2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 25-29, 2020, Las Vegas, NV, USA (Virtual)

Experiments on whole-body control of a dual-arm mobile robot with the Set-Based Task-Priority Inverse Kinematics algorithm

Paolo Di Lillo*, Francesco Pierri†, Fabrizio Caccavale†, Gianluca Antonelli*

Abstract— **In this paper an experimental study of set-based task-priority kinematic control for a dual-arm mobile robot is developed. The control strategy for the coordination of the two manipulators and the mobile base relies on the definition of a set of elementary tasks to be properly handled depending on their functional role. In particular, the tasks have been grouped into three categories: safety, operational and optimization tasks. The effectiveness of the resulting task hierarchy has been validated through experiments on a Kinova Movo robot, in a domestic use case scenario.**

I. Introduction

Many robotic tasks, e.g. parts mating and/or transportation and manipulation of heavy or large object, require the adoption of multi-arm systems. The addition of a mobile base to a dual-arm system allows to accomplish tasks requiring a larger operational space and increases the available degrees of freedom (DOFs), at the expense of an increased system complexity. Thus, the coordinated motion control of the arms and the mobile base for the execution of complex tasks in cluttered environments has become subject of active research in the domain of mobile manipulation.

Pioneering works on mobile manipulators did not deal with the mobile base and the arm coordination problem, since the platform and manipulator were considered as separate entities, to be sequentially moved by exploiting the base for the approach and the arm for the manipulation. Other early approaches only partially exploited the system DOFs. E.g., in [1] the dynamics of a manipulator mounted on a nonholonomic mobile base is used to devise a motion planning approach for a planar two-link mobile manipulator. In [2] the well-known augmented object and virtual linkage model are adopted to devise a decentralized control scheme for cooperative mobile manipulators, where redundancy is handled at the dynamic level. More recent works considered a kinematic approach in which the base and arm coordination is approached via the Jacobian matrix of the overall system [3],[4]. In [5] the mobile base is user-controlled and its motion is seen as a disturbance to be compensated by the arm.

In the above mentioned works, however, kinematic redundancy does not play a major role, while proper handling of redundancy should be exploited to achieve safe and robust achievement of complex tasks in cluttered environment, possibly co-habitated by humans and robots. In order to exploit the redundancy of robotic systems, multi-priority control, based on the Null Space based Behavioral (NSB) approach [6], has been widely adopted both for manipulators and mobile robots [7]. More recently, it was extended and experimentally validated to aerial manipulators in [8] and [9] to achieve coordination between an aerial vehicle and a 6-DOF arm. However, in these papers only tasks defined in terms of equality constraints have been taken into consideration, while in complex robotic system many tasks, e.g., joint limits avoidance, can be more conveniently assigned by resorting to inequality constraints, i.e., control variables that need to be kept in a range of values instead of an exact one (hereafter *set-based* tasks).

Different approaches in the last years tackled the problem of handling inequality constraints, often by transforming such inequality constraints into equivalent equality constraints or potential functions. In humanoid robot control [10] both equality and inequality constraints are taken into account by resorting to a hierarchical multiple least-square quadratic optimization problem. Similarly, in [11], the transformation of the inequality constraints into equality ones in a task-priority architecture is performed by means of slack variables, which are, then, minimized together to the task errors. The potential field approach is adopted in [12] for an underwater vehicle-manipulator system, where smooth potential fields represent the set-based objectives and activation functions. Effectiveness of such an approach was experimentally validated; however, strict priority among the tasks is not ensured, due to the presence of smoothing functions introduced to avoid discontinuities. In order to overcome these drawbacks, in [13], a method that directly embeds set-based tasks into a singularity-robust multiple task-priority inverse kinematics framework, based on the NSB paradigm, was introduced and experimentally validated for a fixed-base manipulator with 7 DOFs. More recently such an approach has been extended to a dual-arm aerial manipulator in [14].

In this paper, the approach developed in [13] is adopted to achieve coordinated motion of a kinematically redundant dual-arm mobile manipulator with holonomic mobile base. The control strategy is based on the definition of a set of elementary tasks to be properly handled depending on their functional role. In particular, the tasks are grouped into three categories: safety, operational and optimization tasks. Safety tasks are aimed at avoid damage of the system and/or of the surrounding environment, and thus an higher priority with respect to the other tasks is assigned; safety tasks are set-based tasks activated only when the corresponding task value

* Department of Electrical and Information Engineering of the University of Cassino and Southern Lazio, Via G. Di Biasio 43, 03043 Cassino (FR), Italy

† School of Engineering, University of Basilicata, Via dell'Ateneo Lucano 10, 85100 Potenza (PZ), Italy

978-1-7281-6211-9/20/$31.00 ©2020 IEEE

exceeds an activation threshold. Operational tasks, usually expressed as equality constraints, are those strictly required to achieve the assigned mission. Finally, optimization tasks are aimed at achieving configurations of the mobile base optimal in some sense (e.g., to achieve more dexterous poses of the object held by the manipulators).

The adopted approach is a kinematic control scheme, i.e., the set-based task-priority inverse kinematics algorithm is used to compute the motion references for the low-level motion controller of the arm and the mobile base.

The effectiveness of the whole control architecture is validated through experiments on a Kinova Movo robot, in a domestic use case scenario, where the two arms hold an object to be transported to a desired final goal.

II. Set-based Task-Priority Inverse Kinematics

A *task* for a robotic system is defined as an m-dimensional function of the n-dimensional system's state, $\boldsymbol{\sigma} = \boldsymbol{f}(\boldsymbol{s})$; then, the mapping between the task velocity $\dot{\boldsymbol{\sigma}}$ and the system velocity $\boldsymbol{\zeta}$ can be expressed as:

$$\dot{\boldsymbol{\sigma}} = \frac{\partial \boldsymbol{f}(\boldsymbol{s})}{\partial \boldsymbol{s}} \boldsymbol{\zeta} = \boldsymbol{J} \boldsymbol{\zeta} \tag{1}$$

where $\boldsymbol{J} \in \mathbb{R}^{m \times n}$ is the task Jacobian matrix. Given a desired task trajectory, $\boldsymbol{\sigma}_d(t)$, the corresponding reference system's velocity can be computed by resorting to a Closed-Loop Inverse Kinematics (CLIK) algorithm [15]:

$$\boldsymbol{\zeta} = \boldsymbol{J}^{\dagger} (\dot{\boldsymbol{\sigma}}_d + \boldsymbol{K} \tilde{\boldsymbol{\sigma}}) \,, \tag{2}$$

where $\boldsymbol{K} \in \mathbb{R}^{m \times m}$ is a positive-definite matrix of gains, $\tilde{\boldsymbol{\sigma}} = \boldsymbol{\sigma}_d - \boldsymbol{\sigma}$ is the task error and $\boldsymbol{J}^{\dagger}$ is the Moore-Penrose pseudoinverse of the task Jacobian.

When the system is redundant with respect to the task ($n > m$), multiple tasks can be commanded simultaneously defining a priority among them and computing the reference system velocity that fulfills the task hierarchy *at best*, removing from the solution the velocity components of the lower-priority tasks that would affect the higher priority ones.

In order to combine a hierarchy of h tasks, it is necessary to assign h levels of priority, i, from the primary one ($i = 1$) to the lowest priority one ($i = h$). Then, the reference system velocity can be computed recursively as in [16]:

$$\boldsymbol{\zeta}_h = \sum_{i=1}^{h} (\boldsymbol{J}_i \boldsymbol{N}_{i-1}^A)^{\dagger} (\dot{\boldsymbol{\sigma}}_{i,d} + \boldsymbol{K}_i \tilde{\boldsymbol{\sigma}}_i - \boldsymbol{J}_i \boldsymbol{\zeta}_{i-1}) \,, \tag{3}$$

where $\boldsymbol{N}_i^A$ projects a vector in the null space of the augmented Jacobian matrix, obtained by stacking all the task Jacobian matrices from task 1 to i, i.e.,

$$\boldsymbol{J}_i^A = \begin{bmatrix} \boldsymbol{J}_1^T & \boldsymbol{J}_2^T & \dots & \boldsymbol{J}_i^T \end{bmatrix}^T \,, \tag{4}$$

and is defined as

$$\boldsymbol{N}_i^A = \boldsymbol{I} - (\boldsymbol{J}_i^A)^{\dagger} \boldsymbol{J}_i^A \,, \tag{5}$$

with $\boldsymbol{N}_0^A = \boldsymbol{I}$, $\boldsymbol{\zeta}_0 = \boldsymbol{0}_n$ (the n-dimensional null vector), and $\boldsymbol{\zeta}_i$ is the solution at the i-th iteration.

The aforementioned task-priority framework has been developed for handling *equality-based* tasks, in which the control objective is to bring the task value to a specific one. Recently, this framework has been extended to handle also *set-based tasks*, in which the task value has to be kept in a certain set of values, namely above a lower threshold and below an upper one.The key idea is to consider a set-based task as an equality-based one that can be inserted or removed from the hierarchy depending on the operational conditions.

Details about the tasks activation/deactivation algorithm can be found in [17].

III. System description

Consider a a mobile robot (e.g., the Kinova Movo robot described in Section V) composed by an holonomic mobile base (e.g., equipped with four Swedish wheels), two kinematically redundant manipulators (without loss of generality, 7-DOFs manipulators are assumed) and additional prismatic DOF (torso joint) allowing to change the vertical position of the two-arm system.

The state of the system is described by the vector:

$$\boldsymbol{s} = \begin{bmatrix} \boldsymbol{p}_b^T & \theta & \boldsymbol{q}_{\mathrm{L}}^T & \boldsymbol{q}_{\mathrm{R}}^T \end{bmatrix}^T \in \mathbb{R}^{18} \,, \tag{6}$$

where $\boldsymbol{p}_b = \begin{bmatrix} x & y & z \end{bmatrix}^T$ is the vector containing the x, y position of the mobile base in inertial frame and the z position of the torso joint, θ is the platform orientation, $\boldsymbol{q}_l = \begin{bmatrix} q_{l,1} & \dots & q_{l,7} \end{bmatrix}^T$ and $\boldsymbol{q}_r = \begin{bmatrix} q_{r,1} & \dots & q_{r,7} \end{bmatrix}^T$ are the joint positions of the left and right manipulator, respectively. The system velocity vector is

$$\boldsymbol{\zeta} = \begin{bmatrix} \boldsymbol{\nu}_b^T & \dot{\theta} & \dot{\boldsymbol{q}}_{\mathrm{L}}^T & \dot{\boldsymbol{q}}_{\mathrm{R}}^T \end{bmatrix}^T \in \mathbb{R}^{18} \,, \tag{7}$$

where $\boldsymbol{\nu}_b = \begin{bmatrix} \nu_x & \nu_y & \dot{z} \end{bmatrix}^T$ is the linear velocity of the platform expressed in the mobile base frame and the velocity of the torso joint, $\dot{\theta}$ is the angular velocity of the platform, $\dot{\boldsymbol{q}}_l = \begin{bmatrix} \dot{q}_{l,1} & \dots & \dot{q}_{l,7} \end{bmatrix}^T$ and $\dot{\boldsymbol{q}}_r = \begin{bmatrix} \dot{q}_{r,1} & \dots & \dot{q}_{r,7} \end{bmatrix}^T$ are the left and right arm joint velocities, respectively. The m-dimensional Jacobian matrix of a generic task $\boldsymbol{\sigma}_x$ can be expressed as

$$\boldsymbol{J}_x = \Bigg[\overbrace{\underbrace{\frac{\partial \boldsymbol{f}_x(\boldsymbol{s})}{\partial \boldsymbol{p}_b}}_{m \times 4} \quad \frac{\partial \boldsymbol{f}_x(\boldsymbol{s})}{\partial \theta}}^{\text{base}} \quad \overbrace{\underbrace{\frac{\partial \boldsymbol{f}_x(\boldsymbol{s})}{\partial \boldsymbol{q}_{\mathrm{L}}}}_{m \times 7}}^{\text{left arm}} \quad \overbrace{\underbrace{\frac{\partial \boldsymbol{f}_x(\boldsymbol{s})}{\partial \boldsymbol{q}_{\mathrm{R}}}}_{m \times 7}}^{\text{right arm}} \Bigg] \in \mathbb{R}^{m \times 18} \,, \tag{8}$$

where the velocity contribution of the mobile base, the left and the right arms on the task derivative with their dimensions are highlighted.

Let us define six frames of interest for the tasks definition conducted in the following Section (see Fig 1): $\boldsymbol{\mathcal{F}}_I$ (inertial frame), $\boldsymbol{\mathcal{F}}_V$ (mobile base frame), $\boldsymbol{\mathcal{F}}_{L,B}$ (left arm base frame), $\boldsymbol{\mathcal{F}}_{R,B}$ (right arm base frame), $\boldsymbol{\mathcal{F}}_{L,E}$ (left end-effector frame), $\boldsymbol{\mathcal{F}}_{R,E}$ (right end-effector frame). The generic transformation matrix that expresses the rototranslation between frame $\boldsymbol{\mathcal{F}}_X$ and $\boldsymbol{\mathcal{F}}_Y$ is $\boldsymbol{T}_{\mathrm{X}}^{\mathrm{Y}} \in \mathbb{R}^{4 \times 4}$ and it is composed by the rotation matrix $\boldsymbol{R}_{\mathrm{X}}^{\mathrm{Y}} \in \mathbb{R}^{3 \times 3}$, the null vector $\boldsymbol{0}_3$ and the position vector $\boldsymbol{p}_{\mathrm{X}}^{\mathrm{Y}} \in \mathbb{R}^3$. For the considered system, the

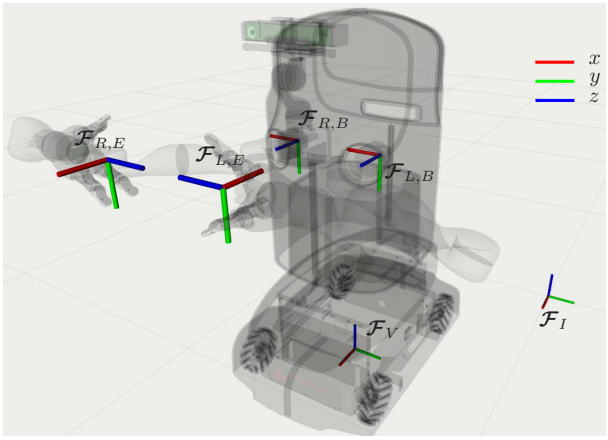

Fig. 1. Frames of interest for the dual-arm mobile robot

matrix $\boldsymbol{T}^{\mathrm{I}}_{\mathrm{V}}$ can be computed via measurements provided by the localization system, that makes use of odometry and a compass for the position and orientation estimation. In the matrices $\boldsymbol{T}^{\mathrm{V}}_{\mathrm{L,B}}$, $\boldsymbol{T}^{\mathrm{V}}_{\mathrm{R,B}}$, the rotation matrices $\boldsymbol{R}^{\mathrm{V}}_{\mathrm{L,B}}$, $\boldsymbol{R}^{\mathrm{V}}_{\mathrm{R,B}}$ and the first two elements of the position vectors, namely $\boldsymbol{p}^{\mathrm{V}}_{\mathrm{L,B}}$ and $\boldsymbol{p}^{\mathrm{V}}_{\mathrm{R,B}}$, are constant, while the third components depend on the position of the prismatic joint at the torso. Finally, the matrices $\boldsymbol{T}^{\mathrm{L,B}}_{\mathrm{L,E}}$, $\boldsymbol{T}^{\mathrm{R,B}}_{\mathrm{R,E}}$ can be obtained by expressing the kinematic chain of the manipulators as Denavit-Hartenberg parameters and then composing the transformation matrices of all the frames from the base to the arm end-effector.

IV. Implemented Tasks

In this Section, the elementary tasks that have been defined to effectively perform a common operation for a service robot in a domestic environment are described.

The assigned mission is to bring a tray, grasped by both hands, in a given location. The operation is split in several sub-tasks, arranged in a hierarchy that can be divided in three categories with decreasing priority order [18], [19]: safety tasks, operational tasks and optimization tasks. In the following, all the elementary tasks are described in terms of dimension m and mathematical formulation σ_x.

A. Safety tasks

In this category, the tasks needed to avoid damage of the system (or the objects/humans in the surrounding environment) are comprised. They are all defined as set-based tasks, activated only when the corresponding value exceeds pre-defined activation thresholds, with higher priority with respect to the tasks in the other two categories.

Joint limits ($m = 1$)**:** upper and lower limits have to placed on the joints of the manipulators subject to mechanical limits, e.g., for the Kinova Movo, the second, fourth and sixth joints of the two manipulators. The task function is the value of the i-th joint variable of the left or right manipulator

$$\sigma_{\mathrm{L,joint},i} = q_{\mathrm{L},i} \ , \qquad \sigma_{\mathrm{R,joint},i} = q_{\mathrm{R},i} \ . \tag{9}$$

Virtual walls ($m = 1$)**:** six virtual walls, arranged to compose a cube, have been placed around each of the two arms base frame, in order to limit the motion of the end-effector. The limits are chosen in order to avoid the singular configurations that would occur at the boundary of the arms workspace. The task function is defined as

$$\sigma_{\mathrm{wall}} = \hat{\boldsymbol{n}}^T (\boldsymbol{p}_e - \boldsymbol{p}_1) \ , \tag{10}$$

where $\hat{\boldsymbol{n}}$ is the outer normal unit vector from the plane computed as:

$$\hat{\boldsymbol{n}} = \frac{(\boldsymbol{p}_2 - \boldsymbol{p}_1) \times (\boldsymbol{p}_3 - \boldsymbol{p}_1)}{||(\boldsymbol{p}_2 - \boldsymbol{p}_1) \times (\boldsymbol{p}_3 - \boldsymbol{p}_1)||} \ , \tag{11}$$

and $\boldsymbol{p}_1$, $\boldsymbol{p}_2$ and $\boldsymbol{p}_3$ are three points belonging to the plane.

B. Operational tasks

In this category, all the tasks strictly needed for the accomplishment of the assigned mission are included. In the case of cooperative transportation of a rigid (or nearly rigid) object held by two robotic manipulators, two elementary tasks are required: the first one is the pose of the end effector of one of the two arms (e.g., the left arm) in the inertial frame, expressed as a function of the mobile base pose and the left arm joint variables, while the second one constraints the relative pose of the two end-effectors expressed in the arm base frame.

Global pose ($m = 6$)**:** the task value is the vector stacking the position and the unit quaternion [20] of the left end-effector with respect to the inertial frame

$$\boldsymbol{\sigma}_{\mathrm{global}} = \begin{bmatrix} \boldsymbol{p}^{\mathrm{I}}_{\mathrm{L,E}} & \mathcal{Q}^{\mathrm{I}}_{\mathrm{L,E}} \end{bmatrix}^T \ , \tag{12}$$

where $\boldsymbol{p}^{\mathrm{I}}_{\mathrm{L,E}}$ is the position of the left end-effector expressed in inertial frame $\boldsymbol{\mathcal{F}}_I$ and $\mathcal{Q}^{\mathrm{I}}_{\mathrm{L,E}}$ is the unit quaternion expressing its orientation, that can be extracted by the transformation matrix $\boldsymbol{T}^I_{L,E} = \boldsymbol{T}^I_V \, \boldsymbol{T}^V_{L,B} \, \boldsymbol{T}^B_{L,E}$.

Relative pose ($m = 6$): the task function is the vector stacking the position and orientation of the left end-effector with respect to the right one [21]:

$$\boldsymbol{\sigma}_{\mathrm{rel}} = \begin{bmatrix} \boldsymbol{p}^{\mathrm{L}}_{\mathrm{R,E}} - \boldsymbol{p}^{\mathrm{L}}_{\mathrm{L,E}} & \mathcal{Q}^{\mathrm{R}}_{\mathrm{L}} \end{bmatrix} \ , \tag{13}$$

where $\boldsymbol{p}^{\mathrm{L}}_{\mathrm{L,E}}$ and $\boldsymbol{p}^{\mathrm{L}}_{\mathrm{R,E}}$ are the position of the left and right end-effectors, respectively, expressed in a common reference frame (the left arm base frame), and $\mathcal{Q}^{\mathrm{R}}_{\mathrm{L}}$ is the unit quaternion expressing the orientation of the left end-effector with respect to the right one.

C. Optimization tasks

When a target global position and orientation is effectively reached, the redundancy could force the system to assume an *unnatural* steady-state configuration, e.g. with the end effectors at the boundary of the virtual walls defined as high-priority tasks. The goal of the optimization tasks is to adjust the position and the orientation of the mobile base in order to keep the absolute position of the dual-arm system [22] (e.g., the mid-point between the two end-effectors) aligned as much as possible with the x axis of the mobile base frame during the motion, resulting in a more *natural* steady-state configuration.

Mobile base configuration optimization ($m = 3$): let us define an auxiliary frame $\boldsymbol{\mathcal{F}}_A$ centered in the mid-point

between the two end-effectors, as shown in Fig. 2. The mobile base configuration optimization task value is:

$$\boldsymbol{\sigma}_{\text{opt}} = \begin{bmatrix} \boldsymbol{p}_{\text{V}}^{\text{A}} & \theta^{\star} \end{bmatrix} , \tag{14}$$

where $\boldsymbol{p}_{\text{V}}^{\text{A}}$ is the position of the mobile base frame with respect to the auxiliary frame, and $\theta^{\star}$ is the rotation angle around the z-axis of the mobile base frame needed to align it with the auxiliary frame.

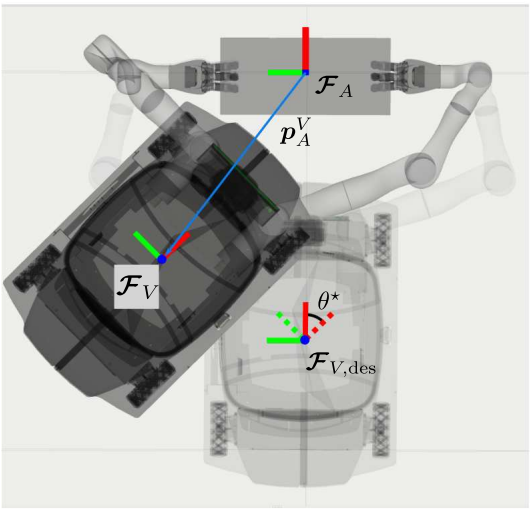

Fig. 2. Mobile base configuration optimization task frames and variables.

V. Experiments

This section is focused on the presentation of the experimental case study proving the effectiveness of the approach for the accomplishment of a typical domestic use case scenario, in which the robot has to bring a tray in a desired location holding it with two hands.

The chosen robotic platform is the Kinova Movo, a mobile robot manufactured by Kinova[1] equipped with two 7 DOF Kinova Ultra lightweight robotic Jaco[2] arms. The mobile base is equipped with four Swedish wheels thus making the robot holonomic, and linked to it there is a prismatic joint that allows to change the height of the torso.

The platform frame of the robot is initially centered in $\boldsymbol{p}_{b,\text{ini}} = \begin{bmatrix} 0 & 0 & 0.2 \end{bmatrix}^T$, while holding a 40 cm-long tray with both hands. The left end effector initial position is $\boldsymbol{\sigma}_{\text{global, pos}} = \begin{bmatrix} 0.66 & 0.15 & 0.85 \end{bmatrix}^T$ m with an orientation $\boldsymbol{\sigma}_{\text{global, ori}} = \begin{bmatrix} 0 & -0.7071 & 0.7071 & 0 \end{bmatrix}^T$. The desired global position and orientation is given as a constant reference to the left end-effector and it is chosen to be behind the robot, with an orientation of $-90°$ around the y-axis with respect to the initial orientation. The reference for the relative pose task is set equal to the initial one, allowing it to hold horizontally the tray. Regarding the platform configuration optimization task, the desired mobile frame position is set as $\boldsymbol{p}_{\text{V, des}}^{\text{A}} = \begin{bmatrix} -0.7 & 0 \end{bmatrix}^T$ m, while the desired orientation is set as $\theta_{\text{des}}^{\star} = 0$, making the mobile base frame centered and aligned with the auxiliary one, as shown in Fig. 2.

In the following, the results obtained by applying two different task hierarchies are discussed, aiming at highlighting the contribution of each task category in the overall motion of the system. A video of the experiments can be found at `https://youtu.be/6GcLKFxdN34`.

A. Safety + operational tasks

In this subsection, the proposed control scheme is implemented by considering safety (joint limits and virtual walls) and operational tasks (global and relative poses) in the hierarchy, composed in the following hierarchical order:

1) Limits on three joints of both arms (6 tasks, $m = 1$ each)
2) Virtual walls around the base frames of both arms (12 tasks, $m = 1$ each)
3) Relative pose ($m = 6$)
4) Global pose ($m = 6$)

The experimental results in Fig. 3 show that both the global and relative position and orientation errors reach a null steady-state value, meaning that the left end-effector reaches the desired location while keeping the relative pose with respect to the right one at a constant value. Regarding the safety tasks plots, the horizontal red lines highlight the imposed safety thresholds and it is clear that they are not violated. In particular, it can be noticed that several safety tasks become active during the motion and that they get stuck at the desired safety thresholds: the second and fourth joints of the right arm, the wall on the z direction of the right arm base frame and the wall on the x direction of the left arm base frame.

It is worth noticing that the steady-state configuration, shown in Fig 4, has three active set-based tasks and it is quite *unnatural*, with the tray on the right side of the robot body and not aligned with it.

B. Safety + operational + optimization tasks

Results obtained in an experiment in which all the task categories are included (safety, operational and optimization tasks) are discussed in this subsection. Thus, the resulting hierarchy is:

1) Limits on three joints of both arms (6 tasks, $m = 1$ each)
2) Virtual walls around the base frames of both arms (12 tasks, $m = 1$ each)
3) Relative pose ($m = 6$)
4) Global pose ($m = 6$)
5) Base configuration optimization ($m = 3$).

Figure 5 shows the time histories relative to the operational tasks errors, the safety task values and the optimization task error. It can be noticed that the optimization task error reaches a null-steady-state value, while all the safety tasks are kept inside the assigned boundaries and the operational tasks are effectively accomplished. Noticeably, in this case, due to the effect of the optimization task that moves the

[1] www.kinovarobotics.com/en/products/mobile-manipulators

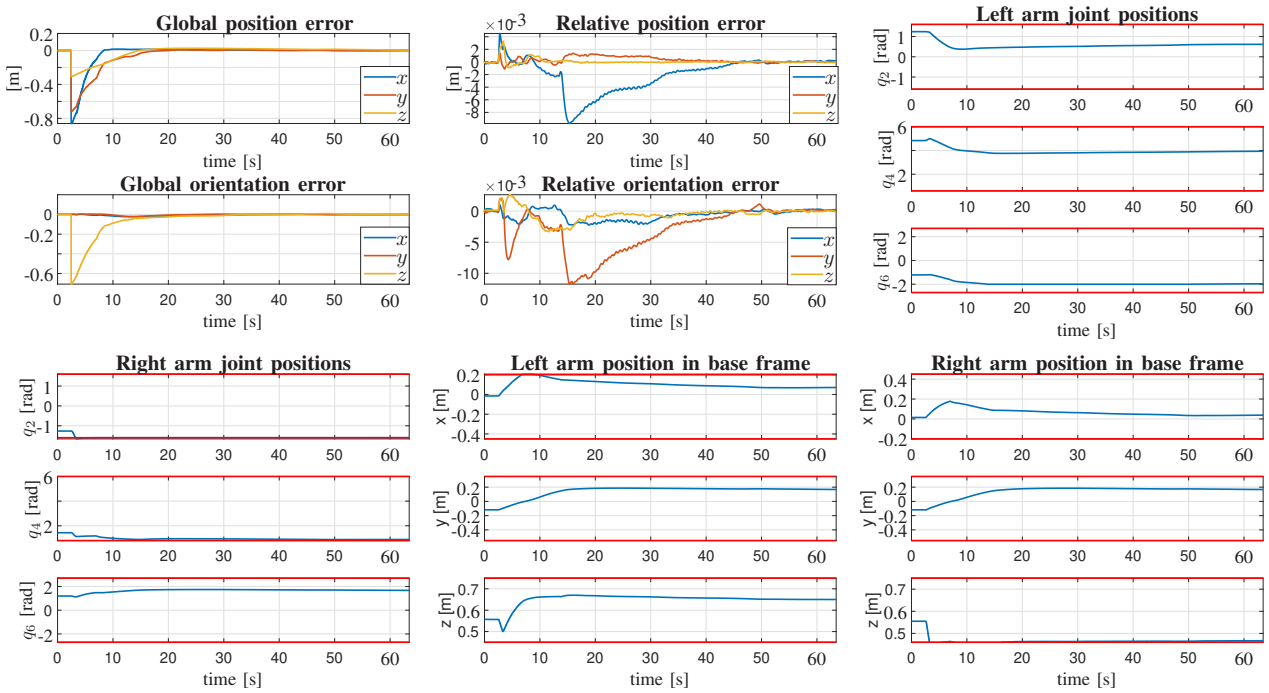

Fig. 3. Experiment, safety + operational tasks. From top-left to bottom-right: global pose error; relative pose error; left and right arm joint positions together with the imposed safety thresholds; left and right arm pose expressed in their base frames together with the safety thresholds imposed by the virtual wall tasks. All the safety tasks are respected.

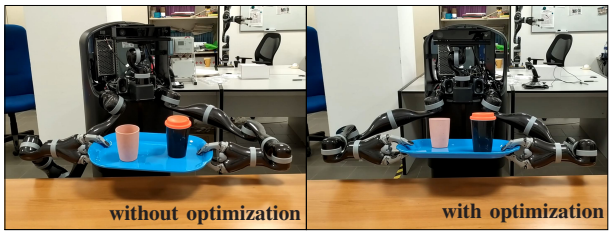

Fig. 4. Left: steady-state configuration for the hierarchy containing safety and operational tasks. Right: steady-state configuration for the hierarchy containing also the optimization task. The tray is centered with respect to the robot body, resulting in a more *natural* posture.

mobile base in order to center and align the tray with the mobile base frame, the steady-state configuration (shown in Fig. 4.b) has all the safety tasks deactivated, resulting in a less constrained system and in a more *natural* posture.

VI. Conclusions and Future Work

In this paper, a set-based task-priority kinematic control scheme for a dual-arm mobile manipulator is devised and experimentally tested. A set of elementary set-based and equality-based tasks has been designed, properly categorized, depending on their functional role (safety, operational and optimization tasks), and prioritized, in order to ensure the integrity of the system and the effectiveness of the mission to be accomplished.

Future work will be focused on the definition of more complex dual-arm tasks, in which proper the definition of the relative motion between the two end-effector plays a key role for the accomplishment of the operation.

Acknowledgement

This paper has been supported by the MIUR program "Dipartimenti di Eccellenza 2018-2022", and partly funded by MIUR PON R&I 2014-2020 Program (project ICOSAF, ARS01_00861).

References

[1] Y. Yamamoto and X. Yun, "Coordinating locomotion and manipulation of a mobile manipulator," *IEEE Transactions on Automatic Control*, vol. 39, no. 6, pp. 1326–1332, 1994.

[2] O. Khatib, K. Yokoi, K. Chang, D. Ruspini, R. Holmberg, and A. Casal, "Coordination and decentralized cooperation of multiple mobile manipulators," *Journal of Robotic Systems*, vol. 13, no. 11, pp. 755–764, 1996.

[3] B. Bayle, J.-Y. Fourquet, and M. Renaud, "Manipulability analysis for mobile manipulators," in *Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation (Cat. No. 01CH37164)*, vol. 2. IEEE, 2001, pp. 1251–1256.

[4] G. Casalino and A. Turetta, "Coordination and control of multiarm, non-holonomic mobile manipulators," in *Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003)(Cat. No. 03CH37453)*, vol. 3. IEEE, 2003, pp. 2203–2210.

[5] R. Jamisola, M. H. Ang, D. Oetomo, O. Khatib, T. M. Lim, and S. Y. Lim, "The operational space formulation implementation to aircraft canopy polishing using a mobile manipulator," in *Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No. 02CH37292)*, vol. 1. IEEE, 2002, pp. 400–405.

[6] G. Antonelli, F. Arrichiello, and S. Chiaverini, "The Null-Space-based Behavioral control for autonomous robotic systems," *Journal of Intelligent Service Robotics*, vol. 1, no. 1, pp. 27–39, Jan. 2008. [Online]. Available: http://dx.doi.org/10.1007/s11370-007-0002-3

[7] ——, "The NSB control: a behavior-based approach for multi-robot systems," *Paladyn Journal of Behavioral Robotics*, vol. 1, no. 1, pp. 48–56, 2010.

[8] K. Baizid, G. Giglio, F. Pierri, M. A. Trujillo, G. Antonelli, F. Caccavale, A. Viguria, S. Chiaverini, and A. Ollero, "Behavioral control of unmanned aerial vehicle manipulator systems," *Autonomous Robots*, vol. 41, no. 5, pp. 1203–1220, 2017.

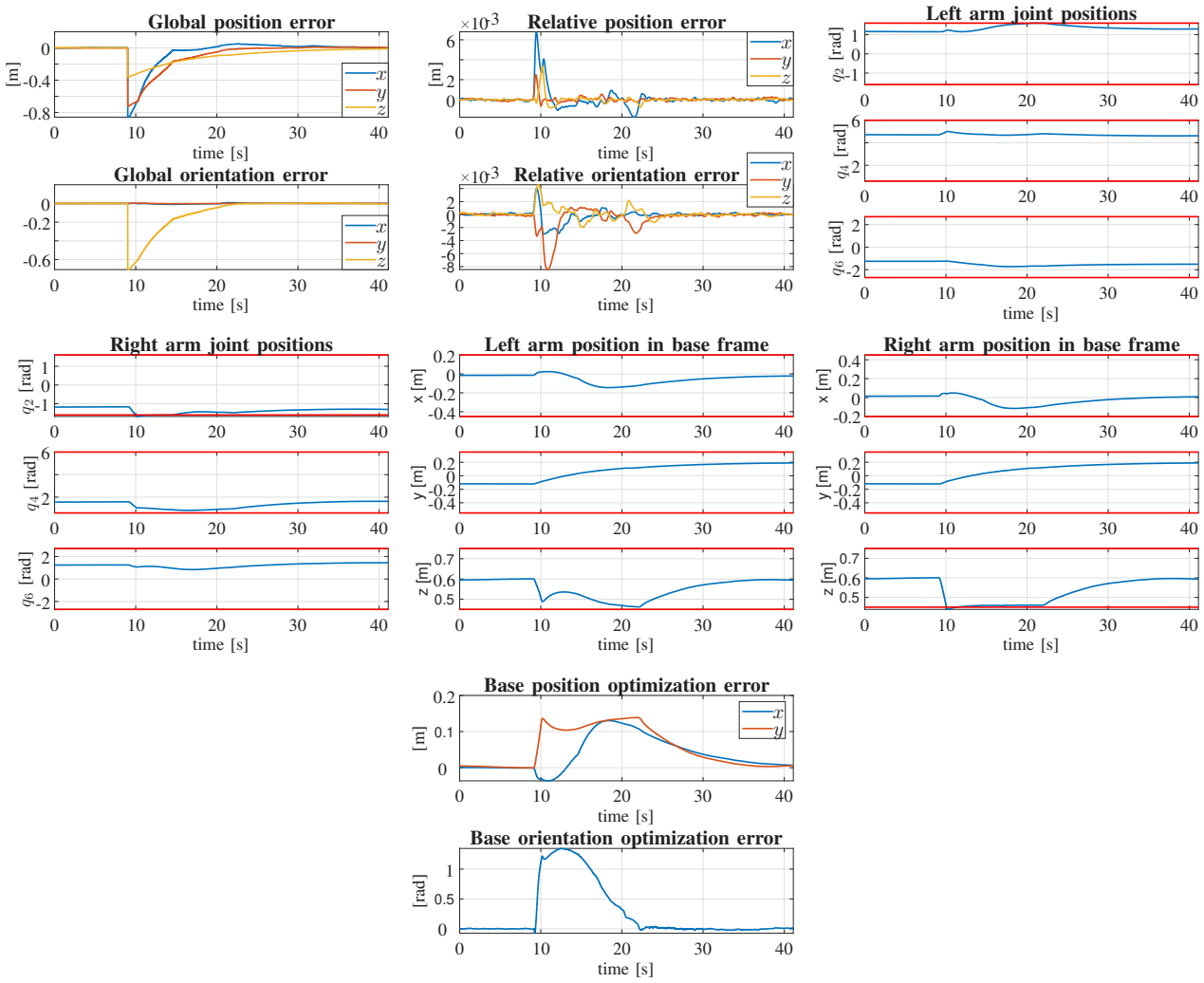

Fig. 5. Experiment, safety + operational + optimization tasks. From top-left to bottom-right: global pose error; relative pose error; left and right arm joint positions together with the imposed safety thresholds; left and right arm pose expressed in their base frames together with the safety thresholds imposed by the virtual wall tasks; optimization task error. All the safety task are respected and the optimization function aligns the center of the tray to the robot body.

[9] G. Muscio, F. Pierri, M. Trujillo, E. Cataldi, G. Antonelli, F. Caccavale, A. Viguria, S. Chiaverini, and A. Ollero, "Coordinated control of aerial robotic manipulators: Theory and experiments," *IEEE Transactions on Control Systems Technology*, vol. 26, no. 4, pp. 1406–1413, 2018.

[10] A. Escande, N. Mansard, and P.-B. Wieber, "Hierarchical quadratic programming: Fast online humanoid-robot motion generation," *International Journal of Robotics Research*, vol. 33, no. 7, pp. 1006–1028, 2014.

[11] H. Azimian, T. Looi, and J. Drake, "Closed-loop inverse kinematics under inequality constraints: Application to concentric-tube manipulators," in *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*. IEEE, 2014, pp. 498–503.

[12] E. Simetti, G. Casalino, S. Torelli, A. Sperindé, and A. Turetta, "Floating underwater manipulation: Developed control methodology and experimental validation within the TRIDENT project," *Journal of Field Robotics*, vol. 31(3), pp. 364–385, 2013.

[13] S. Moe, G. Antonelli, A. Teel, K. Pettersen, and J. Schrimpf, "Set-based tasks within the singularity-robust multiple task-priority inverse kinematics framework: General formulation, stability analysis and experimental results," *Frontiers in Robotics and AI*, vol. 3, p. 16, 2016.

[14] E. Cataldi, F. Real, A. Suarez, P. Di Lillo, F. Pierri, G. Antonelli, F. Caccavale, G. Heredia, and A. Ollero, "Set-based inverse kinematics control of an anthropomorphic dual arm aerial manipulator," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 2960–2966.

[15] S. Chiaverini, "Singularity-robust task-priority redundancy resolution for real-time kinematic control of robot manipulators," *IEEE Transactions on Robotics and Automation*, vol. 13, no. 3, pp. 398–410, 1997.

[16] B. Siciliano and J.-J. E. Slotine, "A general framework for managing multiple tasks in highly redundant robotic systems," in *Proc. Fifth International Conference on Advanced Robotics (ICAR)*, Pisa, Italy, 1991, pp. 1211–1216.

[17] P. Di Lillo, F. Arrichiello, D. Di Vito, and G. Antonelli, "BCI-controlled assistive manipulator: developed architecture and experimental results," *IEEE Transactions on Cognitive and Developmental Systems*, pp. 1–1, 2020.

[18] N. M. Ceriani, A. M. Zanchettin, P. Rocco, A. Stolt, and A. Robertsson, "Reactive task adaptation based on hierarchical constraints classification for safe industrial robots," *IEEE/ASME Transactions on Mechatronics*, vol. 20, no. 6, pp. 2935–2949, 2015.

[19] P. Di Lillo, F. Arrichiello, G. Antonelli, and S. Chiaverini, "Safety-related tasks within the set-based task-priority inverse kinematics framework," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct 2018, pp. 6130–6135.

[20] R. E. Roberson and R. Schwertassek, *P Dynamics of Multibody Systems*. Berlin, D: Springer-Verlag, 1988.

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

[22] F. Caccavale and M. Uchiyama, *Springer Handbook of Robotics*. Heidelberg, D: B. Siciliano, O. Khatib, (Eds.), Springer-Verlag, 2016, ch. Cooperative Manipulation, pp. 989–1006.