms sampling time through a remote PC. The dynamic parameters of the KUKA LWR IV, employed in this work, are based on the model reported in [42] and [43].

6.1. Two levels hierarchy

In this set of experiments, a translational Cartesian path is considered as the main task ($m = 3$) at the first level of hierarchy. Hence, the robot has 4 degrees of redundancy which is exploited for impedance behavior at the second level of hierarchy. The desired trajectory for the end effector is a circle. The performance of the system under the control action (32) and (33) as well as (12)–(14) are compared together without any physical interaction.

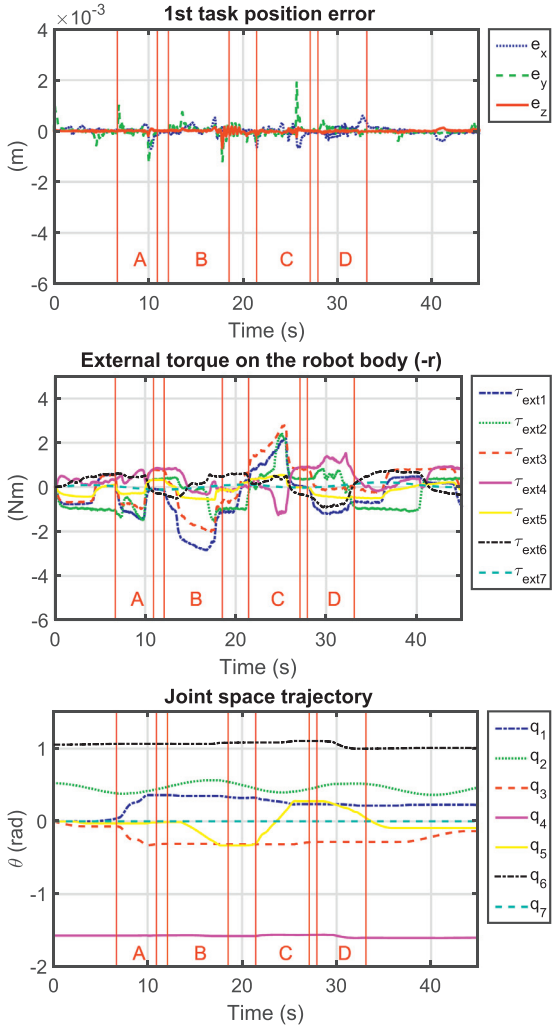

Fig. 6. Two levels hierarchy: The tracking performance of a circular path using proposed controller-observer: Tracking error (Top), Estimated external torque applied to the robot body (Middle) and the joint space trajectory (Bottom).

As mentioned in Section 2, in the case of no physical interaction, the robot is stable using (12)–(14). The parameters in (32) and (12) are identical and the same diagonal matrices are used for $\boldsymbol{K}$ in (14) and $\boldsymbol{K}_{l,d}$ in (33). The tracking error for both cases are illustrated in Fig. 4. It can be seen that the tracking performance is significantly improved using the new proposed algorithm.

Circular trajectory tracking is repeated with physical interaction which is applied by the human hand to the various points of the robot body using (32) and (33) (see Fig. 5). It is noteworthy that in real application this force may be applied to the robot body accidentally by any environment entities including humans.

The performance of the controller is illustrated in the Fig. 6. In this figure, four intervals are shown by "A", "B", "C" and "D". During these intervals external force is applied to the robot body (as shown in Fig. 5). These intervals are almost similar in each set of experiments. As one can see in (29), the disturbance happened in each task space depends on both the interaction position and the robot configuration. Hence, multiple interactions in different time and locations can show the controller

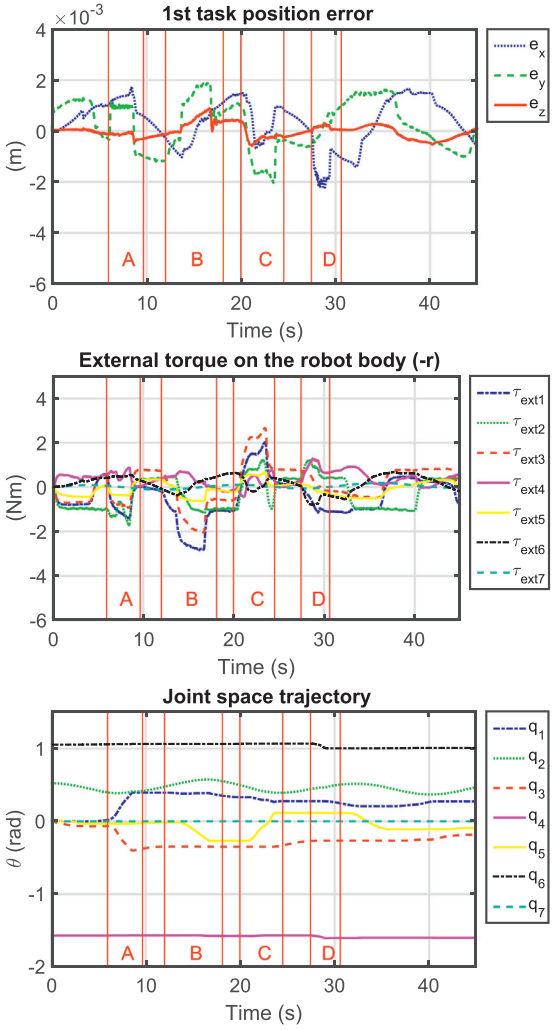

Fig. 7. Two levels hierarchy: The tracking performance of a circular path without using observer: Tracking error (Top), Estimated external torque applied to the robot body (Middle) and the joint space trajectory (Bottom).

functionality much better. Meanwhile, the robot compliance in various locations can be evaluated according to the external torques computed by the observer and also may be reported by the human.

Comparing the first task error using (32) and (33) in Figs. 4 and 6 demonstrates that the end-effector still accurately tracks the desired path in spite of the physical interaction. To evaluate the effect of observer in the control scheme, the last experiment is repeated without external interaction observer. The command acceleration in the task space is given by (12) instead of (32) and the performance is reported in the Fig. 7. Interaction intervals are shown with vertical red lines as before. In both experiments with physical interaction, the external forces are applied almost to the same points of the robot body. The time history and the magnitude of the external torques are almost the same (see Figs. 6 and 7). Moreover, comparing the task space error in these two figures, it can be seen that the error magnitude reduces during the interaction phases and goes to zero when the contact force is almost constant.

It is noteworthy that exploiting high degrees of redundancy besides low stiffness in the null-space, the magnitude of the external torque remains small during the interaction phases.

6.2. Three levels hierarchy

In the second set of experiments, each case comprises three different tasks which is defined in three hierarchy levels. In *Case I*, the end

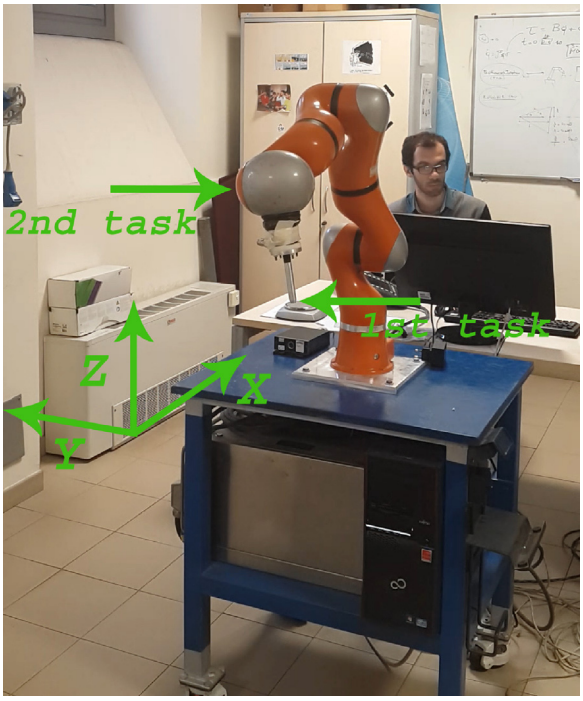

Fig. 8. Snapshot of experimental setup in case I.

effector is commanded to track a predefined trajectory and preserve the initial X and Y position of the distal end of the 5th link as the first and second priority task, respectively (see Fig. 8). Null-space compliance is considered at the third priority level. In *Case II*, the first task is to regulate the end-effector position around a desired point in the Cartesian space. At the second priority level, the distal end of the 5th link is commanded to follow a predefined path in XY plane (tasks locations are the same as the ones in Fig. 8). The null-space compliance is realized at the third priority level as before.

6.2.1. Case I

In this case, the robot end-effector is commanded to track a circular trajectory in the Cartesian space while the external force is applied to its body. The controller performance is shown in Fig. 9. Five interaction phases are specified in this figure (“A”, “B”, “C”, “D” and “E”). Both the first and the second tasks are accomplished accurately and the error converges to zero when the interaction torque is constant.

6.2.2. Case II

In this case, the manipulator end-effector is regulated around a constant position, while the distal end of the 5th link of the robot is commanded to follow a spiral trajectory in XY plane. Time history of the task errors are shown in Fig. 10 which demonstrates precise accomplishment of the regulation and the tracking task. Despite considerable force on the robot body in five intervals the task is executed accurately.

6.3. Discussion

In this work, different task combinations were implemented experimentally using the proposed approach. Three major issues are investigated in these experiments: priority allocation, accurate multiple task accomplishment and compliant behavior during physical interactions. Tracking of the desired trajectory is executed precisely in all hierarchical levels.

Residual vector computed using (30), is exploited to obtain accurate manipulation during external physical interaction with robot body. However, residual vector is not null even when no external force is applied to the robot body. This issue can be seen in other works that utilize

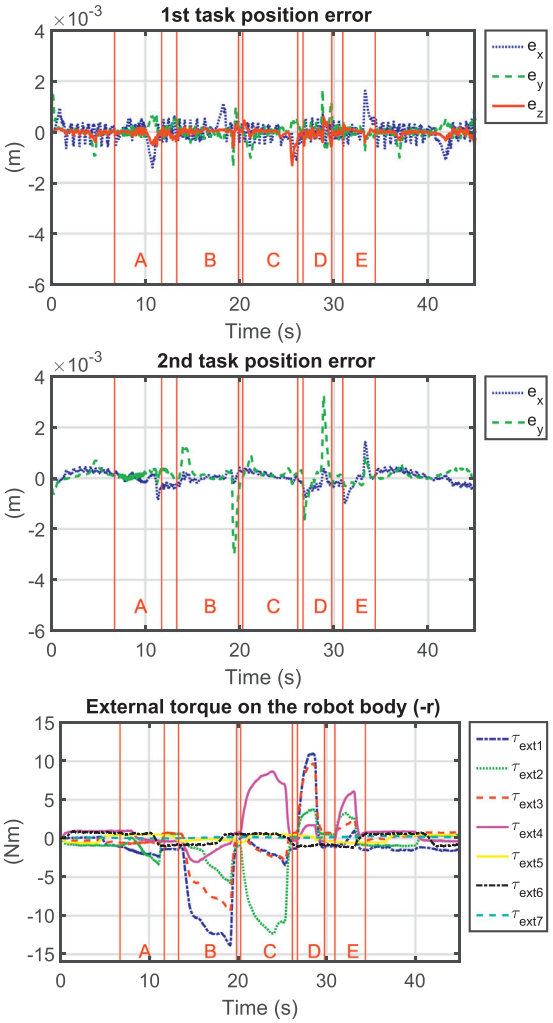

Fig. 9. Case I: Circular path tracking error (Top), 2nd task error (Middle) and Estimated external torque applied to the robot body (Bottom).

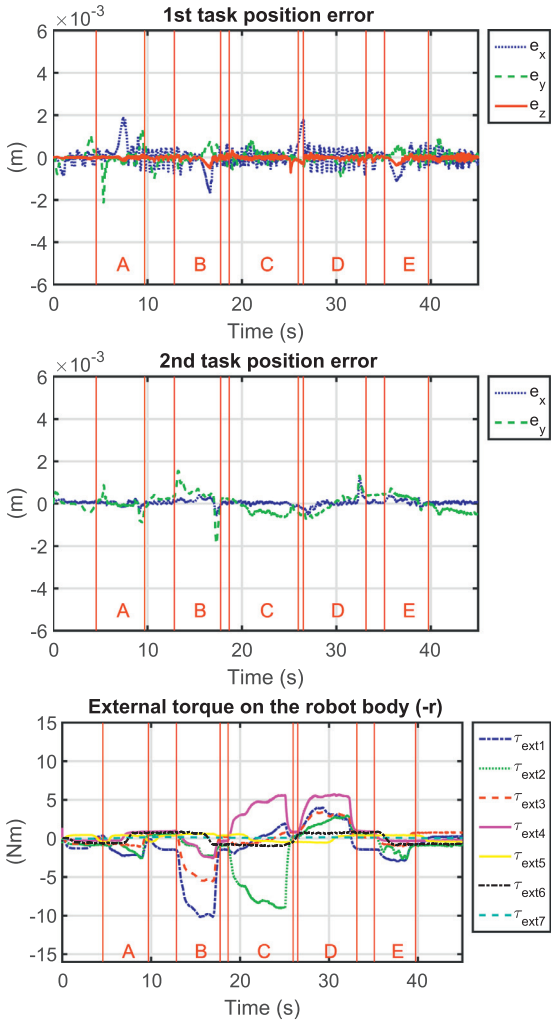

Fig. 10. Case II: 1st task error (Top), Circular path tracking error (2nd task) (Middle), Estimated external torque applied to the robot body (Bottom).

the residual vector during robot motion such as [29] and experiment 2 in [13]. Harmonic drive, actuator noise, joint friction, identification error and numerical computation are the source of this error. Thus, compensating the projected residual vector on each task space improves the manipulation accuracy during external physical interaction, and noticeably decreases the tracking error in the case of no interaction. This issue can be seen in Fig. 4 for two level hierarchy. The same results are also obtained in three level hierarchy which is omitted for the sake of brevity.

Another issue confirmed by the experiment (that are not shown here for brevity), is that, choosing low null-space stiffness decreases the external force arises in the physical interactions with the robot body. In Section 6.1, most of the physical interactions encountered with the robot are almost similar to the other set of experiments. However, high degrees of freedom in the null-space ($m_r = 4$) and low stiffness in that space decrease the magnitude of the residual vector in comparison with the one in Section 6.2. This issue has considerable effect on the tracking performance during physical interaction using (12) (without using observer). However, the results obtained by suggested controller in (32) are almost identical with different null-space stiffness.

7. Conclusion

A general integrated method for hierarchical task control including tracking in any arbitrary levels and compliance control in the lowest priority level was employed in this paper. The proposed controller ensures safe robot manipulation without any extra equipments like joint torque sensors. External torques applied on the robot body are estimated using momentum-based observer and are used in the controller. Using a Lyapanouv stability analysis, it is shown that the task space tracking errors, the external torques estimation and the null-space velocities converge to zero. Moreover, a desired compliance behavior during physical interaction is achieved in the null-space of the tasks using a new Jacobian definition. The theoretical findings and the effectiveness of the algorithm are evaluated through implementing several experiments on KUKA LWR IV. The acquired results are considerably more accurate in comparison with the previous methods in various cases. The proposed method can be employed for any other redundant robot such as humanoids.

Supplementary material

Supplementary material associated with this article can be found, in the online version, at 10.1016/j.mechatronics.2018.09.005

References

[1] Aghakhani N, Geravand M, Shahriari N, Vendittelli M, Oriolo G. Task control with remote center of motion constraint for minimally invasive robotic surgery. In: 2013 IEEE international conference on robotics and automation; 2013. p. 5807–12. doi:10.1109/ICRA.2013.6631412.

[2] Sentis L, Petersen J, Philippsen R. Implementation and stability analysis of prioritized whole-body compliant controllers on a wheeled humanoid robot in uneven terrains. Auton Robots 2013;35(4):301–19. doi:10.1007/s10514-013-9358-8.

[3] Siciliano B, Slotine J-JE. A general framework for managing multiple tasks in highly redundant robotic systems. In: Advanced robotics,1991.'robots in unstructured environments', 91 ICAR., fifth international conference on; 1991. p. 1211–16.
[4] Chiaverini S. Singularity-robust task-priority redundancy resolution for real-time kinematic control of robot manipulators. IEEE Trans Robot Autom 1997;13(3):398–410. doi:10.1109/70.585902.
[5] Flacco F, De Luca A. A reverse priority approach to multi-task control of redundant robots. In: Intelligent robots and systems (IROS 2014), 2014 IEEE/RSJ international conference on; 2014. p. 2421–7.
[6] Ott C, Dietrich A, Albu-Schäffer A. Prioritized multi-task compliance control of redundant manipulators. Automatica 2015;53:416–23.
[7] Ficuciello F, Villani L, Siciliano B. Variable impedance control of redundant manipulators for intuitive human robot physical interaction. IEEE Trans Rob 2015;31(4):850–63. doi:10.1109/TRO.2015.2430053.
[8] Liu M, Tan Y, Padois V. Generalized hierarchical control. Auton Robots 2016;40(1):17–31.
[9] Sadeghian H, Villani L, Kamranian Z, Karami A. Visual servoing with safe interaction using image moments. In: Intelligent robots and systems (IROS), 2015 IEEE/RSJ international conference on; 2015. p. 5479–85. doi:10.1109/IROS.2015.7354153.
[10] Sentis L, Park J, Khatib O. Compliant control of multicontact and center-of-mass behaviors in humanoid robots. IEEE Trans Rob 2010;26(3):483–501. doi:10.1109/TRO.2010.2043757.
[11] Dietrich A, Ott C, Park J. The hierarchical operational space formulation: stability analysis for the regulation case. IEEE Rob Autom Lett 2018;3(2):1120–7. doi:10.1109/LRA.2018.2792154.
[12] Mistry M, Nakanishi J, Schaal S. Task space control with prioritization for balance and locomotion. In: Intelligent robots and systems, 2007. IROS 2007. IEEE/RSJ international conference on; 2007. p. 331–8.
[13] Sadeghian H, Villani L, Keshmiri M, Siciliano B. Dynamic multi-priority control in redundant robotic systems. Robotica 2013;31(07):1155–67.
[14] Khatib O. A unified approach for motion and force control of robot manipulators: the operational space formulation. IEEE J Robot Autom 1987;3(1):43–53.
[15] Modified transpose jacobian control of robotic systems. Automatica 2007;43(7):1226-33. Doi:10.1016/j.automatica.2006.12.029.
[16] Khatib O, Chung SY. Suprapeds: Humanoid contact-supported locomotion for 3d unstructured environments. In: 2014 IEEE international conference on robotics and automation (ICRA); 2014. p. 2319–25. doi:10.1109/ICRA.2014.6907180.
[17] Sadeghian H, Villani L, Keshmiri M, Siciliano B. Task-space control of robot manipulators with null-space compliance. Robotics, IEEE Trans 2014;30(2):493–506.
[18] Khatib O. Inertial properties in robotic manipulation: an object-level framework. Int J Rob Res 1995;14(1):19–36. doi:10.1177/027836499501400103.
[19] Hsu P, Hauser J, Sastry S. Dynamic control of redundant manipulators. In: American control conference, 1988; 1988. p. 2135–9. doi:10.1109/ROBOT.1988.12045.
[20] O'Neil KA. Divergence of linear acceleration-based redundancy resolution schemes. IEEE Trans Robot Autom 2002;18(4):625–31. doi:10.1109/TRA.2002.801046.
[21] Nakanishi J, Cory R, Mistry M, Peters J, Schaal S. Operational space control: a theoretical and empirical comparison. Int J Rob Res 2008;27(6):737–57.
[22] Magrini E, Flacco F, De Luca A. Control of generalized contact motion and force in physical human-robot interaction. In: 2015 IEEE international conference on robotics and automation (ICRA); 2015. p. 2298–304. doi:10.1109/ICRA.2015.7139504.
[23] Gaz C, Magrini E, De Luca A. A model-based residual approach for human-robot collaboration during manual polishing operations. Mechatronics 2018. doi:10.1016/j.mechatronics.2018.02.014.
[24] De Luca A, Flacco F. Integrated control for phri: Collision avoidance, detection, reaction and collaboration. In: 2012 4th IEEE RAS EMBS international conference on biomedical robotics and biomechatronics (BioRob); 2012. p. 288–95. doi:10.1109/BioRob.2012.6290917.
[25] Fritzsche M, Elkmann N, Schulenburg E. Tactile sensing: A key technology for safe physical human robot interaction. In: 2011 6th ACM/IEEE international conference on human-robot interaction (HRI); 2011. p. 139–40. doi:10.1145/1957656.1957700.
[26] Cirillo A, Cirillo P, Pirozzi S. A modular and low-cost artificial skin for robotic applications. In: 2012 4th IEEE RAS EMBS international conference on biomedical robotics and biomechatronics (BioRob); 2012. p. 961–6. doi:10.1109/BioRob.2012.6290685.
[27] Mazzocchi T, Diodato A, Ciuti G, Micheli DMD, Menciassi A. Smart sensorized polymeric skin for safe robot collision and environmental interaction. In: 2015 IEEE/RSJ international conference on intelligent robots and systems (IROS); 2015. p. 837–43. doi:10.1109/IROS.2015.7353469.
[28] De Luca A, Mattone R. Sensorless robot collision detection and hybrid force/motion control. In: Proceedings of the 2005 IEEE international conference on robotics and automation; 2005. p. 999–1004. doi:10.1109/ROBOT.2005.1570247.
[29] De Luca A, Albu-Schaffer A, Haddadin S, Hirzinger G. Collision detection and safe reaction with the dlr-iii lightweight manipulator arm. In: 2006 IEEE/RSJ International conference on intelligent robots and systems; 2006. p. 1623–30. doi:10.1109/IROS.2006.282053.
[30] De Luca A, Ferrajoli L. Exploiting robot redundancy in collision detection and reaction. In: 2008 IEEE/RSJ international conference on intelligent robots and systems; 2008. p. 3299–305. doi:10.1109/IROS.2008.4651204.
[31] Villani V, Pini F, Leali F, Secchi C. Survey on humanrobot collaboration in industrial settings: safety, intuitive interfaces and applications. Mechatronics 2018. doi:10.1016/j.mechatronics.2018.02.009.
[32] Zergeroglu E, Dawson DD, Walker IW, Setlur P. Nonlinear tracking control of kinematically redundant robot manipulators. IEEE/ASME Trans Mechatron 2004;9(1):129–32. doi:10.1109/TMECH.2004.823890.
[33] Mistry M, Buchli J, Schaal S. Inverse dynamics control of floating base systems using orthogonal decomposition. In: 2010 IEEE international conference on robotics and automation; 2010. p. 3406–12. doi:10.1109/ROBOT.2010.5509646.
[34] Yanai H, Takeuchi K, Takane Y. Projection matrices, generalized inverse matrices, and singular value decomposition. Statistics for Social and Behavioral Sciences. Dordrecht: Springer; 2011.
[35] Ott C. Cartesian impedance control of redundant and flexible-Joint robots. Springer Tracts in Advanced Robotics. Springer Berlin Heidelberg; 2008.
[36] Nemec B, lajpah L, Omrcen D. Comparison of null-space and minimal null-space control algorithms. Robotica 2007;25(5):511–20. doi:10.1017/S0263574707003402.
[37] De Luca A, Oriolo G. Nonholonomic behavior in redundant robots under kinematic control. IEEE Trans Robot Autom 1997;13(5):776–82. doi:10.1109/70.631239.
[38] Ott C, Kugi A, Nakamura Y. Resolving the problem of non-integrability of nullspace velocities for compliance control of redundant manipulators by using semi-definite lyapunov functions. In: Robotics and automation, 2008. ICRA 2008. IEEE international conference on; 2008. p. 1999–2004.
[39] Wang Z, Tan Y, Wang G, Neic D. On stability properties of nonlinear time-varying systems by semi-definite time-varying lyapunov candidates. IFAC Proc Vol 2008;41(2):1123–8. doi:10.3182/20080706-5-KR-1001.00194.
[40] Khalil HK, Grizzle J. Nonlinear systems, vol. 3. Prentice hall New Jersey; 1996.
[41] P.-H. Chang, A closed-form solution for inverse kinematics of robot manipulators with redundancy (1987).
[42] Gaz C, Flacco F, De Luca A. Identifying the dynamic model used by the kuka lwr: A reverse engineering approach. In: 2014 IEEE international conference on robotics and automation (ICRA); 2014. p. 1386–92. doi:10.1109/ICRA.2014.6907033.
[43] Gaz C, Flacco F, De Luca A. Extracting feasible robot parameters from dynamic coefficients using nonlinear optimization methods. In: 2016 IEEE international conference on robotics and automation (ICRA); 2016. p. 2075–81. doi:10.1109/ICRA.2016.7487356.

Abbas Karami was born in Shiraz, Iran, in 1989. He received the B.Sc. and M.Sc. degrees in mechanical engineering from the Shiraz University, Shiraz, Iran, in 2011 and 2013, respectively. He received the Ph.D. degree in mechanical engineering (robotics and control) from the Isfahan University of Technology in 2018. From March 2016 to March 2017, he was a Visiting Scholar with the DIAG Robotics Lab, Sapienza University of Rome, Italy. His research interest include human-robot interaction, surgical robotics, nonlinear control and modeling of biomechanical systems.

Hamid Sadeghian was born in Isfahan, Iran, in 1982. He received the B.Sc. and M.Sc. degrees in Mechanical Engineering from the Isfahan University of Technology and the Sharif University of Technology, Tehran, Iran, in 2005 and 2008, respectively. He received the Ph.D. degree in mechanical engineering (robotics and control) from the Isfahan University of Technology in 2013. From November 2010 to January 2013, he was visiting scholar with the PRISMA Laboratory, Department of Electrical Engineering and Information Technology, University of Naples, Naples, Italy. Starting from 2013 he is Assistant professor of Control and Robotics with the Engineering Dept. in University of Isfahan. In Sep. 2015 Dr. Sadeghian visited Institute of Cognitive System at Technical University of Munich and German Aerospace Center (DLR), Munich, Germany, for one year postdoctoral research on locomotion of bipeds. His research interests include physical human-robot interaction, redundant manipulation, and nonlinear control of mechanical systems.

Mehdi Keshmiri was born in Natanz, Iran, in 1961. He received the B.Sc. and M.Sc. degree in mechanical engineering from the Sharif University of Technology, Tehran, Iran, in 1986 and 1989, respectively. He received the Ph.D. degree in mechanical engineering (space dynamics) from McGill University, Montreal, QC, Canada, in 1995. He then joined the Isfahan University of Technology, Isfahan, Iran, where he is currently a Professor with the Department of Mechanical Engineering. His research interests include system dynamics, control systems and dynamics, and control of robotic systems. He has presented and published more than 100 papers in international conferences and journals and supervised more than 60 Ph.D. and Masters students.

Giuseppe Oriolo (S89M92SM02F16) received the Ph.D. degree in control engineering from Sapienza University of Rome, Italy, in 1992. He is currently with the Department of Computer, Control and Management Engineering (DIAG), Sapienza University of Rome, as a Full Professor of automatic control and robotics and the Coordinator of the DIAG Robotics Lab. His research interests are in the general area of planning and control of robotic systems. Prof. Oriolo has been an Associate Editor of the IEEE TRANSACTIONS ON ROBOTICS AND AUTOMATION from 2001 to 2005, and an Editor of the IEEE TRANSACTIONS ON ROBOTICS from 2009 to 2013.