Kinematic Control with Force Feedback for a Redundant Bimanual Manipulation System

Fabrizio Caccavale, Vincenzo Lippiello, Giuseppe Muscio, Francesco Pierri, Fabio Ruggiero, Luigi Villani

Abstract— **In this paper, a kinematic model for motion coordination and control of a redundant robotic dual-arm/hand system is derived, which allows to compute the object pose from the joint variables of each arm and each finger as well as from a suitable set of contact variables. This model is used to design a two-stage control scheme to achieve a desired object motion and maintain desired normal contact forces applied to the object. Several secondary tasks are accomplished through a prioritized task sequencing management of the whole system redundancy. A simulation case study is presented to demonstrate the effectiveness of the proposed approach.**

I. INTRODUCTION

Dual-arm/hand object manipulation with multi-fingered hands is a challenging task, especially in service robotics applications, but it has not investigated as extensively as it should deserve. In order to achieve the desired motion of the manipulated object, arms and fingers should operate in a coordinated fashion. In the absence of physical interaction between the fingers and the object, simple motion synchronization shall be ensured. Further, the execution of object grasping or manipulation requires controlling also the interaction forces to ensure grasp stability [10], [13].

From a kinematics point of view, an object manipulation task can be assigned in terms of the motion of the fingertips and/or in terms of the desired object motion. The planner (or the controller) has to map the desired task into the corresponding joint trajectories of the fingers and the arms, thus requiring the solution of an inverse kinematics problem.

In this paper, starting from the framework presented in [5], a kinematic model for object manipulation using a dual-arm/hand robotic system is derived, which allows to compute the object pose from the joint variables of each arm and each finger (active joints), as well as from a set of contact variables, modelled as passive joints [8]. Suitable conditions are derived ensuring that a given motion can be imposed to the object using only the active joints. Exploiting also the information provided by force sensors mounted inside the fingertips, a two-stage control scheme is proposed so as to achieve the desired object motion and to maintain the desired contact normal forces.

The research leading to these results has been supported by the DEXMART Large-scale integrating project, funded by the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement ICT-216239. The authors are solely responsible for its content.

F. Caccavale, G. Muscio, F. Pierri are with Dipartimento di Ingegneria e Fisica dell'Ambiente, Università della Basilicata, via dell'Ateneo Lucano 10, 85100, Potenza, Italy `francesco.pierri@unibas.it`

V. Lippiello, F. Ruggiero, L. Villani are with PRISMA Lab, Dipartimento di Informatica e Sistemistica, Università di Napoli Federico II, via Claudio 21, 80125, Naples, Italy `fabio.ruggiero@unina.it`

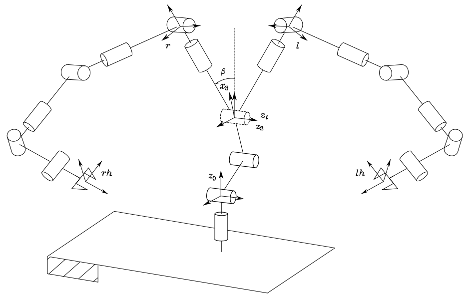

Fig. 1. Kinematic structure of a humanoid manipulator with torso and arms inspired to the DLR Justin.

The kinematic redundancy of the system, deriving also from the presence of the passive joints, is suitably exploited to satisfy a certain number of secondary tasks with lower priority, aimed at ensuring grasp stability and manipulation dexterity – without violating system constraints – besides the main task corresponding to the desired object motion. To this aim, a prioritized task sequencing with smooth transitions between the different tasks [6] is employed.

At the best of authors knowledge, the focus of previous papers on kinematics of multi-fingered manipulation was on constrained kinematic control [4], [8], or manipulability analysis [2], without considering redundancy resolution and the benefits of integrating a force feedback in a kinematic control loop. The effectiveness of the proposed approach is demonstrated in simulation by considering an object exchange task for a planar bimanual system.

II. KINEMATIC MODEL

A. Robot kinematics

Consider a bimanual manipulation system, e.g., the humanoid manipulator of Fig. 1 composed by a three DOFs torso and two DLR manipulators (each with seven DOFs). The direct kinematics can be computed as reported in [14], by introducing a frame Σ_b fixed with the base of the torso, two frames, Σ_r and Σ_l, attached at the base of the right and left arm, respectively, and two frames, Σ_{rh} and Σ_{lh}, attached to the palms of the right and left hand, respectively. Moreover, assuming that each arm ends with a robotic hand composed by N fingers, it is useful to introduce a frame Σ_{rf_i} (Σ_{lf_i}), attached to the distal phalanx of finger i ($i = 1 \dots N$) of the right (left) hand.

The pose of Σ_{rf_i} with respect to the base frame Σ_b can be represented by the well known (4×4) homogeneous transformation matrix $\boldsymbol{T}^b_{rf_i}(\boldsymbol{R}^b_{rf_i}, \boldsymbol{o}^b_{rf_i})$, where $\boldsymbol{R}^b_{rf_i}$ is the (3×3) rotation matrix expressing the orientation of Σ_{rf_i} with respect to the base frame and $\boldsymbol{o}^b_{rf_i}$ is the (3×1) position

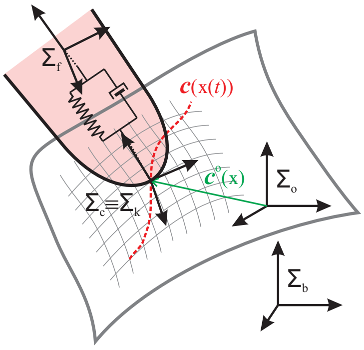

Fig. 2. Local parametrization of the object surface with respect to Σ_o

vector of the origin of Σ_{rf_i} with respect to the base frame. Hence, the direct kinematics can be expressed as

$$\boldsymbol{T}^b_{rf_i} = \boldsymbol{T}^b_r(\boldsymbol{q}_t)\boldsymbol{T}^r_{rh}(\boldsymbol{q}_{rh})\boldsymbol{T}^{rh}_{rf_i}(\boldsymbol{q}_{rf_i}) \tag{1}$$

where $\boldsymbol{T}^b_r$ is the matrix relating the frame at the basis of the right arm to the base frame (which depends, in turn, on the torso joint vector, $\boldsymbol{q}_t$), $\boldsymbol{T}^r_{rh}(\boldsymbol{q}_{rh})$ is the matrix relating the right palm frame to the base frame of the right arm (which depends, in turn, on the joint vector of the right arm, $\boldsymbol{q}_{rh}$), and $\boldsymbol{T}^{rh}_{rf_i}$ is the matrix relating the frame attached to finger i to the palm frame of the right hand (which depends, in turn, on the joint vector $\boldsymbol{q}_{rf_i}$, where the fingers are assumed to be identical). An equation similar to (1) holds for the left hand fingers, with subscript l in place of subscript r.

Due to the branched structure of the manipulator, the kinematic equations of both the right and the left arm depend on the joint vector $\boldsymbol{q}_t$ of the torso and, thus, they are not independent. Without loss of generality, hereafter it is assumed that the torso is motionless, i.e., $\boldsymbol{q}_t$ is constant; therefore, the kinematics of the right and of the left hand can be considered separately. Hence, in the sequel, the superscripts r and l will be omitted and will be used explicitly only when it is required to distinguish between the right and the left arm.

The velocity of frame Σ_{f_i} with respect to the base frame can be represented by the (6×1) twist vector $\boldsymbol{v}_{f_i} = [\dot{\boldsymbol{o}}^T_{f_i} \quad \boldsymbol{\omega}^T_{f_i}]^T$, where $\boldsymbol{\omega}_{f_i}$ is the angular velocity, such that $\dot{\boldsymbol{R}}_{f_i} = \boldsymbol{S}(\boldsymbol{\omega}_{f_i})\boldsymbol{R}_{f_i}$, with $\boldsymbol{S}(\cdot)$ the skew-symmetric operator representing the vector product. The superscript b, denoting the base frame, has been omitted to simplify notation.

The differential kinematics equation relating the joint velocities to the velocity of frame Σ_{f_i} can be written as

$$\boldsymbol{v}_{f_i} = \begin{bmatrix} \boldsymbol{J}_{P_i}(\boldsymbol{q}_i) \\ \boldsymbol{J}_{O_i}(\boldsymbol{q}_i) \end{bmatrix} \dot{\boldsymbol{q}}_i = \boldsymbol{J}_{F_i}(\boldsymbol{q}_i)\dot{\boldsymbol{q}}_i, \tag{2}$$

where $\boldsymbol{q}^T_i = [\boldsymbol{q}^T_h \quad \boldsymbol{q}^T_{f_i}]^T$ and $\boldsymbol{J}_{F_i}$ is the Jacobian of the arm, ending with finger i.

Therefore, the differential kinematics equation of the whole arm-hand system, considering the N fingers as end-effectors, can be written in the form

$$\tilde{\boldsymbol{v}}_f = \boldsymbol{J}(\boldsymbol{q})\dot{\boldsymbol{q}}, \tag{3}$$

where $\tilde{\boldsymbol{v}}^{\mathrm{T}}_f = [\boldsymbol{v}^{\mathrm{T}}_{f_1} \cdots \boldsymbol{v}^{\mathrm{T}}_{f_N}]^{\mathrm{T}}$, $\boldsymbol{q}^{\mathrm{T}} = [\boldsymbol{q}^{\mathrm{T}}_h \quad \boldsymbol{q}^{\mathrm{T}}_1 \cdots \boldsymbol{q}^{\mathrm{T}}_N]^{\mathrm{T}}$, and $\boldsymbol{J}$ is the Jacobian of the overall arm-hand system.

B. Contact kinematics

Assuming that the hand grasps a rigid object, to derive the kinematic mapping between the joint variables of the arm-hand system and the pose (position and orientation) of the object, it is useful introducing an object frame Σ_o attached to the object, usually chosen with the origin in the object center of mass. Let $\boldsymbol{R}_o$ and $\boldsymbol{o}_o$ denote respectively the rotation matrix and the position vector of the origin of Σ_o with respect to the base frame, and let $\boldsymbol{v}_o$ denote the velocity twist vector.

Grasping situations may involve moving rather than fixed contacts: often, both the object and the robotic fingers are smooth surfaces, and manipulation involves rolling and/or sliding of the fingertips on the object surface, depending on the contact type. If the fingers and object shapes are completely known, the contact kinematics can be described introducing contact coordinates defined on the basis of a suitable parametrization of the contact surfaces [7], [9].

In this work, it is assumed that the fingertips are sharp (i.e. they end with a point, denoted as tip point) and covered by an elastic pad. The elastic contact is modeled by introducing a finger contact frame, Σ_{k_i}, attached to the soft pad and with the origin in the tip point $\boldsymbol{o}_{k_i}$, and a spring-damper system connecting $\boldsymbol{o}_{k_i}$ with the origin of frame Σ_{f_i}, attached to the rigid part of the finger (see Fig. 2) and with the same orientation of Σ_{k_i}. The displacement between Σ_{f_i} and Σ_{k_i}, due to the elastic contact force, can be computed as

$$\boldsymbol{o}_{f_i} - \boldsymbol{o}_{k_i} = (l_i - \Delta l_i)\boldsymbol{R}_o\hat{\boldsymbol{n}}^o(\boldsymbol{\xi}), \tag{4}$$

where l_i and $0 \le \Delta l_i \le l_i$ are the rest position and the compression of the spring, respectively, and $\hat{\boldsymbol{n}}^o$ is the vector representing the outward normal to the object's surface at the contact point, referred to Σ_o.

Let Σ_{c_i} be the contact frame attached to the object, with the origin at the contact point $\boldsymbol{o}_{c_i}$. Notice that, instantaneously, the object contact point $\boldsymbol{o}_{c_i}$ and the finger contact point $\boldsymbol{o}_{k_i}$ are coincident. One of the axes of Σ_{c_i}, e.g., the Z axis, is assumed to be the outward normal to the tangent plane to the object surface at the contact point.

It is assumed that, at least locally, the position of the contact point with respect to the object frame $\boldsymbol{o}^o_{o,c_i} = \boldsymbol{o}^o_{c_i} - \boldsymbol{o}^o_o$ can be parameterized in terms of a coordinate chart $\boldsymbol{c}^o_i : U_i \subset \mathbb{R}^2 \mapsto \mathbb{R}^3$ which maps a point $\boldsymbol{\xi}_i = [u_i \quad v_i]^{\mathrm{T}} \in U_i$ to the point $\boldsymbol{o}^o_{o,c_i}(\boldsymbol{\xi}_i)$ of the surface of the object.

Assuming that $\boldsymbol{c}^o_i$ is a diffeomorphism and that the coordinate chart is orthogonal and right-handed, contact frame Σ_{c_i} can be chosen as a Gauss frame [7], where the the relative orientation expressed by the rotation matrix $\boldsymbol{R}^o_{c_i}$ is computed as function of the orthogonal tangent vectors $\boldsymbol{c}^o_{u_i} = \partial\boldsymbol{c}^o_i/\partial u_i$ and $\boldsymbol{c}^o_{v_i} = \partial\boldsymbol{c}^o_i/\partial v_i$ [5].

Consider the contact kinematics from the object point of view. Let $\boldsymbol{c}^o_i(\boldsymbol{\xi}_i(t))$ denote a curve on the surface of the object, with $\boldsymbol{\xi}_i(t) \in U$ (see Fig. 2). The corresponding motion of Σ_{c_i} with respect to the base frame can be determined as a function of: object motion, geometric parameters of the object and the curve geometric features. Namely, the velocity of the contact frame can be expressed as

$$\boldsymbol{v}_{c_i} = \begin{bmatrix} \dot{\boldsymbol{o}}_{c_i} \\ \boldsymbol{\omega}_{c_i} \end{bmatrix} = \boldsymbol{G}_{\xi_i}^{\mathrm{T}}(\boldsymbol{\xi}_i)\boldsymbol{v}_{o_i} + \boldsymbol{J}_{\xi_i}(\boldsymbol{\xi}_i)\dot{\boldsymbol{\xi}}_i, \tag{5}$$

where $\boldsymbol{G}_{\xi_i}(\boldsymbol{\xi}_i)$ and $\boldsymbol{J}_{\xi_i}(\boldsymbol{\xi}_i)$ are respectively (6×6) and (6×2) full rank matrices, whose expressions can be found in [5].

Consider now the contact kinematics from the fingers point of view. The contact can be modeled with an unactuated 3-DOF ball and socket kinematic pair centered at the origin, $\boldsymbol{o}_{k_i}$, of Σ_{k_i}, fixed to the soft pad of the finger; the origin may also move on the surface, if sliding is allowed. Therefore, the relative orientation $\boldsymbol{R}_{c_i}^{k_i}$ of Σ_{c_i} with respect to Σ_{k_i} can be computed in terms of a suitable parametrization of the ball and socked joint, e.g., Euler angles.

A vector $\boldsymbol{\theta}_i = \begin{bmatrix} \theta_{i1} & \theta_{i2} & \theta_{i3} \end{bmatrix}^{\mathrm{T}}$ of XYZ Euler angles can be considered, thus $\boldsymbol{R}_{c_i}^{k_i} = \boldsymbol{R}_{c_i}^{k_i}(\boldsymbol{\theta}_i)$. Singularities occurs for $\theta_{2i} = \pm\pi/2$, but they do not correspond to physical singularities of the kinematics pair.

Notice that, in the presence of a contact force, because of the tip elasticity, frame Σ_{k_i} translates from the finger frame Σ_{f_i} according to (4), but the orientation does not change. Therefore, $\boldsymbol{R}_{c_i}^{k_i} = \boldsymbol{R}_{c_i}^{f_i}$. Moreover, the angular velocity of Σ_{c_i} relative to Σ_{f_i} can be expressed as $\boldsymbol{\omega}_{f_i,c_i}^{f_i} = \boldsymbol{H}(\boldsymbol{\theta}_i)\dot{\boldsymbol{\theta}}_i$, where $\boldsymbol{H}$ is a transformation matrix depending on the joint parameterization. In view of the decomposition $\boldsymbol{\omega}_{c_i} = \boldsymbol{\omega}_{f_i} + \boldsymbol{R}_{f_i}(\boldsymbol{q}_i)\boldsymbol{\omega}_{f_i,c_i}^{f_i}$, and from (2), the angular velocity of Σ_{c_i} can be computed also as a function of joint and contact variables in the form

$$\boldsymbol{\omega}_{c_i} = \boldsymbol{J}_{O_i}(\boldsymbol{q}_i)\dot{\boldsymbol{q}}_i + \boldsymbol{R}_{f_i}(\boldsymbol{q}_i)\boldsymbol{H}(\boldsymbol{\theta}_i)\dot{\boldsymbol{\theta}}_i, \tag{6}$$

with $\boldsymbol{J}_{O_i}$ defined in (2). Moreover, since the origins of Σ_{c_i} and Σ_{k_i} coincide, the following equalities hold

$$\begin{aligned}
\boldsymbol{o}_{c_i} &= \boldsymbol{o}_{k_i} = \boldsymbol{o}_{f_i} - (l_i - \Delta l_i)\boldsymbol{R}_o\hat{\boldsymbol{n}}_i^o(\boldsymbol{\xi}_i), \\
\dot{\boldsymbol{o}}_{c_i} &= \boldsymbol{J}_{P_i}(\boldsymbol{q}_i)\dot{\boldsymbol{q}}_i + \dot{\Delta l}_i\boldsymbol{R}_o\hat{\boldsymbol{n}}_i^o(\boldsymbol{\xi}_i) \\
&\quad + (l_i - \Delta l_i)\boldsymbol{S}(\boldsymbol{R}_o\hat{\boldsymbol{n}}_i^o(\boldsymbol{\xi}_i))\boldsymbol{\omega}_o - (l_i - \Delta l_i)\boldsymbol{R}_o\frac{\partial\hat{\boldsymbol{n}}_i^o(\boldsymbol{\xi}_i)}{\partial\boldsymbol{\xi}_i}\dot{\boldsymbol{\xi}}_i,
\end{aligned} \tag{7}$$

with $\boldsymbol{J}_{P_i}$ defined in (2). Using (6) and (7), the velocity of the contact frame can be expressed as

$$\begin{aligned}
\boldsymbol{v}_{c_i} &= \boldsymbol{J}_{F_i}(\boldsymbol{q})\dot{\boldsymbol{q}} + \boldsymbol{J}_{\theta_i}(\boldsymbol{\theta}_i, \boldsymbol{q}_i)\dot{\boldsymbol{\theta}}_i + \boldsymbol{J}_{\Delta l_i}(\boldsymbol{\xi}_i)\dot{\Delta l}_i \\
&\quad - \boldsymbol{J}_{\xi_i}'(\boldsymbol{\xi}_i, \Delta l_i)\dot{\boldsymbol{\xi}}_i - \boldsymbol{G}_{\Delta l_i}^{T}(\boldsymbol{\xi}_i, \Delta l_i)\boldsymbol{v}_o,
\end{aligned} \tag{8}$$

where $\boldsymbol{J}_{\theta_i}$ is a (6×3) full rank matrix, whose detailed expression can be found in [5], $\boldsymbol{J}_{\Delta l_i}$ is a (6×1) vector

$$\boldsymbol{J}_{\Delta l_i} = \begin{bmatrix} \boldsymbol{R}_o\hat{\boldsymbol{n}}_i^o(\boldsymbol{\xi}_i) \\ \boldsymbol{0} \end{bmatrix},$$

$\boldsymbol{J}_{\xi_i}'$ is a (6×2) full rank matrix

$$\boldsymbol{J}_{\xi_i}' = \begin{bmatrix} (l - \Delta l_i)\boldsymbol{R}_o\dfrac{\partial\hat{\boldsymbol{n}}_i^o(\boldsymbol{\xi}_i)}{\partial\boldsymbol{\xi}_i} \\ \boldsymbol{0} \end{bmatrix},$$

and $\boldsymbol{G}_{\Delta l_i}$ is the (6×6) matrix

$$\boldsymbol{G}_{\Delta l_i} = \begin{bmatrix} \boldsymbol{0} & \boldsymbol{0} \\ (\Delta l_i - l_i)\boldsymbol{S}(\boldsymbol{R}_o\hat{\boldsymbol{n}}_i^o(\boldsymbol{\xi}_i)) & \boldsymbol{0} \end{bmatrix}.$$

Hence, from (5) and (8), the contact kinematics of finger i has the form

$$\begin{aligned}
\boldsymbol{J}_{F_i}(\boldsymbol{q}_i)\dot{\boldsymbol{q}}_i + \boldsymbol{J}_{\eta_i}(\boldsymbol{\eta}_i, \boldsymbol{q}_i, \Delta l_i)\dot{\boldsymbol{\eta}}_i + \boldsymbol{J}_{\Delta l_i}(\boldsymbol{\xi})\dot{\Delta l}_i = \\
\boldsymbol{G}_i^{\mathrm{T}}(\boldsymbol{\eta}_i, \Delta l_i)\boldsymbol{v}_o,
\end{aligned} \tag{9}$$

where $\boldsymbol{\eta}_i = \begin{bmatrix} \boldsymbol{\xi}_i^{\mathrm{T}} & \boldsymbol{\theta}_i^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}}$ is the vector of contact variables, $\boldsymbol{J}_{\eta_i} = \begin{bmatrix} -(\boldsymbol{J}_{\xi_i} + \boldsymbol{J}_{\xi_i}') & \boldsymbol{J}_{\theta_i} \end{bmatrix}$ is a (6×5) full rank matrix, and $\boldsymbol{G}_i = \boldsymbol{G}_{\xi_i} + \boldsymbol{G}_{\Delta l_i}$ is a (6×6) full rank grasp matrix. This equation can be interpreted as the differential kinematics equation of an "extended" finger corresponding to the kinematic chain including the arm and finger joint variables (active joints) and the contact variables (passive joints), from the base frame to the contact frame [8].

It is worth noticing that equation (9) involves all the 6 components of the velocity, differently from the grasping constraint equation usually considered (see, e.g., [9]), which contains only the components of the velocities that are transmitted by the contact. The reason is that the above formulation takes into account also the velocity components not transmitted by contact i, parameterized by the contact variables and lying in the range space of $\begin{bmatrix} \boldsymbol{J}_{\eta_i} & \boldsymbol{J}_{\Delta l_i} \end{bmatrix}$. As a consequence, $\boldsymbol{G}_i$ is always a full rank matrix.

Depending on the considered contact type [13], some of the parameters of $\boldsymbol{\xi}_i$ and $\boldsymbol{\theta}_i$ are constant. Hence, assuming that the contact type remains unchanged during the task, the variable parameters at each contact point are grouped in a $(n_{c_i}\times1)$ vector $\boldsymbol{\eta}_i$ of contact variables, with $n_{c_i} \leq 5$.

Differently form the classical grasp analysis, in this work the elasticity of the soft pad has been explicitly modeled (although using a simplified model). This means that the force along the normal to the contact surface is always of elastic type. The quantity Δl_i, at steady state, is related to the normal contact force f_{ni} by the equation $\Delta l_i = f_{ni}/k_i$, being k_i the elastic constant of the soft pad of finger i.

C. Kinematic analysis of the grasp

Object dynamic manipulation is, in general, a difficult task, since the number of the control variables (the active joints) is lower than the number of configuration variables (active and passive joints). However, in some particular situations, it is possible to simplify the analysis, considering only the kinematics of the system.

To this purpose, assume that force sensors are available on the fingertips and a force control strategy is employed to ensure a desired constant contact forces f_{di} along the direction normal to the contact point. Therefore, $\Delta l_i = \Delta l_{di} = f_{di}/k_i$ can be assumed to be fixed ($\dot{\Delta l}_i = 0$) and equation (9) can be rewritten as

$$\boldsymbol{J}_{F_i}(\boldsymbol{q}_i)\dot{\boldsymbol{q}}_i + \boldsymbol{J}_{\eta_i}(\boldsymbol{\eta}_i, \boldsymbol{q}_i, \Delta l_i)\dot{\boldsymbol{\eta}}_i = \boldsymbol{G}_i^{\mathrm{T}}(\boldsymbol{\eta}_i, \Delta l_i)\boldsymbol{v}_o, \tag{10}$$

On the basis of (10), it is possible to make a kinematic classification of the grasp [13].

A grasp is *redundant* if the null space of the matrix $\begin{bmatrix} \boldsymbol{J}_{F_i} & \boldsymbol{J}_{\eta_i} \end{bmatrix}$ is non-null, for at least one finger i. In this case, the mapping between the joint variables of "extended" finger i and the object velocity is many to one: motions of active and passive joints of the extended finger are possible when the object is locked.

A grasp is *indeterminate* if the intersection of the null spaces of $[-\boldsymbol{J}_{\eta_i} \quad \boldsymbol{G}_i^{\mathrm{T}}]$, for all $i = 1, \ldots, N$, is non-null. In this case, motions of the object and of the passive joints are possible when the active joints of all the fingers are locked.

It is worth noticing that, also in the case of redundant and indeterminate grasps, for a given object pose and fingers configuration, the value of the contact variables is uniquely determined. More details can be found in [5].

III. Control scheme with redundancy resolution

In the case of kinematically determinate and, possibly, redundant grasp, a two-stage control scheme is proposed for the dual arm-hand manipulation system. The first stage is an inverse kinematics scheme with redundancy resolution, which computes the joint references for the active joints corresponding to a desired object's motion –assigned in terms of the homogeneous transformation matrix $\boldsymbol{T}_d$ and the corresponding twist velocity vector $\boldsymbol{v}_{o_d}$– and to the desired normal contact force $\boldsymbol{f}_d^T = \left[f_{d1} \cdots f_{dN}\right]$. The second stage is a parallel control composed by a PD position controller and a PI tip force controller, ensuring the desired object motion and desired contact forces on the basis of the previously computed joint references.

Namely, in ideal conditions, the joint references computed by the kinematic stage ensure tracking of the desired object motion, with the desired contact forces. In the presence of modeling errors and parameters uncertainty, the contact forces may differ from those planned. Using the force sensors at the fingertips, a force control strategy is adopted to ensure the desired contact force by modifying the joint references computed by the inverse kinematics stage. In principle, the joint references of the overall manipulation system could be involved; however, it is reasonable to design a force controller acting only on the joints of the fingers.

In order to derive the equations of the first stage, starting from (9), it is useful to write the differential kinematic equations of the whole (right or left) arm-hand system as

$$\boldsymbol{J}(\boldsymbol{q})\dot{\boldsymbol{q}} + \boldsymbol{J}_\eta(\boldsymbol{\eta}, \boldsymbol{q}, \Delta\boldsymbol{l})\dot{\boldsymbol{\eta}} = \boldsymbol{G}^{\mathrm{T}}(\boldsymbol{\eta}, \Delta\boldsymbol{l})\tilde{\boldsymbol{v}}_o, \qquad (11)$$

where $\boldsymbol{J}$ is the Jacobian of the arm-hand system defined in (3), $\boldsymbol{J}_\eta = \mathrm{diag}\{\boldsymbol{J}_{\eta_1}, \cdots, \boldsymbol{J}_{\eta_N}\}$ is a block diagonal matrix corresponding to the vector of passive joints $\boldsymbol{\eta}^{\mathrm{T}} = \left[\boldsymbol{\eta}_1^{\mathrm{T}} \cdots \boldsymbol{\eta}_N^{\mathrm{T}}\right]^{\mathrm{T}}$, $\boldsymbol{G}$ is the block diagonal grasp matrix $\boldsymbol{G} = \mathrm{diag}\{\boldsymbol{G}_1, \cdots, \boldsymbol{G}_N\}$, $\Delta\boldsymbol{l}^{\mathrm{T}} = \left[\Delta l_1 \cdots \Delta l_N\right]^{\mathrm{T}}$ and $\tilde{\boldsymbol{v}}_o^{\mathrm{T}} = \left[\boldsymbol{v}_o^{\mathrm{T}} \cdots \boldsymbol{v}_o^{\mathrm{T}}\right]^{\mathrm{T}}$.

From Eq. (11), the following closed-loop inverse kinematics algorithm can be derived:

$$\begin{bmatrix} \dot{\boldsymbol{q}}_d \\ \dot{\boldsymbol{\eta}}_d \end{bmatrix} = \tilde{\boldsymbol{J}}^{\dagger}(\boldsymbol{q}_d, \boldsymbol{\eta}_d, \Delta l_d)\boldsymbol{G}^{\mathrm{T}}(\tilde{\boldsymbol{v}}_{o_d} + \boldsymbol{K}_o\tilde{e}_o) + \boldsymbol{N}_o\boldsymbol{\sigma}, \quad (12)$$

where $\tilde{\boldsymbol{J}} = \left[\boldsymbol{J} \quad \boldsymbol{J}_\eta\right]$, the symbol $\dagger$ denotes a right (weighted) pseudo-inverse, $\tilde{\boldsymbol{v}}_{o_d}^{\mathrm{T}} = \left[\boldsymbol{v}_{o_d}^{\mathrm{T}} \cdots \boldsymbol{v}_{o_d}^{\mathrm{T}}\right]^{\mathrm{T}}$, $\boldsymbol{K}_o$ is a diagonal and positive definite matrix gain, $\tilde{\boldsymbol{e}}_o^{\mathrm{T}} = \left[\boldsymbol{e}_{o_1}^{\mathrm{T}} \cdots \boldsymbol{e}_{o_N}^{\mathrm{T}}\right]^{\mathrm{T}}$, being $\boldsymbol{e}_{o_i}$ the pose error between the desired and the current object pose computed on the basis of the direct kinematics of the extended finger i, and $\boldsymbol{N}_o = \boldsymbol{I} - \tilde{\boldsymbol{J}}^{\dagger}\tilde{\boldsymbol{J}}$ is a projector in the null space of the Jacobian matrix $\tilde{\boldsymbol{J}}$. The quantity $\Delta\boldsymbol{l}_d$ in (12) is the vector collecting the finger soft pad deformations $\Delta l_{di} = f_{di}/k_i$ corresponding to the desired contact force f_{di}.

Equation (12) is used to compute the joint reference vector $\boldsymbol{q}_d$ for the controller of the second stage.

In view of the above considerations, any kind of joint motion control can be adopted for the arms of the bi-manual manipulation system, receiving as input the joint references computed by the inverse kinematics scheme. In this paper, the joint torques for finger i are set according to the parallel force/position control law

$$\boldsymbol{\tau}_i = \boldsymbol{J}_i^T(\boldsymbol{q}_i)\left(k_P\Delta\boldsymbol{p}_i + \boldsymbol{f}_{di} + k_F\Delta\boldsymbol{f}_{ni} + k_I\int_0^t \Delta\boldsymbol{f}_{ni}\mathrm{d}\tau\right) - k_d\dot{\boldsymbol{q}}_i + \boldsymbol{g}_i(\boldsymbol{q}_i) \qquad (13)$$

where $\boldsymbol{g}_i(\boldsymbol{q}_i)$ is the vector of the gravity torque of finger i, $\Delta\boldsymbol{p}_i$ denotes the position error of finger i between the desired value computed through direct kinematics starting from $\boldsymbol{q}_{d_i}$ and the current one, and $\Delta\boldsymbol{f}_{ni}$ is the projection of the force error along the normal to the object surface at the contact point. The above control law regulates the contact force to the desired value at the expense of a position error (i.e., a displacement of the positions of the fingers with respect to the palm), in the presence of uncertainties.

A. Redundancy resolution

Since the system may be highly redundant, multiple tasks could be fulfilled, provided that they are suitably arranged in a priority order, according to the *augmented projection method* [1]. Consider m secondary tasks, each expressed by a task function $\boldsymbol{\sigma}_{t_h}(\tilde{\boldsymbol{q}})$ $(h = 1, \ldots, m)$, where $\tilde{\boldsymbol{q}} = \left[\boldsymbol{q}_d^T \quad \boldsymbol{\eta}_d^T\right]^T$. According to the *augmented projection method* [1], the control law (12) can be replaced by

$$\dot{\tilde{\boldsymbol{q}}} = \tilde{\boldsymbol{J}}^{\dagger}(\tilde{\boldsymbol{q}}, \Delta\boldsymbol{l}_d)\boldsymbol{G}^{\mathrm{T}}(\tilde{\boldsymbol{v}}_{o_d} + \boldsymbol{K}_o\tilde{\boldsymbol{e}}_o) + \sum_{h=1}^{m}\boldsymbol{N}(\boldsymbol{J}_{t_h}^A)\boldsymbol{J}_{t_h}^{\dagger}\boldsymbol{K}_{t_h}\boldsymbol{e}_{t_h}, \qquad (14)$$

where $\boldsymbol{J}_{t_h}$ is the Jacobian of the hth task, $\boldsymbol{J}_{t_h}^A$ is the augmented Jacobian, given by

$$\boldsymbol{J}_{t_h}^A(\tilde{\boldsymbol{q}}, \Delta l_d) = \left[\tilde{\boldsymbol{J}}^T(\tilde{\boldsymbol{q}}, \Delta\boldsymbol{l}_d) \ \boldsymbol{J}_{t_1}^T(\tilde{\boldsymbol{q}}) \ \ldots \ \boldsymbol{J}_{t_{h-1}}^T(\tilde{\boldsymbol{q}})\right]^T.$$

$\boldsymbol{N}(\boldsymbol{J}_{t_h}^A)$ is a null projector of the matrix $\boldsymbol{J}_{t_h}^A$, $\boldsymbol{K}_{t_h}$ is a positive definite gain matrix and $\boldsymbol{e}_{t_h} = \boldsymbol{\sigma}_{t_{h\,d}} - \boldsymbol{\sigma}_{t_h}$ is the task error, being $\boldsymbol{\sigma}_{t_{h\,d}}$ the desired value of the hth task variable.

The augmented projection method can be also adopted to fulfill mechanical or environmental constraints, such as joint limits and obstacle avoidance (other fingers or the grasped object). To this aim, each constraint can be described by means of a cost function, $\mathcal{C}(\tilde{\boldsymbol{q}})$, increasing when the manipulator comes close to violate the constraint. In order to minimize the cost function, the manipulator could be moved according to the opposite of the gradient $-\nabla_{\tilde{\boldsymbol{q}}}^T\mathcal{C}(\tilde{\boldsymbol{q}})$, that could be considered as a fictitious force moving the manipulator away from configurations violating the constraints. In order to include the constraints in (14), an overall cost function $\mathcal{C}_\Sigma$, given by

$$\mathcal{C}_\Sigma(\tilde{\boldsymbol{q}}) = \sum_{c_s} \gamma_{c_s} \mathcal{C}_{c_s}(\tilde{\boldsymbol{q}}), \tag{15}$$

is introduced, where γ_{c_s} and $\mathcal{C}_{c_s}$ are a positive weight and a cost function, respectively, referred to the c_sth constraint.

IV. Task sequencing

If the system comes close to violate a constraint, a high level supervisor has to remove some secondary tasks and relax enough DOFs to fulfill the constraint [6]. To manage in a correct way removal/insertion of tasks from/into the stack (task sequencing), a suitable task supervisor was designed, based on a three layers architecture: the lower layer computes the motion variables on the basis of a stack of active tasks; the intermediate layer determines which tasks must be removed from the stack in order to respect the constraints; the upper layer verifies if the previously removed tasks can be pushed back in the stack.

A. Removal and insertion of the tasks

A task must be removed from the stack when the predicted value of the overall cost function at the next time step is above a suitable defined threshold, $\overline{\mathcal{C}}$. Let T be the sampling time adopted to implement the control law and κT the actual time, the configuration at the time instant $(\kappa+1)T$ can be estimated as follows

$$\widehat{\tilde{\boldsymbol{q}}}(\kappa+1) = \tilde{\boldsymbol{q}}(\kappa) + T\dot{\tilde{\boldsymbol{q}}}(\kappa). \tag{16}$$

Hence, a task must be removed from the stack if

$$\mathcal{C}_\Sigma\left(\widehat{\tilde{\boldsymbol{q}}}(\kappa+1)\right) \geq \overline{\mathcal{C}}. \tag{17}$$

Once it has been ascertained that a task must be removed from the stack, the problem is to detect which task has to be removed. To the purpose, several criteria have been proposed in [6], with the aim of verifying the conflict between the constraints and each task. In this paper, the overall cost function gradient is projected in the null space of the task Jacobian, i.e.,

$$\mathcal{P}_{t_h} = \left\| \boldsymbol{N}\left(\boldsymbol{J}_{t_h}\right)\left(-\nabla^T_{\tilde{\boldsymbol{q}}}\mathcal{C}_\Sigma\right)\right\|, \quad h = 1,\dots,m, \tag{18}$$

the task corresponding to the minimum of $\mathcal{P}_{t_h}$ is then removed, since its projection into the null-space of $\boldsymbol{J}_{t_h}$ should be, ideally, zero to ensure constraint fulfillment.

The tasks removed by the second layer must be reinserted in the stack as soon as possible, provided that the constraints will not be violated. A prediction of the $\mathcal{C}_\Sigma$ evolution at the next time step has to be evaluated by considering the effect of each task currently out of the stack, i.e.,

$$\widehat{\tilde{\boldsymbol{q}}}_{t_h}(\kappa+1) = \tilde{\boldsymbol{q}}(\kappa) + \boldsymbol{J}^{\dagger}_{t_h}\boldsymbol{e}_{t_h}(\kappa). \tag{19}$$

Therefore, let $\underline{\mathcal{C}} < \overline{\mathcal{C}}$ be a suitably chosen threshold, a task is pushed back in the stack if

$$\mathcal{C}_\Sigma\left(\widehat{\tilde{\boldsymbol{q}}}_{t_h}(\kappa+1)\right) \leq \underline{\mathcal{C}}. \tag{20}$$

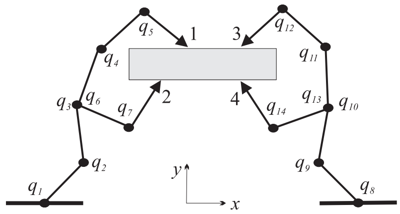

Fig. 3. Manipulation system.

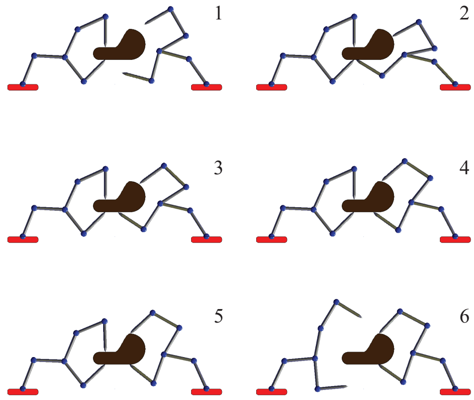

Fig. 4. Snapshots describing the case study.

B. Smooth transition and final control law

Task sequencing might cause discontinuities in the commanded joint velocities due to the change of active tasks in the stack. For each task a variable gain, ρ_{t_h}, is introduced to achieve a smooth behavior of the controller output

$$\rho_{t_h}(t) = \begin{cases} 1-\mathrm{e}^{-\mu(t-\tau)} & \text{if the } h\text{-th task is in the stack,} \\ \mathrm{e}^{-\mu(t-\tau')} & \text{if the } h\text{-th task is out of the stack,} \end{cases}$$

where τ and τ' are the time instant in which the task is inserted in the stack and the time instant in which it is removed, respectively, and $1/\mu$ is a time constant.

Hence, the first stage control law can be written in its complete form

$$\begin{aligned}\dot{\tilde{\boldsymbol{q}}} = {}& \tilde{\boldsymbol{J}}^{\dagger}(\tilde{\boldsymbol{q}}, \Delta\boldsymbol{l}_d)\boldsymbol{G}^{\mathrm{T}}(\tilde{\boldsymbol{v}}_{o_d} + \boldsymbol{K}_o\tilde{\boldsymbol{e}}_o) + \\ & + \sum_{h=1}^{m} \rho_{t_h}\boldsymbol{N}(\boldsymbol{J}^A_{t_h})\boldsymbol{J}^{\dagger}_{t_h}\boldsymbol{K}_{t_h}\boldsymbol{e}_{t_h} - k_\nabla\boldsymbol{N}(\boldsymbol{J}^A_{t_{m+1}})\nabla^T_{\tilde{\boldsymbol{q}}}\mathcal{C}_\Sigma,\end{aligned} \tag{21}$$

where k_∇ is a positive gain.

V. Case study

The presented control scheme has been tested on a manipulation system grasping a certain object, represented in Fig. 3, composed by two identical planar grippers, each with two branches and 7 DOFs, resulting in a total of $N = 4$ fingers and 14 active joints. The idea is that of performing an object exchange.

It is assumed, in its initial configuration, the system grasps the object with only tips 1 and 2, which are in a force closure condition, since the contact normal forces are acting on the same straight line [9]. Tips 3 and 4 approach the object until they reach a condition in which all the tips are in contact with the object. The main task consists in keeping the object still, while tips 3 and 4 move in order to achieve a force closure condition upon the object in a dexterous configuration, without violating a certain number of limits and constraints. Then, fingers 1 and 2 can leave the object, simulating in this way an hand-to-hand object passing.

The force control loop ensures that the planned forces are applied on the object. In this case study, the desired forces for tips 3 and 4 are negligible, since they have to slide on the object's surface so as to reconfigure themselves to reach force closure condition. The desired forces for tips 1 and 2 are dynamically planned, on the basis of the current value of the forces exerted by the fingers, in order to produce zero net force and moment on the object and to balance disturbances caused by movements of the other two fingertips.

A sequence of snapshots representing the described task are shown in Fig. 4. It can be noticed that, in the final configuration (fifth snapshot), fingers 3 and 4 are in a force closure condition, since the normals at the contact points act on the same straight line.

A. Subtasks and constraints

Four different subtasks have been considered: the first two, aimed at choosing the optimal contact points, are related to the grasp quality; the others regard the manipulability and the distance between the palm and the grasped object.

Unit frictionless equilibrium. The grasp quality can be guaranteed by moving the contact points on the object surface until the unit frictionless equilibrium is reached. This condition is a special case of a force-closure grasp; it is satisfied when two positive indices, called frictionless force (ε_f) and moment (ε_m) residuals, are zero [3], [11]

$$\varepsilon_f = \frac{1}{2}\boldsymbol{f}^T\boldsymbol{f}\,, \qquad \boldsymbol{f} = \sum_{i=1}^{4}\hat{\boldsymbol{n}}_i^o\,, \tag{22}$$

$$\varepsilon_m = \frac{1}{2}\boldsymbol{m}^T\boldsymbol{m}\,, \qquad \boldsymbol{m} = \sum_{i=1}^{4}\boldsymbol{c}_i^o \times \hat{\boldsymbol{n}}_i^o\,, \tag{23}$$

where $i = 1, \ldots, 4$, and where $\hat{\boldsymbol{n}}_i^o(\boldsymbol{\xi}_i)$ and $\boldsymbol{c}_i^o(\boldsymbol{\xi}_i)$ are the surface normal and the position of the ith contact point, respectively, both referred to the object frame. It has been shown that, for two or more contact points, unit frictionless equilibrium is a force closure condition for any nonzero friction coefficient [11], [12].

Manipulability. In order to keep the manipulator far from singularities, a manipulability index of each finger can be considered. In detail, the following manipulability measure, which vanishes at a singular configuration, is adopted for the ith finger [14]

$$w_i(\boldsymbol{q}_i) = \sqrt{det\left(\boldsymbol{J}_i(\boldsymbol{q}_i)\,\boldsymbol{J}_i^T(\boldsymbol{q}_i)\right)}\,, \quad i = 1, \ldots, 4\,. \tag{24}$$

The considered task function is then

$$\sigma_{w_i} = \begin{cases} \frac{1}{2}(\overline{w}_i - w_i\,(\boldsymbol{q}_i))^2, & \text{if}\, w_i\,(\boldsymbol{q}_i) < \overline{w}_i \\ 0, & \text{otherwise,} \end{cases} \tag{25}$$

where $\overline{w}_i$ is a threshold for the task activation. The desired value, $\sigma_{w_i d}$, is zero.

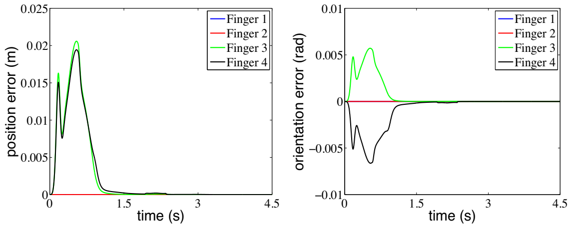

Fig. 5. Object's pose errors for each finger.

Distance between palm and object. Consider the position of the palm centroid in the object frame, $\boldsymbol{p}_c^o$, and a suitably chosen surface surrounding the object, $\mathcal{S}$, characterized by the equation $\mathcal{F}(\boldsymbol{p}^o) = 0$. When the centroid is inside the surface $\mathcal{S}$, a collision can occur; therefore, the centroid must be moved on the boundary, i.e, in a position such that $\mathcal{F}(\boldsymbol{p}_c^o) = 0$. Hence, the task function is the following

$$\sigma_P(\boldsymbol{p}_c^o) = \begin{cases} \mathcal{F}(\boldsymbol{p}_c^o), & \text{if the centroid is inside } \mathcal{S}, \\ 0, & \text{otherwise.} \end{cases} \tag{26}$$

In the following the two considered constraints are described.

Joint-limit avoidance. A physical constraint to the motion of the system is imposed by the mechanical joint limits. The system configuration is considered safe if $q_j \in [\underline{q}_j, \overline{q}_j]$, for $j = 1, \ldots, 14$, with $\underline{q}_j$ and $\overline{q}_j$ suitable chosen values far enough from the limits. The cost function, directly defined in the joint space, is the following

$$\mathcal{C}_{JL}(\boldsymbol{q}) = \sum_{j=1}^{14} c_j(q_j), \tag{27}$$

$$c_j(q_j) = \begin{cases} k_j\, e^{\delta(q_j - \underline{q}_j)^2} - 1, & \text{if } q_j \le \underline{q}_j, \\ 0, & \text{if } \underline{q}_j < q_j \le \overline{q}_j, \\ k_j\, e^{\delta(q_j - \overline{q}_j)^2} - 1, & \text{if } q_j > \overline{q}_j. \end{cases}$$

where k_j and δ are positive constants.

Collision avoidance. In order to avoid collisions between the fingers, it is imposed the distance between the fingers be larger than a safety value, d_s; hence, if $d_{ii'}$ denotes the distance between the ith and the i'th finger, the following cost function can be formalized

$$\mathcal{C}_{CA}(\tilde{\boldsymbol{q}}) = \sum_{i,i'} c_{ii'}(\tilde{\boldsymbol{q}}), \tag{28}$$

where the sum is extended to all the couples of fingers,

$$c_{ii'}(d_{ii'}) = \begin{cases} k_{ii'}\dfrac{d_s - d_{ii'}}{d_{ii'}^2}, & \text{if } d_{ii'} \le d_s, \\ 0, & \text{if } d_{ii'} > d_s, \end{cases} \tag{29}$$

and $k_{ii'}$ is a positive gain.

B. Simulation results

The parameters of the elastic contact are: $5 \cdot 10^4$ N/m for the springs elastic coefficients, 5 Ns^2/ m for the springs damper coefficients and $l_i = 5 \cdot 10^{-3}$ m for the springs rest condition. The parameters used to define the subtasks are chosen as follows: $\overline{w}_i = 2.55$ for the manipulability

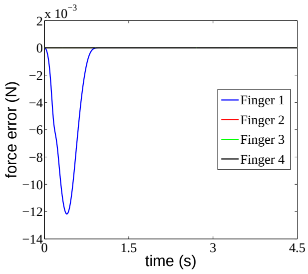

Fig. 6. Finger force errors.

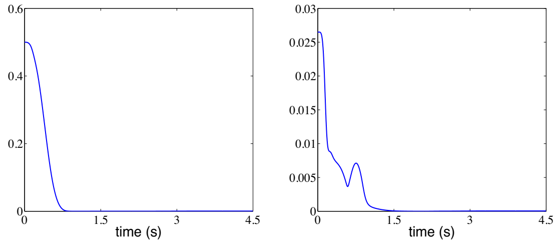

Fig. 7. Force and moment residuals.

subtask, $\underline{q}_j = -110^o, \overline{q}_j = 110^o, k_j = 5, \delta = 2$ for the joint-limit avoidance and $k_{ii'} = 1, d_s = 5$ cm for the collision avoidance. In the system of Fig. 3 the palm is represented by the ramification point of the right manipulator. The task has a duration of 4.5 s; a Runge-Kutta integration method, with a step size of 2 ms, has been used to simulate the system.

Fig. 5 shows the time history of the norm of the object's pose error for each finger (position on the left, orientation on the right). Fig. 6 shows the evolution of the force error for each finger: in detail, finger 1 is much more affected by the motion of fingers 3 and 4 than 2; the desired value for the normal force at tips 3 and 4 is very small and it is impossible to see remarkable variations in the time history.

Fig. 7 shows the force and moment residuals, ε_f and ε_m. Since both residuals asymptotically converge to zero, it is clear that fingers 3 and 4 reach a force closure condition.

Fig. 8 shows the time history of the manipulability measure (left) and the distance from the palm function σ_P in (26) (right). The manipulability measure of each finger is above the limit value $\overline{w}_i$, while, σ_P is zero when the task is not activated, since the palm is sufficiently far from the object.

Finally Fig. 9 depicts the time history of the stack status during the simulation. It can be noticed that the main task is never removed from the stack, while the other tasks are removed when some constraints are near to be violated. When the system is in a safe condition with respect to the constraints, the task are re-inserted in the stack maintaining their previous priorities. Notice that the label assigned at each task denotes its priority in the stack.

VI. Conclusion and Future Work

In this paper the kinematic model of a redundant robotic manipulation system has been considered. A two-stage control scheme has been designed to achieve the desired object motion and the desired normal contact forces. The redundancy of the whole system has been exploited to fulfil a set of prioritized constraints and secondary tasks. Simulation results show that the adopted control scheme ensures successful achievement of the main task, without violating any imposed constraint. Future work will be focused on manipulating an object with unknown shape, with consequent online estimation of the related Jacobians, and on an experimental validation of the proposed algorithm.

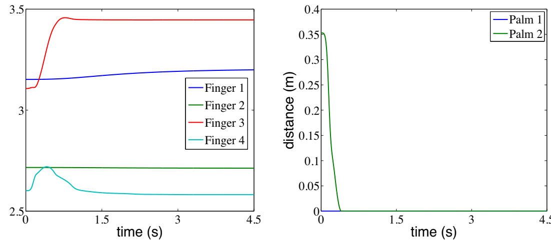

Fig. 8. Subtask functions.

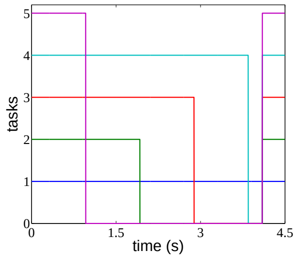

Fig. 9. Time history about the tasks in the stack status: task 1 is the main task, corresponding to keep the object fixed; 2 and 3 are the force and moment residual tasks, respectively; 4 is the manipulability task, while 5 is the task about distance between the palms and the object.

References

[1] G. Antonelli, "Stability analysis for prioritized closed-loop inverse kinematic algorithms for redundant robotic systems," *IEEE Transaction on Robotics*, vol. 25 pp. 985-994, 2009.

[2] A. Bicchi, D. Prattichizzo, "Manipulability of cooperative robots with unactuated joints and closed-chain mechanisms", *IEEE Transactions on Robotics and Automation*, vol. 16, pp. 336–345, 2000.

[3] J. Coelho, R. Grupen "A control basis for learning multifingered grasps" *Journal of Robotic Systems*, vol. 14, pp. 545-557, 1997.

[4] L. Han, J. C. Trinkle, "The instantaneous kinematics of manipulation," *Proc. IEEE International Conference on Robotics and Automation*, pp. 1944–1949, 1998.

[5] V. Lippiello, F. Ruggiero, L. Villani, "Exploiting redundancy in closed-loop inverse kinematics for dexterous object manipulation," *International Conference on Advanced Robotics*, Munich, 2009.

[6] N. Mansard, F. Chaumette "Task Sequencing for High-Level Sensor-Based Control", *IEEE Transactions on Robotics and Automation*, vol. 23, pp. 60–72, 2007.

[7] D. Montana, "The Kinematics of Contact and Grasp," *International Journal of Robotics Research*, vol. 7, no. 3, pp. 17–32, 1988.

[8] D. Montana, "The kinematics of multi-fingered manipulation", *IEEE Transactions on Robotics and Automation*, vol. 11, pp. 491–503, 1995.

[9] R.M. Murray, Z.X. Li, S.S. Sastry, *A mathematical introduction to robotic manipulation*, CRC press, Boca Raton, 1993.

[10] A. M. Okamura, N. Smaby, M.R. Cutkosky, "An Overview of Dextrous Manipulation," *Proc. IEEE International Conference on Robotics and Automation*, pp. 255–262, 2000.

[11] R. Platt, A.H. Fagg, R. Grupen "Null-Space Grasp Control: Theory and Experiments," *IEEE Transactions on Robotics*, vol. 26, pp. 282-295, 2010.

[12] J. Ponce, S. Sullivan, A. Sudsang, J. Boissonnat, J. Merlet, "On computing four-finger equilibrium and force-closure grasps of polyhedral objects," *International Journal of Robotic Research*, vol. 16, pp. 11-35, 1996.

[13] D. Prattichizzo, J.C. Trinkle, "Grasping" *Springer Handbook of Robotics*, B. Siciliano, O. Khatib (Eds.), Springer,, pp. 671–700, 2008.

[14] B. Siciliano, L. Sciavicco, L.Villani, G. Oriolo, "Robotics. Modelling, Planning and Control", *Springer*, 2009.