P. Masarati
Computed Torque Control of Redundant Manipulators Using General-Purpose software in Real-Time
Multibody System Dynamics, Vol. 32, N. 4, 2014, p. 403-428
doi:10.1007/s11044-013-9377-4

Multibody System Dynamics manuscript No.
(will be inserted by the editor)

Computed Torque Control of Redundant Manipulators Using General-Purpose Software in Real-Time

Pierangelo Masarati

Received: date / Accepted: date

Abstract This work proposes the simultaneous solution of inverse kinematics and inverse dynamics of redundant manipulators for (nearly) real-time joint trajectory design and feedforward control torque computation using general-purpose multibody formulations and software tools based on redundant coordinate approaches. The proposed scheme consists of a staggered sequence of three inverse kinematics problems that compute positions, velocities, and accelerations, followed by an inverse dynamics problem that computes feedforward generalized driving forces. The soundness of the proposed scheme is illustrated by its application to several problems of increasing complexity.

Keywords Redundant Manipulators · Inverse Kinematics · Inverse Dynamics · Feedforward Control · Redundant Coordinate Approach

1 Introduction

Robotic manipulators can be used to automate repetitive or dangerous tasks, replacing human operators. The control of manipulators involves several aspects, including trajectory design, and feedforward and feedback control. Redundancy, dexterity and failover handling are desirable features. The problem is discussed, for example, in [1, 2].

The prescription of the motion of the end effector, typically a tool or a payload, implies the need to prescribe a certain number of conditions (usually up to six for simple manipulators) on the kinematics of rigid bodies. Redundant manipulators, i.e. manipulators whose number of joints and thus of motors is higher than the number of prescribed conditions on the trajectory of the end effector, typically offer increased dexterity. This gives the user the freedom to introduce further requirements on the trajectory of the joints, for example to minimize the work or the power required to perform a given task, avoid obstacles, tolerate faults, and so on.

Politecnico di Milano, Dipartimento di Ingegneria Aerospaziale
Via La Masa 34, 20156 Milano, Italy
Tel.: +39 02 23998309
Fax: +39 02 23998334
E-mail: pierangelo.masarati@polimi.it

A different, although related problem is that of underactuated systems, whose number of motors is less than the actual number of degrees of freedom of the manipulator. An active field of research in this area is related to the so-called control constraint approach [3–5].

As soon as the desired joint trajectory is determined, feedback control is needed to ensure that the manipulator actually follows it. In addition, feedforward control may improve the accuracy in trajectory tracking. Feedforward is intrinsically model based; it can be accomplished using Computed Torque Control (CTC) (see for example [6, 7]).

The problem of the inverse dynamics of redundant manipulators has been widely discussed in the literature for decades (see for example [8]), and is still an active field of research (see for example [9–14]). Typically, locally optimal solutions are found by efficiently minimizing instantaneous cost functions. This makes the design of joint trajectories possible in real-time. Integral approaches to trajectory optimization of redundant manipulators (e.g. [15,16]) can produce better trajectories but typically cannot meet the real-time requirement.

This work discusses how inverse kinematics and inverse dynamics of fairly general redundant manipulator configurations can be performed using general-purpose multibody analysis in (nearly) real-time. A generic redundant coordinate formulation for constrained dynamics analysis is presented in section 2, along with its projection in the manifold of the passive constraints to produce a minimum coordinate set, to discuss how the inverse kinematics/inverse dynamics problem can be formulated in an intuitive manner. The inverse kinematics problem is discussed using the minimum coordinate set approach in section 2.1, and subsequently formulated directly using redundant coordinates in sections 2.1.1–2.1.3. The actual implementation of the inverse dynamics problem is discussed in detail in section 2.2. Feedforward/feedback control is discussed in section 2.3. Extensive numerical results are presented in section 3.

2 Inverse Kinematics and Dynamics Formulation

Consider a fully actuated system, i.e. a system with j joint variables $\boldsymbol{\theta}$; the motion of each joint is controlled by a motor. The problem is written according to the redundant coordinate set approach (see [17], Eqs. (33) and (34)), where $\mathbf{x}$ are the n coordinates that describe the motion of the system; usually, $n \gg j$ (e.g. $n = 6N_b$ for a spatial problem, where N_b represents the number of bodies).

No reference to a specific formulation is made, since the considerations that follow apply to a broad variety of formulations that meet the definition of redundant coordinate set, regardless of the way the corresponding equations of motion are formulated. For the records, the proposed formulation has been implemented in the free, general purpose multibody solver MBDyn (`http://www.mbdyn.org/`). Limited modifications to the solution procedure were needed.

The equations that describe the dynamics of a constrained system are

$$\mathbf{M}(\mathbf{x})\,\ddot{\mathbf{x}} + \boldsymbol{\phi}^T_{/\mathbf{x}}\boldsymbol{\lambda} = \mathbf{f}(\mathbf{x}, \dot{\mathbf{x}}, t) + \boldsymbol{\theta}^T_{/\mathbf{x}}\mathbf{c} \tag{1a}$$

$$\boldsymbol{\phi}(\mathbf{x}) = \mathbf{0}, \tag{1b}$$

where vector $\boldsymbol{\phi}$ contains the b passive constraint equations, and $\mathbf{c}$ the $j = n - b$ motor torques. Actually, the term 'generalized driving forces' should be used instead

of 'torques', as the latter term is specific of actuators associated with revolute joints; in the following, the two terms are used interchangeably.

In principle, j equations $\boldsymbol{\theta} = \boldsymbol{\theta}(\mathbf{x})$ can be written to express the joint variables as functions of the redundant coordinates of the system, such that

$$\boldsymbol{\phi}_{/\mathbf{x}} \boldsymbol{\theta}^{+}_{/\mathbf{x}} \equiv \mathbf{0}, \tag{2}$$

where $(\cdot)^{+}$ denotes the Moore-Penrose Generalized Inverse (MPGI) [18]. $\boldsymbol{\theta}^{+}_{/\mathbf{x}}$ is the MPGI of the Jacobian matrix of the relationship between the joint variables and the redundant coordinates of the system, since by definition they represent the Lagrangian coordinates of the mechanism, and thus the motion they describe intrinsically lies in the manifold of the passive constraints, $\boldsymbol{\phi}$. Since $j = n - b$, the problem is fully actuated, as already mentioned. This is typical of industrial robots, but it is also the case of biodynamics, for example.

In principle, the problem can be rewritten as a function of the joint coordinates by considering the first derivative of the passive constraints of Eq. (1b), namely

$$\dot{\boldsymbol{\phi}} = \boldsymbol{\phi}_{/\mathbf{x}} \dot{\mathbf{x}} = \mathbf{0}, \tag{3}$$

by replacing $\dot{\mathbf{x}}$ and $\ddot{\mathbf{x}}$ with their expressions as functions of the joint coordinates,

$$\dot{\mathbf{x}} = \boldsymbol{\theta}^{+}_{/\mathbf{x}} \dot{\boldsymbol{\theta}} \tag{4a}$$

$$\ddot{\mathbf{x}} = \boldsymbol{\theta}^{+}_{/\mathbf{x}} \ddot{\boldsymbol{\theta}} - \boldsymbol{\theta}^{+}_{/\mathbf{x}} \left(\dot{\boldsymbol{\theta}}\right)_{/\mathbf{x}} \boldsymbol{\theta}^{+}_{/\mathbf{x}} \dot{\boldsymbol{\theta}}, \tag{4b}$$

which, thanks to Eq. (2), guarantees that Eq. (1b) and its derivatives are intrinsically satisfied, and by projecting the equation of motion (1a) in the space of the joint coordinates, i.e.

$$\underbrace{\left(\boldsymbol{\theta}^{+}_{/\mathbf{x}}\right)^{T} \mathbf{M} \boldsymbol{\theta}^{+}_{/\mathbf{x}}}_{\hat{\mathbf{M}}} \ddot{\boldsymbol{\theta}} + \underbrace{\left(\boldsymbol{\theta}^{+}_{/\mathbf{x}}\right)^{T} \boldsymbol{\phi}^{T}_{/\mathbf{x}}}_{\mathbf{0},\ \text{Eq. (2)}} \boldsymbol{\lambda} = \underbrace{\left(\boldsymbol{\theta}^{+}_{/\mathbf{x}}\right)^{T} \left(\mathbf{M} \boldsymbol{\theta}^{+}_{/\mathbf{x}} \left(\dot{\boldsymbol{\theta}}\right)_{/\mathbf{x}} \boldsymbol{\theta}^{+}_{/\mathbf{x}} \dot{\boldsymbol{\theta}} + \mathbf{f}\right)}_{\hat{\mathbf{f}}} + \underbrace{\left(\boldsymbol{\theta}^{+}_{/\mathbf{x}}\right)^{T} \boldsymbol{\theta}^{T}_{/\mathbf{x}}}_{\mathbf{I}} \mathbf{c}, \tag{5}$$

where $\hat{\mathbf{M}}$ is the projection of the mass matrix in the space of the joint coordinates. The approach of Eq. (5) is at the roots of minimum coordinate set approaches based on coordinate projection. In [19], a review of techniques to compute matrix $\boldsymbol{\theta}^{+}_{/\mathbf{x}}$ is presented, including coordinate partitioning, zero eigenvalue theorem, and singular-value, QR, pseudo-upper triangular, and Schur decompositions.

However, such projection may be impractical when one wants to exploit readily available general-purpose formulations based on the redundant coordinate set approach, which provide a straightforward manner to solve the inverse kinematics and dynamics problems directly in the redundant coordinates $\mathbf{x}$, although providing a consistent result that can be interpreted in terms of the joint coordinates $\boldsymbol{\theta}$. The latter approach is pursued in the present work.

2.1 Inverse Kinematics

Consider now a set of control constraints, i.e. a set of c constraint equations

$$\boldsymbol{\psi}\left(\mathbf{x}\right) = \boldsymbol{\alpha}(t) \tag{6}$$

that prescribe the motion of a part of the system, subjected to $\boldsymbol{\phi}_{/\mathbf{x}}\boldsymbol{\psi}^{+}_{/\mathbf{x}} \equiv \mathbf{0}$. When $c = j$, and $\boldsymbol{\psi}_{/\mathbf{x}}$ is full row rank, the problem is fully determined and its inverse kinematics can be computed in a straightforward manner, as discussed (and implemented) for example in [9], by solving Eqs. (6) and (1b) and their time derivatives up to second-order, provided $\boldsymbol{\phi}_{/\mathbf{x}}$ is full row rank as well.

However, if $c < j$ the problem is underdetermined. A solution is needed in the joint coordinate space $\boldsymbol{\theta}$ that yields the desired end effector motion $\boldsymbol{\alpha}(t)$, i.e., at the velocity level,

$$\underbrace{\boldsymbol{\psi}_{/\mathbf{x}}\boldsymbol{\theta}^{+}_{/\mathbf{x}}}_{\mathbf{J}}\dot{\boldsymbol{\theta}} = \dot{\boldsymbol{\alpha}}, \tag{7}$$

which implies a solution of the form

$$\dot{\boldsymbol{\theta}} = \mathbf{J}^{+}\dot{\boldsymbol{\alpha}} + \left(\mathbf{I} - \mathbf{J}^{+}\mathbf{J}\right)\boldsymbol{\omega}, \tag{8}$$

where the portion of an arbitrary set of joint angular velocities $\boldsymbol{\omega}$ that lies in the nullspace of the Jacobian matrix $\mathbf{J}$ can be added to those resulting from a least-squares solution of Eq. (7), and optionally used to meet some local optimality condition. The Jacobian matrix $\mathbf{J}$ can be weighted[1] by a symmetric, positive definite matrix $\mathbf{W}$ (usually diagonal) before inversion:

$$\dot{\boldsymbol{\theta}} = \mathbf{W}\left(\mathbf{J}\mathbf{W}\right)^{+}\dot{\boldsymbol{\alpha}} = \mathbf{J}^{+}_{\mathbf{W}}\dot{\boldsymbol{\alpha}}; \tag{9}$$

$\mathbf{W} = \hat{\mathbf{M}}^{-1/2}$, the inverse of the square root of the mass matrix in the space of the joint coordinates, is often used (see [20] for a discussion about weighting with the square root of the mass matrix). The resulting motion is then used to compute the j torques $\mathbf{c}$ by solving the corresponding inverse dynamics problem.

The possibility to exploit $\boldsymbol{\omega}$, i.e. the redundancy of the system, to improve the quality of robots' actuation has been a research topic for more than 30 years, as discussed for example by Hollerbach and Suh, [21]. Suggested criteria include the minimization of position-dependent scalar performance indexes, e.g. joint limits avoidance (Liégeois, [22]), dexterity measures like kinematic and dynamic manipulability indexes (e.g. Yoshikawa [23], although that measure has been generalized and at the same time criticized by Doty et al., [24]), or the norm of the joint torques themselves, as discussed by Hollerbach and Suh in [21] and subsequent works. The latter condition requires to

[1] $\mathbf{J}^{+}_{\mathbf{W}}$ in Eq. (9) is readily obtained by pseudo-inverting $\mathbf{J}\dot{\boldsymbol{\theta}} = (\mathbf{J}\mathbf{W})\mathbf{W}^{-1}\dot{\boldsymbol{\theta}} = \dot{\boldsymbol{\alpha}}$. It corresponds to solving

$$\begin{bmatrix} \mathbf{W}^{-2} & \mathbf{J}^{T} \\ \mathbf{J} & \mathbf{0} \end{bmatrix} \begin{Bmatrix} \dot{\boldsymbol{\theta}} \\ \boldsymbol{\lambda} \end{Bmatrix} = \begin{Bmatrix} \mathbf{0} \\ \dot{\boldsymbol{\alpha}} \end{Bmatrix},$$

disregarding $\boldsymbol{\lambda}$, which minimizes the ℓ^2-norm of $\dot{\boldsymbol{\theta}}$ subjected to $\mathbf{J}\dot{\boldsymbol{\theta}} = \dot{\boldsymbol{\alpha}}$, weighted by $\mathbf{W}^{-2}$.

express Eq. (8) at the accelerations level, in order to estimate the torques as functions of the motion, leading to what has been termed *unweighted* or *inertia-weighted null-space algorithm*: the (weighted) joint accelerations are

$$\ddot{\boldsymbol{\theta}} = \mathbf{J}_{\mathbf{W}}^{+} \left(\ddot{\boldsymbol{\alpha}} - \dot{\mathbf{J}}_{\mathbf{W}} \dot{\boldsymbol{\theta}} \right). \tag{10}$$

The corresponding joint torques thus are

$$\mathbf{c} = \hat{\mathbf{M}} \ddot{\boldsymbol{\theta}} + \dot{\hat{\mathbf{M}}} \dot{\boldsymbol{\theta}} + \hat{\mathbf{f}} = \hat{\mathbf{M}} \mathbf{J}_{\mathbf{W}}^{+} \ddot{\boldsymbol{\alpha}} + \left(\dot{\hat{\mathbf{M}}} - \hat{\mathbf{M}} \mathbf{J}_{\mathbf{W}}^{+} \dot{\mathbf{J}}_{\mathbf{W}} \right) \dot{\boldsymbol{\theta}} + \hat{\mathbf{f}}. \tag{11}$$

In [21] the authors suggest to minimize the norm of the difference between the torques $\mathbf{c}$ and the average of the minimum and maximum value, $(\mathbf{c}_{\max} + \mathbf{c}_{\min})/2$, often zero.

As noted by Suh and Hollerbach in [25], an approach like this may have undesirable effects; for example, local minimization of the torques does not prevent the achievement of singular operating conditions, and the manipulator does not necessarily come to a rest position when the end-effector stops.

In general, several criteria can be formulated to eliminate the local arbitrariness associated with redundant actuation, keeping in mind that local optima may not necessarily lead to globally optimum behavior, although local criteria are required, for example, to achieve the capability to design the joints' trajectory in real-time.

In this work, the procedure of Eq. (5) is not directly implemented; on the contrary, a straightforward manner to obtain the same result using an approach based on a redundant coordinate set formulation is proposed, with the objective of avoiding the need to explicitly compute $\mathbf{J}$, but rather resorting to existing, well established general-purpose implementations. An important aspect of this approach is that it can be applied to arbitrary configurations, including open and closed-loop kinematic chains, and thus parallel manipulators [26].

In order to guarantee *a priori* the compliance with the constraints, the configuration is predicted first, followed by the computation of the corresponding velocities and accelerations resulting from the numerical differentiation of the predicted configuration, corrected to comply with the derivatives of the constraints.

2.1.1 Position

Following the approach proposed in [27], the inverse kinematics problem at the position level is recast into the computation of a weighted minimal norm solution of the kineto-static problem

$$\begin{aligned} &\text{minimize} && J'(\mathbf{x}) \\ &\text{subjected to} && \boldsymbol{\phi}(\mathbf{x}) = \mathbf{0} \\ & && \boldsymbol{\psi}(\mathbf{x}) = \boldsymbol{\alpha}(t) \end{aligned} \tag{12}$$

where the 'prime' $(\cdot)'$ denotes entities associated with the position problem, which results in

$$\boldsymbol{\phi}_{/\mathbf{x}}^{T} \boldsymbol{\lambda}' + \boldsymbol{\psi}_{/\mathbf{x}}^{T} \boldsymbol{\mu}' = \mathbf{f}'(\mathbf{x}) \tag{13a}$$

$$\boldsymbol{\phi}(\mathbf{x}) = \mathbf{0} \tag{13b}$$

$$\boldsymbol{\psi}(\mathbf{x}) = \boldsymbol{\alpha}(t), \tag{13c}$$

with $\mathbf{f}' = -J'_{/\mathbf{x}}$. Here J' can be constructed as the combination of an 'ergonomy' function, corresponding to some potential energy-like function with respect to a measure of the relative motion between the parts of the system, and of a 'proximity' function that weights the change of configuration with respect to that at the previous time step, i.e.

$$J'(\mathbf{x}) = \hat{J}'(\boldsymbol{\theta}) + w' \frac{1}{2} (\mathbf{x} - \mathbf{x}_{\text{prev}})^T \mathbf{M} (\mathbf{x} - \mathbf{x}_{\text{prev}}) . \tag{14}$$

The use of the mass matrix $\mathbf{M}$, scaled by the coefficient w', intuitively makes it possible to weigh proximity contributions with respect to the inertia associated with changing the related configuration.

Other cost functions can be envisaged. For example, along the lines of the null-space algorithm illustrated in [21], a quadratic form like

$$J'_{\mathbf{c}} = \frac{1}{2} (\mathbf{c} - \mathbf{c}_{\text{ave}})^T \hat{\mathbf{M}}^{-1} (\mathbf{c} - \mathbf{c}_{\text{ave}}) \tag{15}$$

can be considered, where $\mathbf{c}$ are defined in Eq. (11) whereas $\mathbf{c}_{\text{ave}}$ is the average of the minimum and maximum torque that each motor can produce, to obtain a weighted least-squares solution that minimizes the norm of the torques instantaneously required to perform the task. Since at this stage the problem is only being solved at the positions level, an estimate of the current acceleration $\ddot{\mathbf{x}} = \ddot{\mathbf{x}}(\mathbf{x})$ based on the configuration and its derivatives at the previous steps and on the current configuration is needed. This approach has been formulated and investigated but, in view of the limitations highlighted by the authors themselves in [25], it has been discarded, and thus it is no longer mentioned in the present work.

The gradient

$$\mathbf{f}' = -J_{/\mathbf{x}} = -\boldsymbol{\theta}^T_{/\mathbf{x}} \hat{J}'_{/\boldsymbol{\theta}} - w' \mathbf{M} (\mathbf{x} - \mathbf{x}_{\text{prev}}) \tag{16}$$

can be interpreted as a set of elastic torques. A quadratic form of $\hat{J}'$,

$$\hat{J}' = \frac{1}{2} (\boldsymbol{\theta} - \boldsymbol{\theta}_0)^T \mathcal{K} (\boldsymbol{\theta} - \boldsymbol{\theta}_0) , \tag{17}$$

yields linear elastic torques; higher-order forms

$$\hat{J}' = \sum_{i=1,j} \left(\frac{1}{2} \mathcal{K}_{ii} (\theta_i - \theta_{i0})^2 + \frac{1}{l} \mathcal{H}_{ii} (\theta_i - \theta_{i0})^l \right) , \tag{18}$$

with $l \in 2 + 2\mathbb{N}$ (i.e. $[4, 6, \ldots]$), increasingly penalize the departure from the centered configuration ($l = 4$, with $\mathcal{H}_{ii}/l = 16 w_i / (\theta_{i_{\max}} - \theta_{i_{\min}})^4$ and $\theta_{i0} = (\theta_{i_{\max}} + \theta_{i_{\min}})/2$, was proposed in [28] to estimate ergonomy solutions of the motion of a human arm); however, the local penalty matrix that corresponds to the Hessian matrix of $\hat{J}'$ could be singular or poorly conditioned when any of the dummy springs is close to the optimal centered position (i.e. when $\theta_i \cong \theta_{i0}$), unless $\mathcal{K}_{ii} > 0$.

In practice, when $w' \equiv 0$ this phase can be easily implemented in general-purpose multibody solvers using generic elastic elements of arbitrary complexity, related to the joint coordinates, while solving a static problem. The availability of special elastic elements with appropriate constitutive properties can be exploited (see for example [29, 30]), as done in the present work. The multipliers $\boldsymbol{\lambda}'$ and $\boldsymbol{\mu}'$ do not have any physical meaning and their value is discarded; the same occurs in the subsequent kinematics solution phases.

The problem of Eqs. (13) can be solved resorting to a Newton-like iterative procedure as long as $\boldsymbol{\phi}_{/\mathbf{x}}$ and $\boldsymbol{\psi}_{/\mathbf{x}}$ are full row rank. Rank-deficient Jacobian matrices indicate a singular condition, which could be overcome resorting to pseudo-inversion. Since the redundant coordinate set approach is used, singular Jacobian matrices imply an actually singular configuration, rather than a singularity related to the choice of the Lagrangian coordinates. To some extent, such configurations can be avoided by appropriately crafting the ergonomy functions.

The problem is nonlinear as long as the constraint equations and the dummy elastic torques are nonlinear. In order to exemplify the process, let us consider a(n oversimplified) linear problem, with

$$\left\{ \begin{array}{c} \boldsymbol{\phi} \\ \boldsymbol{\psi} \end{array} \right\} = \mathbf{A}\mathbf{x} \qquad\qquad \left\{ \begin{array}{c} \mathbf{0} \\ \boldsymbol{\alpha}(t) \end{array} \right\} = \boldsymbol{\alpha}^*(t) \tag{19}$$

and

$$\mathbf{f}' = -\mathbf{K}(\mathbf{x} - \mathbf{x}_0) + \mathbf{f}'_0. \tag{20}$$

The solution would be

$$\begin{aligned} \mathbf{x} = {} & \mathbf{K}^{-1}\mathbf{A}^T \left(\mathbf{A}\mathbf{K}^{-1}\mathbf{A}^T \right)^{-1} \boldsymbol{\alpha}^*(t) \\ & + \left(\mathbf{I} - \mathbf{K}^{-1}\mathbf{A}^T \left(\mathbf{A}\mathbf{K}^{-1}\mathbf{A}^T \right)^{-1} \mathbf{A} \right) \left(\mathbf{x}_0 + \mathbf{K}^{-1}\mathbf{f}'_0 \right), \end{aligned} \tag{21}$$

which, for $\mathbf{f}'_0 = \mathbf{0}$, $\mathbf{x}_0 = \mathbf{0}$, corresponds to a weighted least squares solution and, with $\mathbf{K} = k\mathbf{I}$, to $\mathbf{x} = \mathbf{A}^+\boldsymbol{\alpha}^*(t)$, to the minimal norm solution. In practice, $\mathbf{K}$ is ideally associated with springs, i.e. elastic elements that react relative motion, thus a minimum of the related strain energy minimizes the difference between the current and the reference (e.g. maximum ergonomy) configuration. The external force $\mathbf{f}'_0$ can be used to further influence the optimal solution, e.g. by adding gravity.

2.1.2 Velocity

Subsequently, a similar problem must be solved for velocities and accelerations. Those problems are simpler, since the derivatives of the constraint equations are linear respectively in the velocities and accelerations. In the velocity case, the problem is

$$\begin{array}{ll} \text{minimize} & J''(\dot{\mathbf{x}}) \\ \text{subjected to} & \boldsymbol{\phi}_{/\mathbf{x}}\dot{\mathbf{x}} = \mathbf{0} \\ & \boldsymbol{\psi}_{/\mathbf{x}}\dot{\mathbf{x}} = \dot{\boldsymbol{\alpha}}(t) \end{array} \tag{22}$$

which results in

$$\boldsymbol{\phi}^T_{/\mathbf{x}}\boldsymbol{\lambda}'' + \boldsymbol{\psi}^T_{/\mathbf{x}}\boldsymbol{\mu}'' = \mathbf{f}''(\mathbf{x}) \tag{23a}$$

$$\boldsymbol{\phi}_{/\mathbf{x}}\dot{\mathbf{x}} = \mathbf{0} \tag{23b}$$

$$\boldsymbol{\psi}_{/\mathbf{x}}\dot{\mathbf{x}} = \dot{\boldsymbol{\alpha}}(t) \tag{23c}$$

with $\mathbf{f}'' = -J''_{/\mathbf{x}}$. The norm of the error between $\dot{\mathbf{x}}$ and its estimate obtained by numerical differentiation of the previously computed configuration $\mathbf{x}$, weighted by the mass matrix, is used, namely

$$J''(\dot{\mathbf{x}}) = \frac{1}{2}\left(\dot{\mathbf{x}} - \dot{\hat{\mathbf{x}}}\right)^T \mathbf{M}\left(\dot{\mathbf{x}} - \dot{\hat{\mathbf{x}}}\right), \tag{24}$$

where a generic linear interpolation operator $\mathcal{L}$, function of the time step h, is used to estimate

$$\dot{\hat{\mathbf{x}}} = \mathcal{L}(\mathbf{x}, h). \tag{25}$$

In order to be able to run the process in a time marching fashion, e.g. to evaluate the trajectory in real-time, a forward operator is needed. The simplest one is single-step; the first-order accurate derivative estimate at step k is

$$\dot{\hat{\mathbf{x}}}_k = \frac{1 + \|\rho_\infty\|}{h}(\mathbf{x}_k - \mathbf{x}_{k-1}) - \|\rho_\infty\|\,\dot{\mathbf{x}}_{k-1}; \tag{26}$$

the asymptotic spectral radius ρ_∞ must satisfy the condition $0 \leq \|\rho_\infty\| < 1$ for algorithmic stability. In practice, implicit Euler (i.e. $\|\rho_\infty\| = 0$) is often needed to avoid spurious oscillations in the torques estimated in the subsequent inverse dynamics phase.

The linear problem

$$\begin{bmatrix} \mathbf{M} & \boldsymbol{\phi}^T_{/\mathbf{x}} & \boldsymbol{\psi}^T_{/\mathbf{x}} \\ \boldsymbol{\phi}_{/\mathbf{x}} & \mathbf{0} & \mathbf{0} \\ \boldsymbol{\psi}_{/\mathbf{x}} & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{Bmatrix} \dot{\mathbf{x}} \\ \boldsymbol{\lambda}'' \\ \boldsymbol{\mu}'' \end{Bmatrix} = \begin{Bmatrix} \mathbf{M}\dot{\hat{\mathbf{x}}} \\ \mathbf{0} \\ \dot{\boldsymbol{\alpha}}(t) \end{Bmatrix} \tag{27}$$

needs to be solved. As shown in [31], weighting the norm of the velocity correction with the mass matrix minimizes (and guarantees to reduce, in case of scleronomic constraints) the correction of the kinetic energy.

2.1.3 Acceleration

A similar problem arises for the accelerations,

$$\begin{aligned} &\text{minimize} && J'''(\ddot{\mathbf{x}}) \\ &\text{subjected to} && \boldsymbol{\phi}_{/\mathbf{x}}\ddot{\mathbf{x}} = -\left(\boldsymbol{\phi}_{/\mathbf{x}}\dot{\mathbf{x}}\right)_{/\mathbf{x}}\dot{\mathbf{x}} \\ & && \boldsymbol{\psi}_{/\mathbf{x}}\ddot{\mathbf{x}} = \ddot{\boldsymbol{\alpha}}(t) - \left(\boldsymbol{\psi}_{/\mathbf{x}}\dot{\mathbf{x}}\right)_{/\mathbf{x}}\dot{\mathbf{x}} \end{aligned} \tag{28}$$

where a quadratic form analogous to that of Eq. (24),

$$J'''(\ddot{\mathbf{x}}) = \frac{1}{2}\left(\ddot{\mathbf{x}} - \ddot{\hat{\mathbf{x}}}\right)^T \mathbf{M}\left(\ddot{\mathbf{x}} - \ddot{\hat{\mathbf{x}}}\right), \tag{29}$$

is used, which resembles the function $G(\ddot{\mathbf{x}})$ that is minimized in Gauss' principle of least constraint [32], where $\ddot{\hat{\mathbf{x}}}$ plays the role of the acceleration of the unconstrained system, $\mathbf{M}^{-1}\mathbf{f}$. The reference acceleration $\ddot{\hat{\mathbf{x}}}$ can be estimated with the same operator, Eq. (25), used for the velocity. The linear problem

$$\begin{bmatrix} \mathbf{M} & \boldsymbol{\phi}^T_{/\mathbf{x}} & \boldsymbol{\psi}^T_{/\mathbf{x}} \\ \boldsymbol{\phi}_{/\mathbf{x}} & \mathbf{0} & \mathbf{0} \\ \boldsymbol{\psi}_{/\mathbf{x}} & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{Bmatrix} \ddot{\mathbf{x}} \\ \boldsymbol{\lambda}''' \\ \boldsymbol{\mu}''' \end{Bmatrix} = \begin{Bmatrix} \mathbf{M}\ddot{\hat{\mathbf{x}}} \\ -\left(\boldsymbol{\phi}_{/\mathbf{x}}\dot{\mathbf{x}}\right)_{/\mathbf{x}}\dot{\mathbf{x}} \\ \ddot{\boldsymbol{\alpha}}(t) - \left(\boldsymbol{\psi}_{/\mathbf{x}}\dot{\mathbf{x}}\right)_{/\mathbf{x}}\dot{\mathbf{x}} \end{Bmatrix} \tag{30}$$

needs to be solved, which uses the same matrix of the velocity solution, Eq. (27).

2.2 Inverse Dynamics

Under the previously stated assumptions, the problem is now fully determined, since all joints are equipped with a motor. The problem of Eqs. (1) formally becomes

$$\mathbf{M}(\mathbf{x})\,\ddot{\mathbf{x}} + \boldsymbol{\phi}_{/\mathbf{x}}^T \boldsymbol{\lambda} = \mathbf{f} + \boldsymbol{\theta}_{/\mathbf{x}}^T \mathbf{c} \tag{31a}$$

$$\boldsymbol{\phi}(\mathbf{x}) = \mathbf{0} \tag{31b}$$

$$\boldsymbol{\theta}(\mathbf{x}) = \boldsymbol{\theta}(\mathbf{x}) \;\; \text{[yes, an identity]}, \tag{31c}$$

where the previously computed motion $\mathbf{x}$ (i.e. $\mathbf{x}$ and its time derivatives as resulting from inverse kinematics) is prescribed, and the corresponding torques $\mathbf{c}$ need to be computed, resulting in the control forces $\mathbf{u} = \boldsymbol{\theta}_{/\mathbf{x}}^T \mathbf{c}$. The motion is now fully determined by the previous inverse kinematics solution, and the passive constraints of Eq. (31b), as well as the closure constraints of Eq. (31c), are intrinsically satisfied; as a consequence, the problem of Eqs. (31) reduces to Eq. (31a) only, which is reformulated as

$$\left[\boldsymbol{\phi}_{/\mathbf{x}}^T \; -\boldsymbol{\theta}_{/\mathbf{x}}^T \right] \left\{ \begin{array}{c} \boldsymbol{\lambda} \\ \mathbf{c} \end{array} \right\} = \mathbf{f} - \mathbf{M}(\mathbf{x})\,\ddot{\mathbf{x}}. \tag{32}$$

It is worth noticing that Eq. (31c) in practice could be actually written as

$$\boldsymbol{\theta}(\mathbf{x}) + \mathbf{W}\boldsymbol{\phi}(\mathbf{x}) = \hat{\boldsymbol{\theta}}(\mathbf{x}), \tag{33}$$

where $\mathbf{W}$ is an arbitrary $j \times b$ weight matrix. For example, this is the case of the free, general purpose multibody solver MBDyn, which provides the versatile *total* joint element discussed in [33]. That formulation was successfully exploited to implement the proposed formulation as well as inverse dynamics for the computation of muscular activation [34] and the control-constraint formulation proposed in [4].

In this work, during the inverse dynamics phase each motor is modeled using a special constraint derived from the previously mentioned total joint [33]. Such constraint provides a contribution to the matrix in Eq. (32) that corresponds to $\boldsymbol{\theta}_{/\mathbf{x}}$, but actually is in the form of $\hat{\boldsymbol{\theta}}_{/\mathbf{x}}$, for each pair that includes a motor, regardless of its type, which can be an arbitrary combination of linear and angular motions.

The solution of the problem of Eq. (32), with $\boldsymbol{\theta}_{/\mathbf{x}}$ replaced by $\hat{\boldsymbol{\theta}}_{/\mathbf{x}}$, requires matrix

$$\left[\boldsymbol{\phi}_{/\mathbf{x}}^T \; -\hat{\boldsymbol{\theta}}_{/\mathbf{x}}^T \right]^{-1} = \left[\begin{array}{c} \left(\boldsymbol{\phi}_{/\mathbf{x}}^+\right)^T - \mathbf{W}^T \left(\boldsymbol{\theta}_{/\mathbf{x}}^+\right)^T \\ -\left(\boldsymbol{\theta}_{/\mathbf{x}}^+\right)^T \end{array} \right]. \tag{34}$$

As a consequence, the torques $\mathbf{c}$ are formally computed as

$$\mathbf{c} = -\left(\boldsymbol{\theta}_{/\mathbf{x}}^+\right)^T \left(\mathbf{f} - \mathbf{M}(\mathbf{x})\,\ddot{\mathbf{x}}\right), \tag{35}$$

i.e. the torques do not depend on matrix $\mathbf{W}$, and thus $\boldsymbol{\theta}_{/\mathbf{x}}$ does not need to be computed, as $\hat{\boldsymbol{\theta}}_{/\mathbf{x}}$ suffices.

If the exact expression of $\boldsymbol{\theta}_{/\mathbf{x}}$ for any reason is needed, such that it is exactly orthogonal to $\boldsymbol{\phi}_{/\mathbf{x}}$, a value in the form of the Jacobian matrix of Eq. (33) can be modified by projecting it as

$$\boldsymbol{\theta}_{/\mathbf{x}}^T = \left(\mathbf{I} - \boldsymbol{\phi}_{/\mathbf{x}}^T \left(\boldsymbol{\phi}_{/\mathbf{x}} \boldsymbol{\phi}_{/\mathbf{x}}^T \right)^{-1} \boldsymbol{\phi}_{/\mathbf{x}} \right) \hat{\boldsymbol{\theta}}_{/\mathbf{x}}^T = \left(\mathbf{I} - \boldsymbol{\phi}_{/\mathbf{x}}^+ \boldsymbol{\phi}_{/\mathbf{x}} \right) \hat{\boldsymbol{\theta}}_{/\mathbf{x}}^T, \tag{36}$$

since by construction $\mathbf{W} = \hat{\boldsymbol{\theta}}_{/\mathbf{x}} \boldsymbol{\phi}_{/\mathbf{x}}^T (\boldsymbol{\phi}_{/\mathbf{x}} \boldsymbol{\phi}_{/\mathbf{x}}^T)^{-1} = \hat{\boldsymbol{\theta}}_{/\mathbf{x}} \boldsymbol{\phi}_{/\mathbf{x}}^+$.

This is the case, for example, when the contribution of redundant motors to the torque exerted about one joint needs to be estimated; a typical occurrence is in biomechanics, where multiple antagonistic and synergetic muscles contribute to the torque about each articulation. In those cases, $(\boldsymbol{\theta}_{/\mathbf{x}}^+)^T = \boldsymbol{\theta}_{/\mathbf{x}} (\boldsymbol{\theta}_{/\mathbf{x}} \boldsymbol{\theta}_{/\mathbf{x}}^T)^{-1}$ is needed to project the muscles' forces and moments, expressed in the space of the redundant coordinates, $\mathbf{x}$, exactly on the joint coordinates. However, this discussion is outside the scope of the present work.

2.3 Feedforward and Feedback Control

Formally, the projected equation of motion, Eq. (5), yields the feedforward control torques

$$\mathbf{c}_{\text{f.f.}} = \hat{\mathbf{M}} \ddot{\hat{\boldsymbol{\theta}}} - \hat{\mathbf{f}} \tag{37}$$

that constitute the computed torque control (CTC) when $\ddot{\hat{\boldsymbol{\theta}}}$ is equal to the desired joint acceleration $\ddot{\boldsymbol{\theta}}_d$. By setting the joint acceleration $\ddot{\hat{\boldsymbol{\theta}}}$ that drives the torque demand to

$$\ddot{\hat{\boldsymbol{\theta}}} = \ddot{\boldsymbol{\theta}}_d + \mathbf{K}_D \left(\dot{\boldsymbol{\theta}}_d - \dot{\boldsymbol{\theta}} \right) + \mathbf{K}_P \left(\boldsymbol{\theta}_d - \boldsymbol{\theta} \right), \tag{38}$$

where the desired joint angles $\boldsymbol{\theta}_d$ and their derivatives result from the solution of the cascaded inverse kinematics problems, the actuated forward dynamics problem becomes

$$\hat{\mathbf{M}} \left(\left(\ddot{\boldsymbol{\theta}}_d - \ddot{\boldsymbol{\theta}} \right) + \mathbf{K}_D \left(\dot{\boldsymbol{\theta}}_d - \dot{\boldsymbol{\theta}} \right) + \mathbf{K}_P \left(\boldsymbol{\theta}_d - \boldsymbol{\theta} \right) \right) = \mathbf{0}. \tag{39}$$

Under the assumption that $\hat{\mathbf{M}} > 0$ (positive-definite), which is true as long as $\mathbf{M} > 0$ and $\boldsymbol{\theta}_{/\mathbf{x}}^+$ is full column rank, i.e. the configuration is not singular, and by defining appropriate values of $\mathbf{K}_D$ and $\mathbf{K}_P$ (symmetric, positive definite), the dynamics of the tracking error in the joint space is described by a linear, time invariant homogeneous system. The choice of $\mathbf{K}_D$ and $\mathbf{K}_P$ gives some freedom in tuning the dynamics of the error, e.g. the peak time and the overshooting, although special attention needs to be paid to prevent problems like spillover, essentially related to modeling errors and unmodeled dynamics, especially those resulting from compliance of the joints and flexibility of the arms.

It is worth noticing that Eq. (5) needs not be written explicitly; as soon as measures of the joint coordinates and their derivatives are available during the forward analysis (e.g. in form of measurements or run-time processing of the analysis), the feedback torque correction is

$$\mathbf{c}_{\text{f.b.}} = \hat{\mathbf{M}} \left(\mathbf{K}_D \left(\dot{\boldsymbol{\theta}}_d - \dot{\boldsymbol{\theta}} \right) + \mathbf{K}_P \left(\boldsymbol{\theta}_d - \boldsymbol{\theta} \right) \right). \tag{40}$$

Given their feedback nature, when the projected mass matrix of the system is diagonally dominant, as it is the case in usual applications, an estimate of $\hat{\mathbf{M}}$ suffices, although no formal proof of stability can be given. In practice, $\hat{\mathbf{M}}$ is often approximated by a diagonal matrix, and diagonal $\mathbf{K}_D$ and $\mathbf{K}_P$ are chosen as well, thus making the feedback torque corrections co-located.

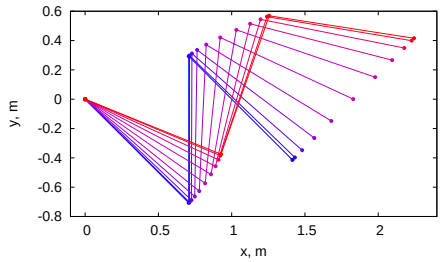

Fig. 1 Three-link arm motion resolved using linear ergonomy functions (every 20 steps, about 0.182 s).

3 Numerical Results

The proposed formulation has been implemented in the previously mentioned free, general-purpose multibody solver MBDyn. Three examples, consisting of a three-link arm, a PA10-like robot and a bio-inspired robot are presented.

3.1 Three-link Arm

The first application is a simple three-link arm, which is essentially used to exemplify the effects of different weighting on the quality of the on-line kinematics and feedforward control computation. This problem has been widely used in robotics to demonstrate redundancy-related features.

The data and one of the test cases proposed by Hollerbach and Suh in [21] are used to validate the proposed approach. Each link of the arm is 1 m long; the mass of each link is 10 kg, and is uniformly distributed. In the initial configuration, the joint angles are $\theta_1 = -45$ deg, $\theta_2 = 135$ deg, and $\theta_3 = -135$ deg.

The prescribed trajectory of the free end of link #3 is a constant acceleration/constant deceleration straight path along a straight line $\Delta y = \Delta x$. In the case presented here the total length of the path is 0.83 m in each direction. Figure 1 shows a sketch of the motion of the three-link arm where the redundancy is resolved using the proposed approach with linear ergonomy functions applied to the rotation of each joint. The ergonomy torques are zero in the initial configuration. Figure 2 compare the torques resulting from the inverse dynamics analysis with those computed in [21] using the "unweighted pseudoinverse" approach, formally analogous to the present one. The line labeled 'present, ergo' eliminates the redundancy using identical ergonomy springs, while the line labeled 'present, proxy' uses the proximity function weighted by the mass matrix as in Eq. (14). Figure 3 shows the corresponding joint angles.

3.2 Feedforward Control of a PA10-like Robot

The second application consists in a Mitsubishi PA10-like robot, a 7 joint machine of general use in the industry and in robotics research. A sketch of the robot is presented in Figure 4. Table 1 contains the Denavit-Hartenberg parameters that define the geometry

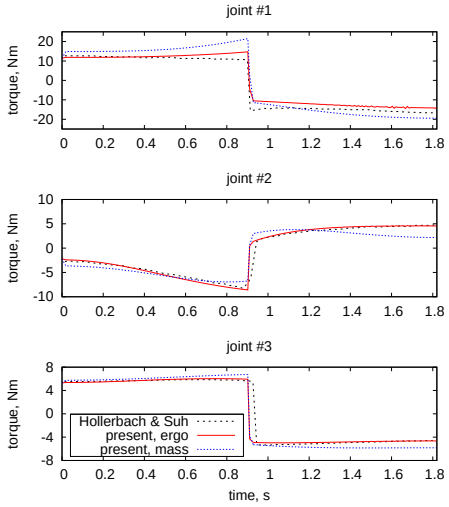

Fig. 2 Three-link arm torques.

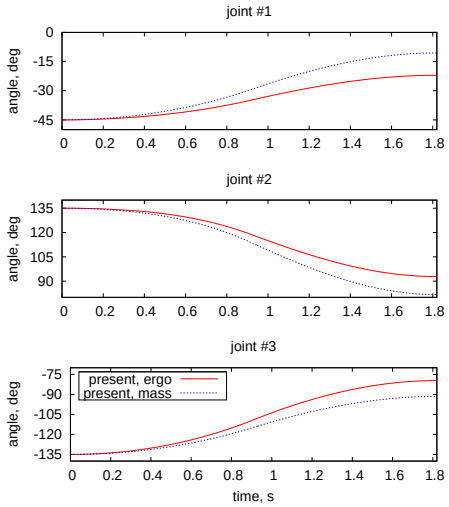

Fig. 3 Three-link arm joint rotations.

of the PA10-like robot. The value of θ_0 refers to each joint angle's initial value. Table 2 contains the simplified inertia properties of each body, starting from #2. The inertia tensor about the center of mass has been neglected, since no reliable data could be gathered.

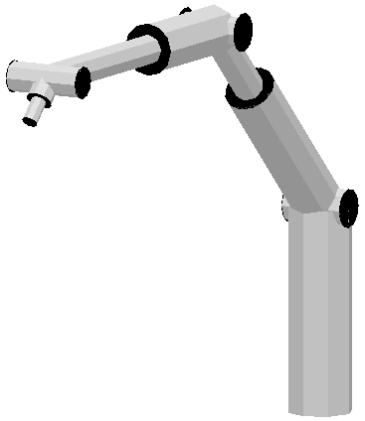

Fig. 4 Sketch of the PA10-like robot.

Table 1 Denavit-Hartenberg parameters of the PA10-like robot.

Joint #	d, m	θ_0, deg	a, m	α, deg
1	0.315	0.0	0.0	-90.0
2	0.0	45.0	0.0	90.0
3	0.450	0.0	0.0	-90.0
4	0.0	45.0	0.0	90.0
5	0.500	0.0	0.0	-90.0
6	0.0	45.0	0.0	90.0
7	0.080	0.0	0.0	0.0

Table 2 Inertia properties of the PA10-like robot.

Body #	M, kg	x_{CM}, m
2	8.41	0.06325
3	3.51	0.08944
4	4.31	0.04609
5	3.45	0.16470
6	1.46	-0.03000
7	0.24	-0.02900

3.2.1 Anisotropic Ergonomy Functions

The case of anisotropic ergonomy functions is considered first. The prescribed trajectory of the node that represents the last body is

$$\left\{ \begin{array}{c} x \\ y \\ z \end{array} \right\} = \left\{ \begin{array}{c} -0.6\,(1-\cos(\pi t)) \\ -0.3\,(1-\cos(2\pi t)) \\ 0.0 \end{array} \right\} \text{ m.} \tag{41}$$

The motion is periodic, with period $T = 1$ s. No specific orientation is prescribed; as a consequence the problem is $7 - 3 = 4$ times underdetermined.

Figures 5 and 6 respectively show the end-effector trajectory in a Cartesian x–y plane and the time histories of the x and y components of the end-effector trajectory. Figure 8 shows the joint angles and torques for two sets of ergonomy functions. In both cases the functions are linear. In the case labeled 'isotropic' the same coefficient

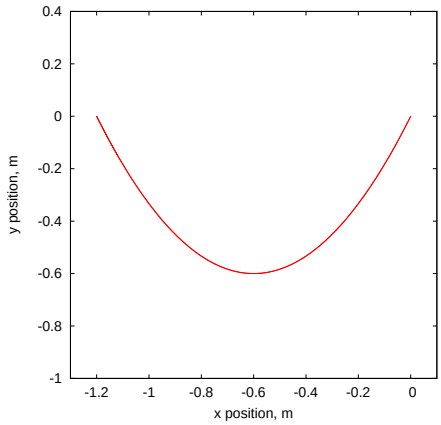

Fig. 5 PA10-like robot: x–y components of end effector trajectory.

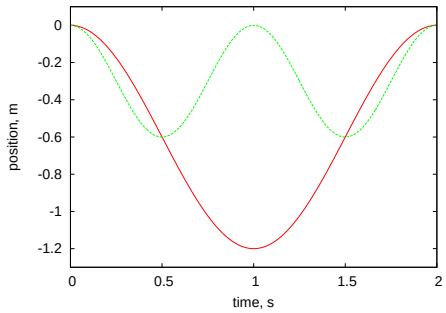

Fig. 6 PA10-like robot: end effector trajectory.

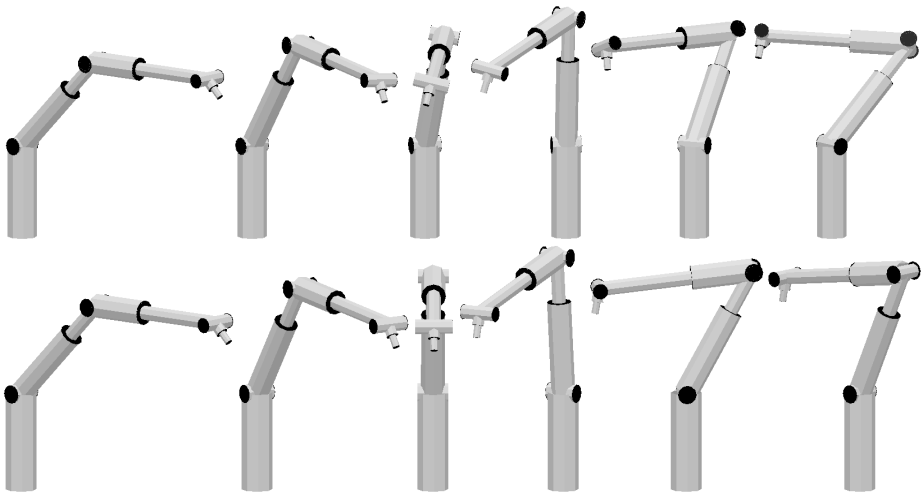

Fig. 7 Sketch of PA10-like robot motion in case of isotropic (top) and anisotropic (bottom) ergonomy functions ($0 \leq t \leq 1$ s, every 0.2 s).

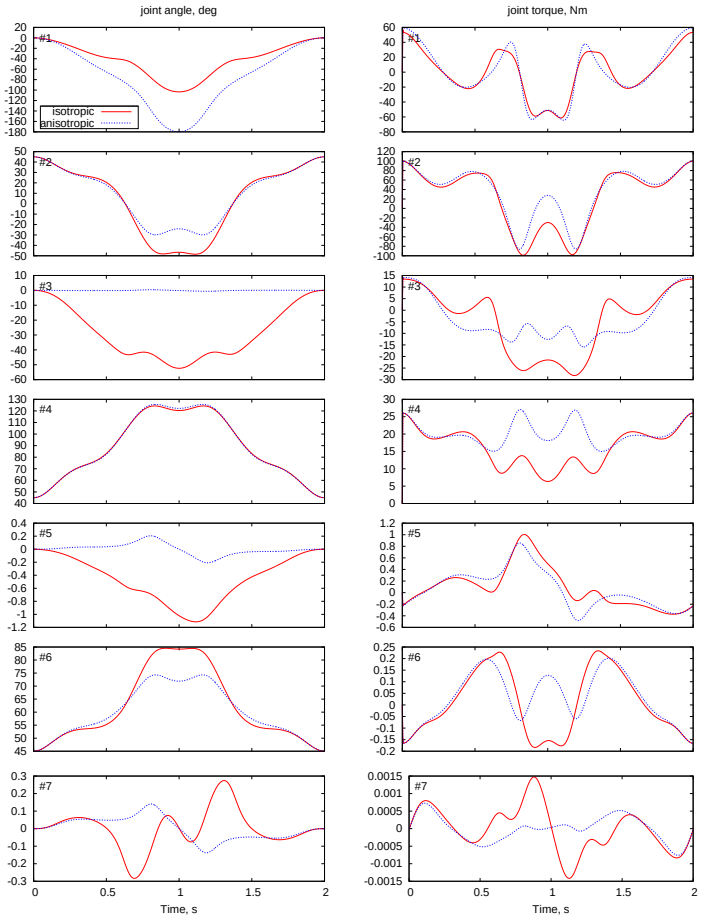

Fig. 8 PA10-like robot: joint angles (degrees) and torques (Nm) in case of isotropic and anisotropic ergonomy functions.

$\mathcal{K}_{ii} = \mathcal{K}$ is used for all joints, while in the one labeled 'anisotropic' $\mathcal{K}_{11} = 10^{-3}\mathcal{K}$ is used for joint #1. This allows joint #1 to provide most of the motion, as opposed to the previous case; as a consequence, joints #3 and #5 do not essentially move, as clearly shown in Figure 8 on the left.

Table 3 PA10-like robot, trajectory data

L	0.18	m
t_0	0.0	s
t_1	1.0	s
t_2	2.0	s
t_3	2.5	s
ω	$\pi/2$	radian/s

3.2.2 Obstacle Avoidance

This problem tests the obstacle avoidance functionality, which is obtained by penalizing the absolute position of the manipulator. The prescribed trajectory is

$$x(t) = \begin{cases} 0 & t_0 \le t \le t_1 \\ L\left(\dfrac{1}{\left(1+\tan^3\left(\omega(t-t_1)\right)\right)^{1/3}} - 1\right) & t_1 \le t \le t_2 \\ -L\left(1+\omega(t-t_2)\right) & t_2 \le t \le t_3 \end{cases} \tag{42a}$$

$$y(t) = \begin{cases} L\omega t & t_0 \le t \le t_1 \\ L\left(\omega t_1 + \dfrac{\tan\left(\omega(t-t_1)\right)}{\left(1+\tan^3\left(\omega(t-t_1)\right)\right)^{1/3}}\right) & t_1 \le t \le t_2 \\ L\left(\omega t_1 + 1\right) & t_2 \le t \le t_3 \end{cases} \tag{42b}$$

$$z(t) = 0 \tag{42c}$$

with $\omega = \pi/(2(t_2 - t_1))$ and data from Table 3. It is inspired by [35], and corresponds to a rectilinear segment in the y direction at constant speed, followed by an arc of a curve of the form $(x/L)^3 + (y/L)^3 = 1$, at nearly constant speed, that bends the trajectory towards the negative direction of the x axis, and finally by another rectilinear segment in the $-x$ direction, again at constant speed. The axis of the tool is required to remain vertical, i.e. oriented along z. As a consequence, 5 components of tool motion are prescribed. The initial configuration of the robot only differs from that of Table 1 for the angle of joint #4, which is now $\theta_0 = 90$ deg. Isotropic ergonomy functions are considered; in the restricted case, joint #4 is required to stay away from point P = {0.3,0.5,0.6} in space using a linear elastic rod between that joint and point P. The rod is active only during the inverse kinematics phase of the analysis and penalizes the reduction of the distance. The distance of joint #4 from point P in the unrestricted and restricted cases is shown in Fig. 9, while a sketch of the robot motion is provided in Fig. 10. The corresponding joint angles and torques are illustrated in Fig. 11.

3.3 Feedforward and Feedback Control of a Bio-inspired Robot

The third application consists of a bio-inspired robot, sketched in Figure 12, made of 11 identical modules connected by revolute hinges. An early version of this problem was analyzed in [27,9]. The hinges of two consecutive modules are rotated $6/11\pi$ radian apart. Each hinge is located 0.0634 m radially from the main axis of the manipulator. The hinge axis is oriented tangentially and normal to the main axis of the manipulator. In the reference configuration each module is spaced 0.058 m along the main axis of the manipulator.

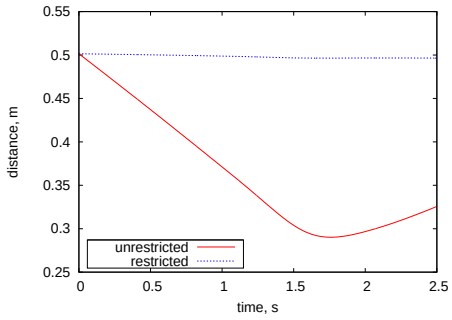

Fig. 9 PA10-like robot: distance of joint #4 from point P = {0.3,0.5,0.6} in unrestricted and restricted cases.

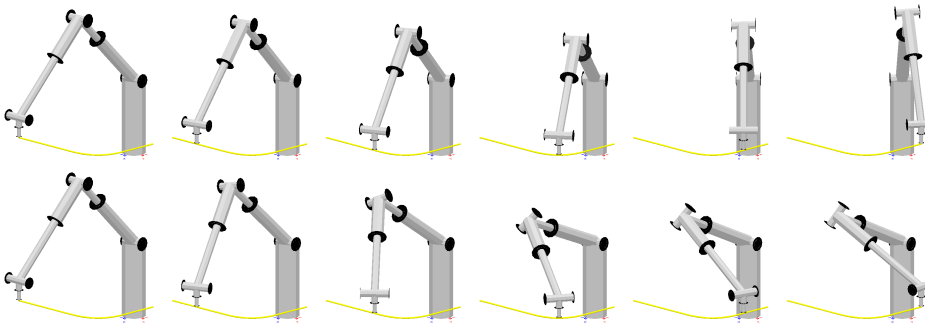

Fig. 10 Sketch of PA10-like robot motion in unrestricted (top) and restricted (bottom) cases ($0 \le t \le 2.5$ s, every 0.5 s).

Figures 13–16 compare motion and loads resulting from the inverse kinematics and inverse dynamics analysis, labeled 'inv. dyn.', with those resulting from direct integration in time. In the case labeled 'f.b. + f.f.' the torques computed by the inverse dynamics analysis are applied in feedforward, while the corresponding motion is used in feedback. In the case labeled 'f.f., no f.b.' the inverse kinematics motion is used in feedback with the same gains used for the previous case, but no feedforward is used.

Figure 13 shows a projection on the x–y plane of the trajectory of the end effector (a lemniscate) from 2 s to 6 s, i.e. after the start-up transient. The prescribed trajectory, including the start-up phase, is

$$\left\{ \begin{array}{c} x \\ y \\ z \end{array} \right\} = f(t) \left\{ \begin{array}{c} 0.125\,\cos(\pi t) \\ 0.125\,\cos(2\pi t) \\ 0.050 \end{array} \right\} \text{ m}$$

$$f(t) = \left\{ \begin{array}{ll} \dfrac{1-\cos(\pi t)}{2} & t < 1 \text{ s} \\ 1 & t \ge 1 \text{ s}. \end{array} \right. \tag{43}$$

The rotation of the end effector is only allowed about the global z axis. As a consequence, the problem is $11 - 5 = 6$ times underdetermined.

Figure 14 shows the time histories of all components of the trajectory of the end effector, while Figure 15 shows those of the joint coordinates. Figure 16 shows the torques of the joints.

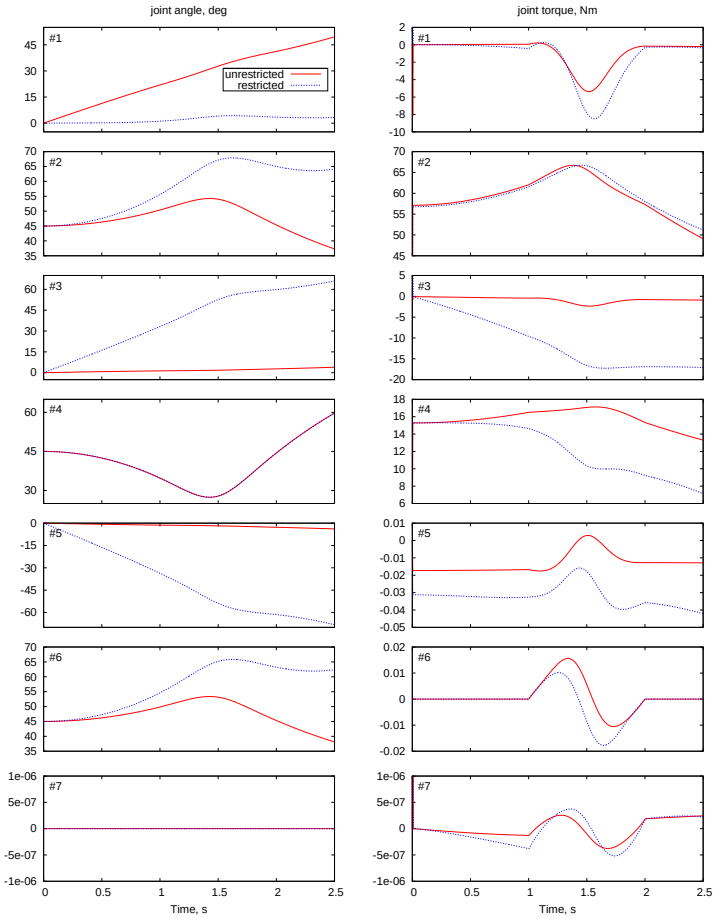

Fig. 11 PA10-like robot: joint angles (degrees) and torques (Nm) in unrestricted (top) and restricted (bottom) cases.

The trajectories obtained using feedforward and feedback are essentially coincident with the ones from inverse analysis; the maximum error in the position of the end effector is of the order of $5 \cdot 10^{-5}$ m. On the contrary, the trajectories obtained without feedforward show significant errors, thus justifying the effort placed in predicting the feedforward torques in addition to the desired joint trajectories.

Figures 17–20 show the same results of the previous ones with a +20% perturbation of the mass of the last body, which represents an uncertainty on the mass of the end-effector. The curves labeled 'f.f. + f.b., nom.' use the nominal feedback gains, while those labeled 'f.f. + f.b.' are obtained using twice as large feedback gains. In both cases the curves differ from the nominal ones, but the error is limited, and can be reduced using the feedback control.

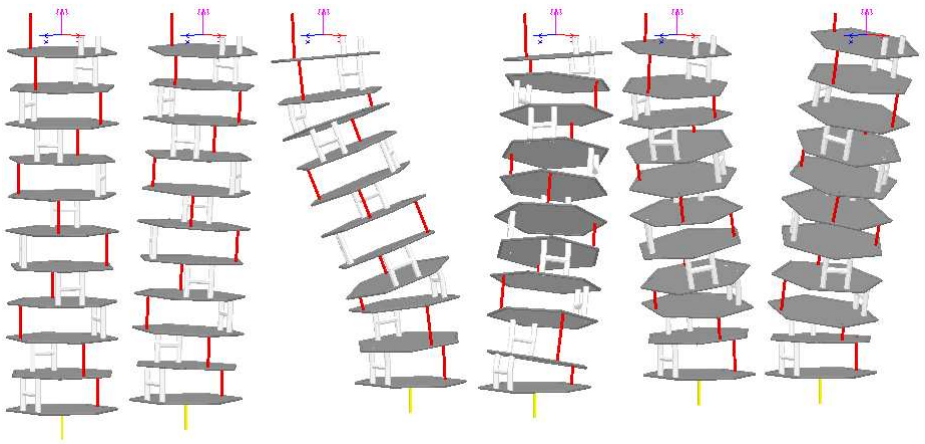

Fig. 12 Sketch of bio-inspired robot during the start-up transient.

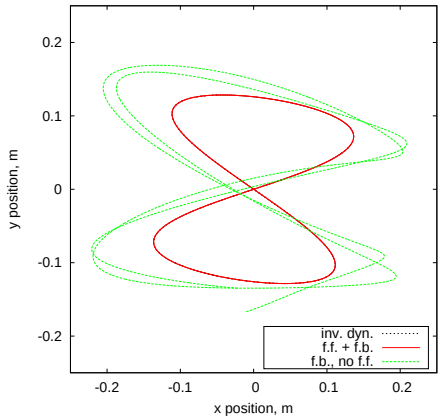

Fig. 13 Bio-inspired robot: x–y components of end effector trajectory.

4 Conclusions

An algorithm for the real-time solution of inverse kinematics and inverse dynamics of redundant manipulators formulated in redundant coordinates has been presented. The configuration is determined first, by solving a nonlinear problem consisting of the passive and the control constraints, augmented by a functional consisting of a choice of several local optimality conditions usually formulated as least-squares minimization of cost functions. Ergonomy, proximity and torque boundary avoidance are proposed. Velocity and acceleration are subsequently computed by solving linear problems where the redundancy is eliminated by minimizing the distance with respectively the velocity and accelerations resulting from the numerical differentiation of the already computed configuration change from the previous time step. The solution of the inverse dynamics problem yields the joint torques required to implement the computed torque control of the actual system. The algorithm has been implemented in a free, general-purpose multibody solver, exploiting the availability of general-purpose elastic

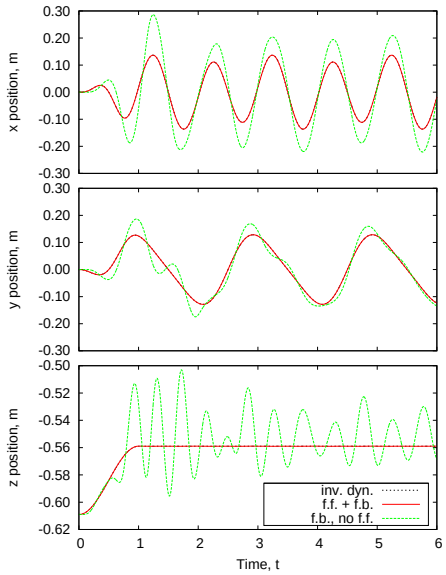

Fig. 14 Bio-inspired robot: end effector trajectory.

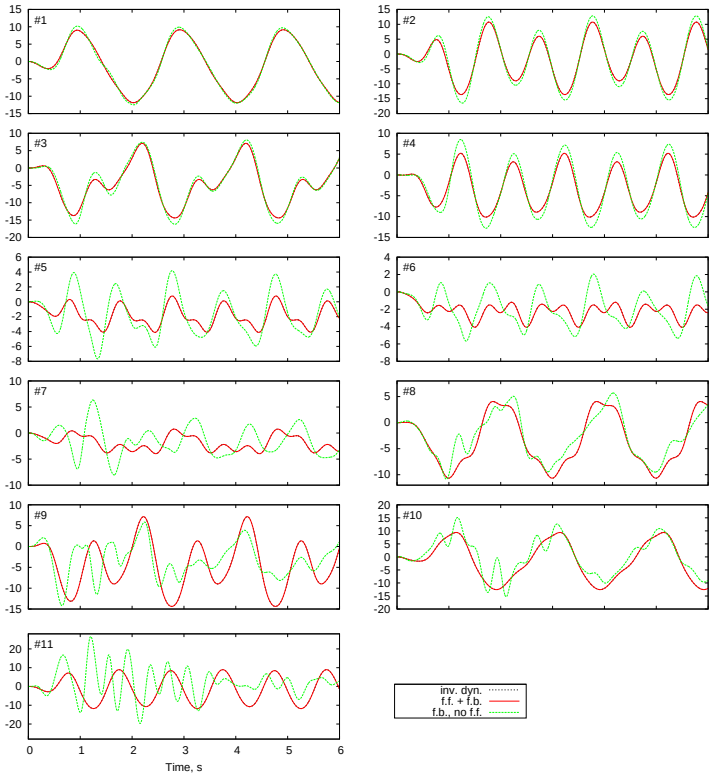

Fig. 15 Bio-inspired robot: joint angles (degrees).

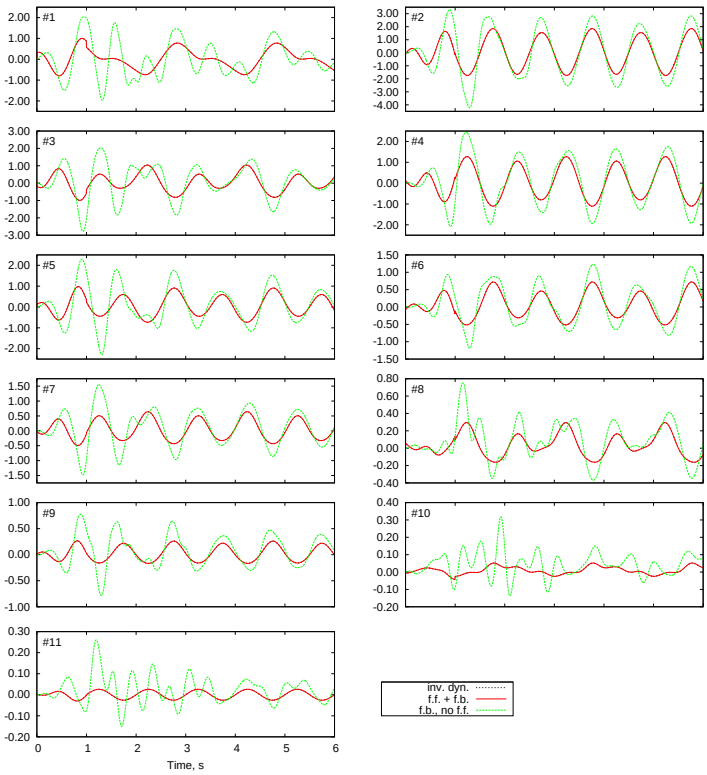

Fig. 16 Bio-inspired robot: joint torques (Nm).

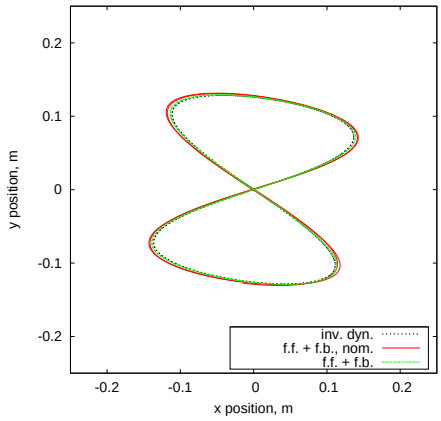

Fig. 17 Bio-inspired robot: x–y components of end effector trajectory with +20% mass perturbation.

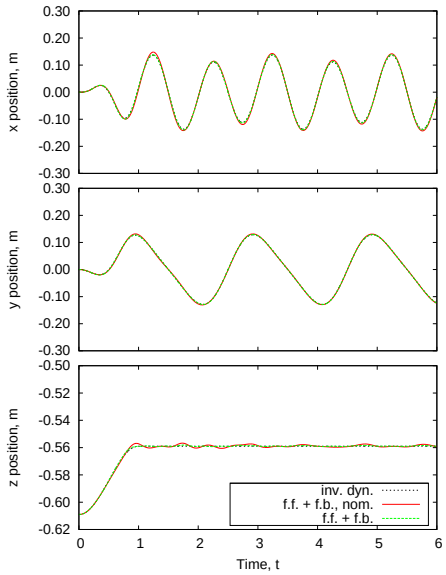

Fig. 18 Bio-inspired robot: end effector trajectory with +20% mass perturbation.

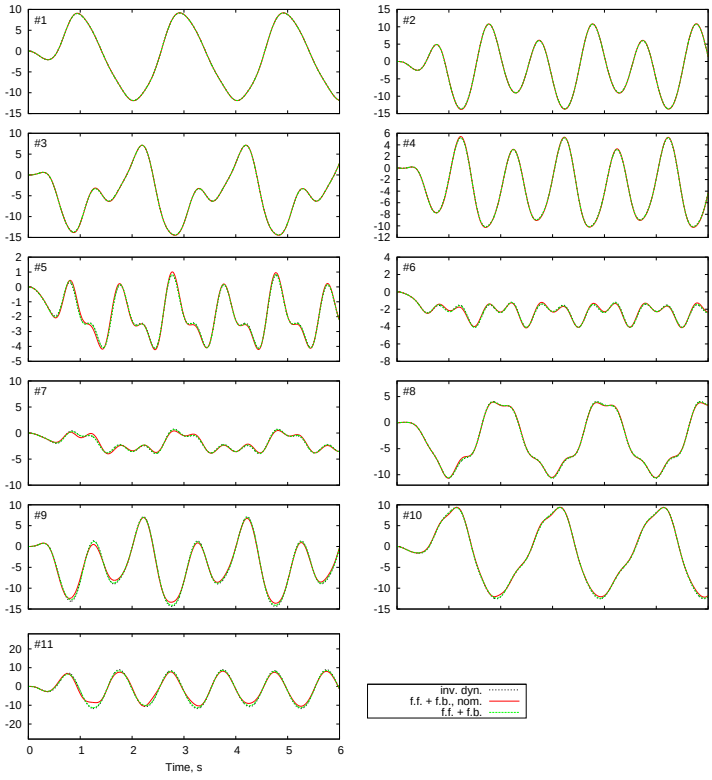

Fig. 19 Bio-inspired robot: joint angles (degrees) with +20% mass perturbation.

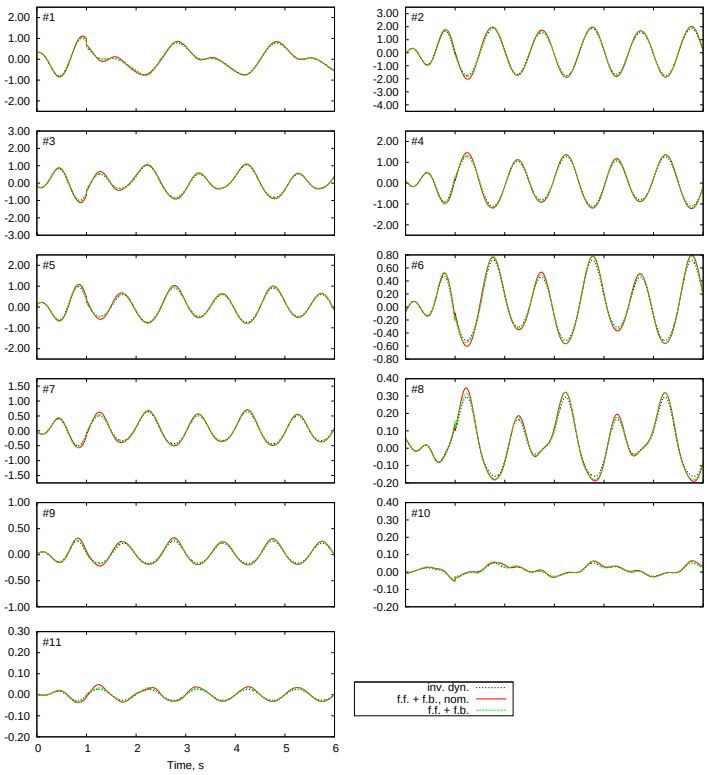

Fig. 20 Bio-inspired robot: joint torques (Nm) with +20% mass perturbation.

elements to model arbitrary ergonomy functions, and of the mass matrix to implement proximity cost functions. Applications of increasing redundancy and complexity are presented, to highlight the capability of the approach to produce accurate predictions of the kinematics and dynamics of the system and to support the implementation of robust feedback control systems based on feedback linearization.

Acknowledgments

The author gratefully acknowledges the role of Dr. Alessandro Fumagalli in the implementation of the fully determined inverse dynamics problem in MBDyn.

References

1. J. Angeles. *Fundamentals of Robotic Mechanical Systems — Theory, Methods, and Algorithms*. Springer, 3rd edition, 2007.
2. L. Sciavicco and B. Siciliano. *Modeling and Control of Robot Manipulators*. Springer, 2000.
3. W. Blajer and K. Kołodziejczyk. Improved DAE formulation for inverse dynamics simulation of cranes. *Multibody System Dynamics*, 25(2):131–143, 2011. doi:10.1007/s11044-010-9227-6.

4. A. Fumagalli, P. Masarati, M. Morandini, and P. Mantegazza. Control constraint realization for multibody systems. *J. of Computational and Nonlinear Dynamics*, 6(1):011002 (8 pages), January 2011. doi:10.1115/1.4002087.
5. R. Seifried. Two approaches for feedforward control and optimal design of underactuated multibody systems. *Multibody System Dynamics*, 27(1):75–93, 2012. doi:10.1007/s11044-011-9261-z.
6. A. Chae, C. Atkeson, J. Griffiths, and J. Hollerbach. Experimental evaluation of feedforward and computed torque control. In *IEEE International Conference on Robotics and Automation*, volume 3, pages 165–168, Raleigh, NC, USA, March 31–April 3 1987. doi:10.1109/ROBOT.1987.1088020.
7. A. Codourey. Dynamic modeling of parallel robots for computed-torque control implementation. *The Intl. J. of Robotics Research*, 17(12):1325–1336, December 1998. doi:10.1177/027836499801701205.
8. L. Sciavicco and B. Siciliano. A solution algorithm to the inverse kinematic problem for redundant manipulators. *Robotics and Automation*, 4(4):403–410, August 1988. doi:10.1109/56.804.
9. A. Fumagalli and P. Masarati. Real-time computed torque control using general-purpose multibody software. *Multibody System Dynamics*, 22(1):47–68, 2009. doi:10.1007/s11044-009-9153-7.
10. S. Staicu. Matrix modeling of inverse dynamics of spatial and planar parallel robots. *Multibody System Dynamics*, 27(2):239–265, 2012. doi:10.1007/s11044-011-9281-8.
11. T. Zilic, J. Kasac, Z. Situm, and M. Essert. Simultaneous stabilization and trajectory tracking of underactuated mechanical systems with included actuators dynamics. *Multibody System Dynamics*, 29(1):1–19, January 2013. doi:10.1007/s11044-012-9303-1.
12. A. Taghvaeipour, J. Angeles, and L. Lessard. Constraint-wrench analysis of robotic manipulators. *Multibody System Dynamics*, 29(2):139–168, February 2013. doi:10.1007/s11044-012-9318-7.
13. A. Akbarzadeh, J. Enferadi, and M. Sharifnia. Dynamics analysis of a 3-RRP spherical parallel manipulator using the natural orthogonal complement. *Multibody System Dynamics*, Available online 23 July 2012. doi:10.1007/s11044-012-9321-z.
14. X. Liu, H. Baoyin, and X. Ma. Optimal path planning of redundant free-floating revolute-jointed space manipulators with seven links. *Multibody System Dynamics*, 29(1):41–56, January 2013. doi:10.1007/s11044-012-9323-x.
15. D. P. Martin, J. Baillieul, and J. M. Hollerbach. Resolution of kinematic redundancy using optimization techniques. *IEEE Transactions on Robotics and Automation*, 5(4):529–533, August 1989. doi:10.1109/70.88067.
16. A. R. Hirakawa and A. Kawamura. Trajectory planning of redundant manipulators for minimum energy consumption without matrix inversion. In *IEEE International Conference on Robotics and Automation*, volume 3, pages 2415–2420, Albuquerque, NM, USA, April 20–25 1997. doi:10.1109/ROBOT.1997.619323.
17. W. Schiehlen. Multibody system dynamics: Roots and perspectives. *Multibody System Dynamics*, 1(2):149–188, June 1997. doi:10.1023/A:1009745432698.
18. A. Ben-Israel and T. N. E. Greville. *Generalized inverses: theory and applications*. Wiley, New York, 1974.
19. L. Mariti, N. P. Belfiore, E. Pennestrì, and P. P. Valentini. Comparison of solution strategies for multibody dynamics equations. *Intl. J. Num. Meth. Engng.*, 2011. doi:10.1002/nme.3190.
20. A. Fumagalli and P. Masarati. Efficient application of Gauss' principle to generic mechanical systems. DIA SR-08-01, Politecnico di Milano, Dipartimento di Ingegneria Aerospaziale, 2008.
21. J. M. Hollerbach and K. C. Suh. Redundancy resolution of manipulators through torque optimization. *IEEE Journal of Robotics and Automation*, 3(4):308–316, August 1987. doi:10.1109/JRA.1987.1087111.
22. A. Liégeois. Automatic supervisory control of the configuration and behavior of multibody mechanisms. *IEEE Transactions on Systems, Man and Cybernetics*, 7(12):868–871, December 1977. doi:10.1109/TSMC.1977.4309644.
23. T. Yoshikawa. Dynamic manipulability of robot manipulators. In *IEEE International Conference on Robotics and Automation*, pages 1033–1038, St. Louis, Missouri, USA, March 25–28 1985. doi:10.1109/ROBOT.1985.1087277.
24. K. L. Doty, C. Melchiorri, E. M. Schwartz, and C. Bonivento. Robot manipulability. *IEEE Transactions on Robotics and Automation*, 11(3):462–468, June 1995. doi:10.1109/70.388791.

25. K. C. Suh and J. M. Hollerbach. Local versus global torque optimization of redundant manipulators. In *IEEE International Conference on Robotics and Automation*, pages 619–624, March 1987. doi:10.1109/ROBOT.1987.1087955.
26. A. Müller and P. Maißer. Kinematic and dynamic properties of parallel manipulators. *Multibody System Dynamics*, 5(3):223–249, 2001. doi:10.1023/A:1011484402247.
27. A. Fumagalli, G. Gaias, and P. Masarati. A simple approach to kinematic inversion of redundant mechanisms. In *ASME IDETC/CIE 2007*, Las Vegas, Nevada, 4–7 September 2007. (DETC2007-35285).
28. E. Pennestrì, R. Stefanelli, P. P. Valentini, and L. Vita. Virtual musculo-skeletal model for the biomechanical analysis of the upper limb. *Journal of Biomechanics*, 40(6):1350–1361, 2007. doi:10.1016/j.jbiomech.2006.05.013.
29. P. Masarati and M. Morandini. Intrinsic deformable joints. *Multibody System Dynamics*, 23(4):361–386, 2010. doi:10.1007/s11044-010-9194-y.
30. O. A. Bauchau, L. Li, P. Masarati, and M. Morandini. Tensorial deformation measures for flexible joints. *J. of Computational and Nonlinear Dynamics*, 6(3), July 2011. doi:10.1115/1.4002517.
31. P. Masarati. Constraint stabilization of mechanical systems in ODE form. *Proc. IMechE Part K: J. Multi-body Dynamics*, 225(1):12–33, 2011. doi:10.1177/2041306810392117.
32. C. F. Gauss. Ueber ein neues algemeines Grundgesetz der Mechanik. *J. fuer die reine und angewandte Mathematik*, 4:232–235, 1829. In German.
33. P. Masarati. A formulation of kinematic constraints imposed by kinematic pairs using relative pose in vector form. *Multibody System Dynamics*, 29(2):119–137, 2013. doi:10.1007/s11044-012-9320-0.
34. A. Zanoni, P. Masarati, and G. Quaranta. Rotorcraft pilot impedance from biomechanical model based on inverse dynamics. In *International Mechanical Engineering Congress & Exposition (IMECE) 2012*, Houston, Texas, November 9–15 2012. Paper No. IMECE2012-87533.
35. J.-F. Gauthier, J. Angeles, and S. Nokleby. Optimization of a test trajectory for SCARA systems. In *Advances in Robot Kinematics: Analysis and Design*, volume 4, pages 225–234. Springer, 2008.