
Alessandro De Luca
Giuseppe Oriolo

Dipartimento di Informatica e Sistemistica
Università di Roma "La Sapienza"
Via Eudossiana 18, 00184 Roma, Italy
deluca@dis.uniroma1.it
oriolo@dis.uniroma1.it

Trajectory Planning and Control for Planar Robots with Passive Last Joint

Abstract

We present a method for trajectory planning and control of planar robots with a passive rotational last joint. These underactuated mechanical systems, which are subject to nonholonomic second-order constraints, are shown to be fully linearized and input-output decoupled by means of a nonlinear dynamic feedback. This objective is achieved in a unified framework, both in the presence or absence of gravity. The linearizing output is the position of the center of percussion of the last link. Based on this result, one can plan smooth trajectories joining in finite time any initial and desired final state of the robot; in particular, transfers between inverted equilibria and swing-up maneuvers under gravity are easily obtained. We also address the problem of avoiding the singularity induced by the dynamic linearization procedure through a careful choice of output trajectories. A byproduct of the proposed method is the straightforward design of exponentially stable tracking controllers for the generated trajectories. Simulation results are reported for a 3R robot moving in a horizontal and vertical plane. Possible extensions of the approach and its relationships with the differential flatness technique are briefly discussed.

KEY WORDS—underactuated robots, dynamic feedback linearization, nonholonomic trajectory planning, differential flatness, swing-up maneuvers

1. Introduction

One of the most interesting consequences of nonholonomy in robotic systems is that it allows one to control the configuration of the whole mechanism with a reduced number of inputs (Bicchi and Goldberg 1996). The price for this benefit is that planning and controlling trajectories become much harder than in the case of holonomic systems; for example, stabilization at a point cannot be obtained via smooth time-invariant feedback (Brockett 1983). Many well-known exam-

ples of nonholonomy are found in the class of kinematic or dynamic models of systems with Pfaffian (i.e., first-order and linear in the generalized velocities) differential constraints, such as wheeled mobile robots (Laumond 1998), multifingered hands with rolling contacts (Murray, Li, and Sastry 1994; Bicchi 2000), and space manipulators subject to angular momentum conservation (Vafa and Dubowsky 1990).

There is, however, another class of controlled mechanical systems that exhibit a nonholonomic behavior, namely, robots with passive degrees of freedom (also called *underactuated* in the literature). These mechanisms arise in a number of situations, ranging from nonprehensile manipulation (Lynch and Mason 1999) to robot acrobatics (Nakanishi, Fukuda, and Koditschek 2000), from legged locomotion (Spong 1999) to surgical robotics (Funda et al. 1996), from free-floating robots (Faiz and Agrawal 1998) to manipulators with flexibility concentrated at the joints (De Luca and Lucibello 1998) or distributed along the links (De Luca et al. 2001). Another particularly interesting example is that of manipulators to be operated in spite of actuator failure (Arai and Tachi 1991). In this latter case, in order to preserve active operation of the system, one needs to take into account the arising nonholonomic constraints at both the trajectory planning and control level.

In a dynamical setting, all robots with n DoF's and $m < n$ actuators are subject to a set of $n - m$ second-order differential constraints in the form

$$A(q)\ddot{q} + b(q, \dot{q}) + c(q) = 0, \quad (1)$$

with q parameterizing the robot configuration space. At a given robot state (q, \dot{q}) , constraints (1) impose restrictions on the admissible generalized accelerations for any actuation command. As first shown in Oriolo and Nakamura (1991), if these constraints are nonintegrable then the whole robot state space may be still accessible by suitable maneuvering. Note that eq. (1) is *affine* in the acceleration, thus generalizing the most studied case of first-order nonholonomic constraints, namely those of the Pfaffian type. From the planning and control point of view, the realization of the nonholonomic nature of underactuated robots has thus presented to researchers a

theoretically challenging example of nonholonomic systems with *nontrivial drift*. Moreover, it has suggested that trajectory planning and control may be attacked by taking inspiration from techniques developed for driftless nonholonomic systems.

During the last decade, a remarkable research activity has focused on planning and control of underactuated robots, e.g., see Spong (1998) and references therein. General nonholonomy and state controllability conditions for these systems can be found in Oriolo and Nakamura (1991). Controllability properties when starting from an equilibrium configuration have been studied in Lewis and Murray (1997). The related concept of kinematic controllability (Bullo and Lynch 2001) allows, under certain conditions, to backstep the planning of feasible rest-to-rest trajectories from a dynamic to a kinematic (i.e., first-order) problem. In Rathinam and Murray (1998), general conditions were given for mechanical systems underactuated by one control (i.e., with $m = n - 1$) to be flat. In spite of these advances in the analysis, a general theory for planning and control of underactuated robots is not yet available and the most successful solutions were obtained tailoring the approach to the specific case considered. The following review is limited to ground-based rigid manipulators with passive joints (and no brakes).

Starting with 2-DoF robots, one should note that the problem of planning feasible point-to-point trajectories with a single actuated joint is still unsolved. To this date, asymptotic transfer trajectories that comply with the system nonholonomic constraint may only be generated as a byproduct of feedback stabilization strategies. In the absence of gravity, stabilization of a planar 2R robot with a passive elbow joint has been obtained in Nakamura, Suzuki, and Koinuma (1997), in which a time-varying feedback is designed via Poincaré map analysis, and in De Luca, Mattone, and Oriolo (2000), where an iterative steering technique is used to design a nonsmooth feedback that guarantees robust convergence to a desired configuration. The same technique was used in De Luca, Iannitti, and Oriolo (2001) for achieving configuration control of a PR robot with a second passive joint. Such non-conventional approaches are needed because—exactly as in the case of Pfaffian nonholonomic systems—smooth stabilization at an equilibrium is not possible.

In the presence of gravity, the case of 2R planar robots with a single actuator has been considered, among the others, in Spong (1995) and De Luca and Oriolo (1998) (Acrobot, passive first joint) and Spong and Block (1995) (Pendubot, passive second joint). Since the approximate linearization of these systems is controllable, they are in principle easier to control, at least locally. However, due to the gravitational drift, the region of the state space where the robot can be kept in equilibrium is reduced, and consists of two disjoint manifolds. Moving between these two requires appropriate swing-up maneuvers, whose synthesis has been so far tackled by energy and/or passivity-based control techniques.

Another class of underactuated robots that has been studied in some detail are three-link planar manipulators with passive rotational third joint in the absence of gravity. Arai et al. (1998) have shown how to plan rest-to-rest trajectories through a sequence of elementary maneuvers consisting of pure translations of the third link or pure rotations around its center of percussion (CP). Under the assumption that the first two joints are prismatic, Imura et al. (1996) proved that the system can be transformed into second-order chained form (another concept derived from motion planning techniques for Pfaffian nonholonomic systems) via static state feedback, and later (Yoshikawa, Kobayashi, and Watanabe 2000) exploited this result to plan trajectories for the robot. In De Luca and Oriolo (2000a, b) we have built upon these works, showing that the position of the third link CP becomes a linearizing output under the action of a dynamic state feedback, even in the presence of gravity. On the resulting linear and decoupled closed-loop system, trajectory planning can be performed using smooth trajectories that interpolate, in a given finite time, any initial and desired final state.

We mention that trajectory planning results for systems that are mechanically equivalent to the latter class have been obtained based on differential flatness (Martin, Devasia, and Paden 1996; Faiz and Agrawal 1998; Rathinam and Murray 1998). On the other hand, the approach presented in De Luca and Oriolo (2000a, b), and further extended here, allows to proceed in a single framework for the gravity/no gravity case, provides a simple solution to the associated trajectory tracking problem, and explicitly addresses the issue of possible singularities affecting both planning and control, so far overlooked in other papers. In particular, this is a basic problem in swing-up maneuvers for gymnast robots; clearly, such maneuvers may not be relevant for the VTOL aircraft considered in Martin, Devasia, and Paden 1996) or in the absence of gravity (Rathinam and Murray 1998). In any case, for the sake of clarity, we shall discuss later in the paper the relationships between dynamic feedback linearization and differential flatness, providing also a brief historical perspective.

In this paper, we extend the powerful feedback linearization approach to a special class of underactuated mechanisms called $(n-1)X_a-R_n$ planar robots, having $n-1$ active joints of any type and a passive rotational last joint, so as to design a trajectory planning method that is valid both in the absence and presence of gravity. Under an appropriate regularity assumption, the robot can be transformed into a fully linear, input-output decoupled system by using a second-order dynamic feedback compensator. As a result of dynamic feedback linearization, each coordinate of the CP is driven independently by an auxiliary input through a chain of integrators. Therefore, it is sufficient to solve an interpolation problem for the CP point to generate a feasible point-to-point trajectory and the associated nominal inputs. In particular, in the presence of gravity, robot transfers between inverted equilibria and swing-up maneuvers between disjoint equilibrium manifolds can be

generated in a rather straightforward way. As a byproduct of this approach, global exponential tracking of the generated trajectory is guaranteed by adding a linear feedback (in the linearizing coordinates) to the feedforward command.

The paper is organized as follows. In the next section, we recall some background on feedback linearization and compare the use of linearizing dynamic feedback with the approach based on differential flatness. In Section 3 we manipulate the general structure of the dynamics of $(n-1)X_a-R_u$ planar robots, choosing a specific set of generalized coordinates and performing a partial feedback linearization so as to obtain a simpler problem setting. The dynamic feedback linearization design is presented in Section 4, while the problem of planning and tracking state-to-state trajectories is formulated and solved in Section 5. Details and numerical results are given for a 3R underactuated robot in the absence (Section 5.1) and presence (Section 5.2) of gravity; in particular, we discuss strategies guaranteeing that the regularity assumption is always satisfied. A discussion on possible extensions completes the paper.

2. Background on Feedback Linearization

In this section, we recall briefly the basics of feedback linearization theory. For a detailed treatment, see Isidori (1995).

Consider a generic nonlinear dynamic system

$$\dot{x} = f(x) + g(x)u, \quad (2)$$

where x is the n -dimensional state and u is the m -dimensional input. The exact state linearization problem via *static feedback* consists in finding a control law of the form

$$u = \alpha(x) + \beta(x)v, \quad (3)$$

with $\beta(x)$ nonsingular and v an auxiliary input, and a change of coordinates $z = \phi(x)$ such that, in the new coordinates, the closed-loop system is linear and controllable. Necessary and sufficient conditions exist for the solvability of this problem (Isidori 1995, Theorem 5.2.3). For fully actuated robots, i.e., with a number of generalized coordinates equal to the number of input commands, these conditions are trivially satisfied and the control law (3) leads to the well-known computed torque method.

If static feedback does not allow one to solve the problem, one can try to obtain exact state linearization by means of a *dynamic feedback* compensator of the form

$$\begin{aligned} u &= \alpha(x, \xi) + \beta(x, \xi)v \\ \dot{\xi} &= \gamma(x, \xi) + \delta(x, \xi)v, \end{aligned} \quad (4)$$

where ξ is the ν -dimensional compensator state, together with a change of coordinates $z = \phi(x, \xi)$. Only sufficient conditions are available for the solvability of this problem (see, e.g., Isidori 1995, Proposition 5.4.4). In robotics, dynamic

feedback has been used for the exact linearization of manipulators with elastic joints (De Luca and Lucibello 1998) and of nonholonomic wheeled mobile robots (d'Andrea-Novel, Bastin, and Campion 1995).

In both the static and the dynamic feedback case, if the sufficient conditions are satisfied, the actual construction of the control law requires us to identify an auxiliary m -dimensional output $y = h(x)$, such that the corresponding vector relative degree is well defined and the sum of its elements equals n . In particular, this output vector, together with its derivatives up to a certain order, defines the linearizing coordinates z . As a byproduct, the control laws (3) or (4) also yield input-output decoupling between v and y .

In practice, however, the linearizing output y is not known in advance. Therefore, one typically defines a candidate class of output functions to which the linearization algorithm is applied. The algorithm progresses by differentiating each component of the candidate output until some component of the input explicitly appears; call n_i the differentiation order of the i th component ($i = 1, \dots, m$). If the Jacobian of the resulting input-output differential map (referred to as the *decoupling matrix* of the system) is nonsingular, then system (2) with the candidate output has well-defined vector relative degree $\{n_1, \dots, n_m\}$. If, in addition, it is $\sum_i n_i = n$, the control input obtained by inversion of the differential map provides exact state linearization via static feedback in the form (3).

If the decoupling matrix is singular, one can proceed by adding integrators on (a subset of) the input channels, possibly after a state-dependent transformation in the input space. This operation, called *dynamic extension*, converts input components into states of a dynamic compensator, which are driven in turn by new inputs. Differentiation of the output may now continue until either it is possible to solve for the new inputs or the dynamic extension process has to be repeated. If the algorithm terminates after a finite number of iterations, the system is *invertible* from the chosen output and the number of added integrators gives the dimension ν of the dynamic compensator (4). If $\sum_i n'_i = n + \nu$, being $\{n'_1, \dots, n'_m\}$ the vector relative degree of the *extended* system, the control input obtained by inversion of the final input-output differential map, together with the dynamic extension, provides exact state linearization via dynamic feedback in the form (4).

It should be noted that the fulfillment of the condition on the sum of the relative degrees is strictly related to the absence of zero dynamics of the original system with respect to the linearizing output.

2.1. Dynamic Feedback Linearization vs. Differential Flatness

The use of dynamic feedback for input-output decoupling and exact state linearization dates back to the mid 1980s, e.g., see Descusse and Moog (1985) and Charlet, Lévine, and Marino (1989). A closely related concept arose in the early 1990s under the name of *differential flatness* (for an introduction

and a summary of applications, see Fliess et al. (1995)). Essentially, a system of the form (2) is flat if one may find a set of outputs y such that the state and the input can be expressed algebraically in terms of y and a certain number r of its derivatives, i.e.,

$$\begin{aligned} x &= x(y, \dot{y}, \ddot{y}, \dots, y^{[r]}) \\ u &= u(y, \dot{y}, \ddot{y}, \dots, y^{[r]}). \end{aligned} \tag{5}$$

It is easy to understand that flat outputs play the same role of linearizing outputs in the dynamic feedback linearization approach. As a matter of fact, differential flatness is *equivalent* to dynamic feedback linearizability on an open and dense set of the state space. The two properties have already been critically compared in Martin, Murray, and Rouchon (1997) and van Nieuwstadt and Murray (1998). Below we debate, mostly from an operative point of view, some of the arguments proposed therein.

First of all, establishing the existence and providing the expression of a flat or linearizing output is by no means trivial and is the major difficulty of both approaches. In fact, only necessary *or* sufficient conditions are available for testing the two properties. It was argued that physical arguments can be exploited in the search of flat outputs, but this clearly applies also to linearizing outputs.

It is a popular opinion that flatness is particularly effective for solving the problem of generating feasible point-to-point trajectories for nonholonomic and/or underactuated systems. In particular, purely algebraic computations are required, namely planning an arbitrary interpolating trajectory $y = y_d(t)$ for the flat output and transforming it back to a state trajectory and the associated nominal input command via eqs. (5). However, the same result is obtained within the dynamic feedback linearization framework by inverting the transformation $z = \phi(x, \xi)$, with z being composed of y and its derivatives up to $\{n'_1 - 1, \dots, n'_m - 1\}$, evaluated for $y = y_d(t)$. The nominal input is readily provided by eq. (4) with $v = (y_{1d}^{[n'_1]}(t), \dots, y_{md}^{[n'_m]}(t))$. Note that singularities in the input and/or state transformations may affect both techniques and must be safely avoided when planning the output trajectory $y_d(t)$. This potential problem, often overlooked in the flatness literature, is explicitly addressed in this paper within the feedback linearization approach.

The difference between the two techniques becomes significant when designing tracking controllers for the generated output (and associated state) trajectories, in order to allow the recovery of initial state errors and the rejection of exogenous disturbances. With dynamic feedback linearization, the problem essentially reduces to stabilizing chains of integrators in the transformed coordinates z , so that exponentially stable controllers are easily derived. A purely flatness-based approach would instead require the solution of a nonlinear stabilization problem formulated in the original coordinates—often, one may need to settle for a controller of local validity.

Common arguments in favor of flatness claim that the control design is more natural in the original coordinates and that no inversion of the system dynamics is performed. However, many applications have shown that the latter strategy, implicit in the exact linearization approach, has also its advantages, for it realizes an automatic scheduling of control gains. As an example, the computed torque method for fully actuated robots results in the convenient modulation of the constant PD gains through the inertia matrix evaluated at the current configuration. In any case, the feedback linearization approach exhibits other benefits, namely global validity of the tracking controller and guaranteed performance in the linearizing coordinates. In particular, linearity of the error transients implies a degree of predictability even for the motion in the original coordinates, especially when the linearizing output is a physically significant quantity, as common in mechanical systems. In this paper, we will show that planar robots with a single degree of underactuation are an example of such situation.

3. Dynamic Model of $(n - 1)X_a$ - R_u Planar Robots

Consider a planar robot with n joints. The first $n - 1$ joints are actuated (active) and of generic nature, i.e., any combination of prismatic ($X=P$) and/or rotational ($X=R$) joints. The last joint is unactuated (passive) and rotational. Let $q = (q_1, \dots, q_n)$ be any set of generalized coordinates such that $q_n = \theta$, the orientation of the last link with regard to the x -axis. The dynamic model of this robot has the form

$$B(q)\ddot{q} + c(q, \dot{q}) + g(q) = \begin{bmatrix} \tau \\ 0 \end{bmatrix}, \tag{6}$$

where $\tau = (\tau_1, \dots, \tau_{n-1})$ is the vector of generalized forces performing work on the active degrees of freedom, $B(q)$ is the symmetric positive definite generalized inertia matrix, $c(q, \dot{q})$ is the vector of Coriolis and centrifugal terms, and $g(q)$ is the vector of gravitational terms. We shall assume that the plane of motion for the robot is tilted by a generic angle $\psi \in [0, 90^\circ]$ with regard to the vertical plane through the x -axis. Thus, $g(q) \equiv 0$ for $\psi = 90^\circ$ (zero gravity).

To simplify model analysis and control design, it is convenient to use a specific set of generalized coordinates. In particular, let $q = (q_1, \dots, q_{n-3}, x, y, \theta) = (q_a, \theta)$, where (x, y) are the cartesian coordinates of the base of the last link (see Figure 1). Letting $s\theta = \sin \theta$ and $c\theta = \cos \theta$, the dynamic model takes the following partitioned form:

$$\left[\begin{array}{ccc|c} B_a(q_a) & & \begin{matrix} 0_{n-3} \\ -m_n d_n s\theta \\ m_n d_n c\theta \end{matrix} & \ddot{q}_a \\ \hline 0_{n-3}^T & -m_n d_n s\theta & m_n d_n c\theta & \ddot{\theta} \end{array} \right]$$

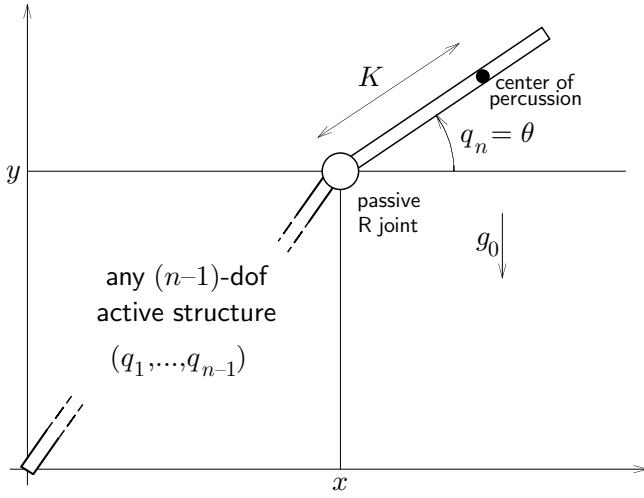


Fig. 1. Coordinate assignment for a $(n - 1)X_a-R_n$ planar robot.

$$+ \begin{bmatrix} c_a(q, \dot{q}) \\ 0 \end{bmatrix} + \begin{bmatrix} g_a(q_a) \\ g_0 m_n d_n c\theta \end{bmatrix} = \begin{bmatrix} F_a \\ 0 \end{bmatrix}, \quad (7)$$

where $F_a = (F_1, \dots, F_{n-3}, F_x, F_y)$ are the generalized forces performing work on the q_a coordinates and $g_0 = 9.81 \cdot \cos \psi$. For the n th link, I_n , m_n , and d_n are, respectively, the baricentral inertia, mass, and distance of the center of mass from its base. Note that F_x and F_y are cartesian forces acting at the base of the last link; in the following, we assume that they can be independently assigned. To this end, the planar robot must have at least two active joints, i.e., $n \geq 3$.

The last dynamic equation in (7) is a scalar second-order differential constraint that should be satisfied during any feasible motion. Whenever $d_n \neq 0$, this turns out to be a completely nonholonomic constraint in the sense of Oriolo and Nakamura (1991). Note also that the last component of the vector of Coriolis and centrifugal terms vanishes by virtue of the particular choice of generalized coordinates.

Through the virtual work principle, the original generalized forces τ are recovered from F_a as

$$\tau = \begin{bmatrix} \tau_1 \\ \vdots \\ \tau_{n-1} \end{bmatrix} = \begin{bmatrix} I_{(n-3) \times (n-3)} \\ 0_{2 \times (n-3)} \end{bmatrix} \begin{bmatrix} F_1 \\ \vdots \\ F_{n-3} \end{bmatrix} + J^T(q_1, \dots, q_{n-1}) \begin{bmatrix} F_x \\ F_y \end{bmatrix}, \quad (8)$$

being J the $2 \times (n - 1)$ Jacobian matrix of the direct kinematics function k in

$$\begin{bmatrix} x \\ y \end{bmatrix} = k(q_1, \dots, q_{n-1}). \quad (9)$$

3.1. Partial Feedback Linearization

To make the analysis independent from the nature of the first $n - 1$ joints, we preliminarily perform a partial linearization of eq. (7) via a globally defined static feedback. As in the computed torque method, the idea is to reduce the dynamics of the active joints to $n - 1$ chains of double integrators, so that they can be controlled via acceleration inputs. To this end, one isolates $\ddot{\theta}$ from the passive joint dynamics, i.e., the last row of the vector equation (7), and plugs it into the active joint dynamics to compute the expression of \ddot{q}_a . Equating the latter to a generalized acceleration vector $a = (a_1, \dots, a_{n-3}, a_x, a_y)$, the partially linearizing static feedback is easily obtained in the form

$$F_a = \hat{B}_a(q)a + c_a(q, \dot{q}) + \hat{g}_a(q). \quad (10)$$

In eq. (10), the $(n - 1) \times (n - 1)$ matrix

$$\hat{B}_a(q) = B_a(q_a) - \frac{m_n^2 d_n^2}{I_n + m_n d_n^2} \begin{bmatrix} 0_{(n-3) \times (n-3)} & | & 0_{(n-3) \times 2} \\ \hline 0_{2 \times (n-3)} & | & \begin{matrix} s^2\theta & -s\theta c\theta \\ -s\theta c\theta & c^2\theta \end{matrix} \end{bmatrix}$$

is always nonsingular, being the Schur complement of diagonal element b_{nn} of the positive definite inertia matrix B , while

$$\hat{g}_a(q) = g_a(q_a) - \frac{g_0 m_n d_n c\theta}{I_n + m_n d_n^2} \begin{bmatrix} 0_{n-3} \\ -m_n d_n s\theta \\ m_n d_n c\theta \end{bmatrix}.$$

Putting together eqs. (7) and (10), the complete closed-loop system becomes

$$\begin{aligned} \ddot{q}_1 &= a_1 \\ &\vdots \\ \ddot{q}_{n-3} &= a_{n-3} \\ \ddot{x} &= a_x \\ \ddot{y} &= a_y \\ \ddot{\theta} &= \frac{1}{K} (s\theta a_x - c\theta (a_y + g_0)), \end{aligned} \quad (11)$$

where $K = (I_n + m_n d_n^2)/m_n d_n$ is precisely the distance of the center of percussion (CP) of the last link from its base (see Figure 1). If uniform mass distribution is assumed, it is $K = 2\ell_n/3$ (ℓ_n is the length of the n th link).

Equations (11) show that the dynamics of the coordinates q_i , $i = 1, \dots, n - 3$, among the actuated degrees of freedom is now completely decoupled from the dynamics of the remaining coordinates (x, y, θ) . In particular, if a reconfiguration task is to be executed, each q_i can be driven independently to its desired value by choosing a_i , for $i = 1, \dots, n - 3$, as an open-loop command or a linear feedback law. Therefore, from now on we shall set $n = 3$ and consider only the core

of the problem, namely trajectory planning and control for the variables x , y , and θ . Accordingly, after partial feedback linearization (PFL) we shall proceed only with the equations

$$\begin{aligned}\ddot{x} &= a_x \\ \ddot{y} &= a_y \\ \ddot{\theta} &= \frac{1}{K} (s\theta a_x - c\theta (a_y + g_0)).\end{aligned}\quad (12)$$

We remark that when the robot motion occurs on a horizontal plane ($\psi = 90^\circ$) then $g_0 = 0$.

4. Design of a Linearizing Dynamic Feedback

We show here that the robot dynamic model in the form (12) can be transformed into a linear controllable system by means of nonlinear dynamic feedback and change of coordinates. To this end, we use the linearization algorithm mentioned in Section 2, starting from the choice of the CP position as system output. The algorithm will provide a singular decoupling matrix in an intermediate step, which would then require the addition of an integrator on a single input channel. The only adaptation to the general algorithm consists in adding directly two integrators at a time in this dynamic extension step, in view of the second-order nature of the mechanical system.

Define the cartesian position of the CP of the last link as output:

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} x \\ y \end{bmatrix} + K \begin{bmatrix} c\theta \\ s\theta \end{bmatrix}.\quad (13)$$

Differentiation of eq. (13) yields

$$\begin{bmatrix} \dot{y}_1 \\ \dot{y}_2 \end{bmatrix} = \begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} + K\dot{\theta} \begin{bmatrix} -s\theta \\ c\theta \end{bmatrix}\quad (14)$$

and

$$\begin{bmatrix} \ddot{y}_1 \\ \ddot{y}_2 \end{bmatrix} = \begin{bmatrix} c^2\theta & s\theta c\theta \\ s\theta c\theta & s^2\theta \end{bmatrix} \begin{bmatrix} a_x \\ a_y \end{bmatrix} - R(\theta) \begin{bmatrix} K\dot{\theta}^2 \\ g_0 c\theta \end{bmatrix},$$

where eq. (12) has been used and $R(\theta)$ is the matrix associated to a planar rotation of an angle θ . Since the matrix multiplying the acceleration vector (a_x, a_y) is singular, we define the invertible feedback transformation¹

$$\begin{bmatrix} a_x \\ a_y \end{bmatrix} = R(\theta) \begin{bmatrix} \xi + K\dot{\theta}^2 \\ \sigma_2 \end{bmatrix},\quad (15)$$

where ξ and σ_2 are two auxiliary input variables. Note that σ_2 is the linear acceleration of the base of the last link along the normal to its axis (see Figure 2). Moreover, as a result of (15), we have

$$\begin{bmatrix} \ddot{y}_1 \\ \ddot{y}_2 \end{bmatrix} = R(\theta) \begin{bmatrix} \xi \\ -g_0 c\theta \end{bmatrix},\quad (16)$$

and therefore ξ is the linear acceleration of the CP along the last link axis (see Figure 2).

In order to further differentiate the output, while avoiding differentiation of the input ξ , we add two integrators on the first channel

$$\dot{\xi} = \eta\quad (17)$$

$$\dot{\eta} = \sigma_1,\quad (18)$$

with σ_1 the new auxiliary input in place of ξ . From eq. (16), the third output derivative is

$$\begin{bmatrix} y_1^{[3]} \\ y_2^{[3]} \end{bmatrix} = R(\theta) \begin{bmatrix} \eta + g_0 c\theta \dot{\theta} \\ \xi \dot{\theta} + g_0 s\theta \dot{\theta} \end{bmatrix},\quad (19)$$

where we have used eq. (17) and the property

$$\dot{R}(\theta) = R(\theta)S(\dot{\theta}) = \begin{bmatrix} c\theta & -s\theta \\ s\theta & c\theta \end{bmatrix} \begin{bmatrix} 0 & -\dot{\theta} \\ \dot{\theta} & 0 \end{bmatrix}.$$

From eq. (19), η can be interpreted as the component not due to gravity of the linear jerk of the CP along the last link axis. Finally, being $\ddot{\theta} = -(\sigma_2 + g_0 c\theta)/K$, the fourth output derivative is computed as

$$\begin{aligned}\begin{bmatrix} y_1^{[4]} \\ y_2^{[4]} \end{bmatrix} &= R(\theta) \left\{ \begin{bmatrix} 1 & -\frac{g_0}{K}c\theta \\ 0 & -\frac{\xi + g_0 s\theta}{K} \end{bmatrix} \begin{bmatrix} \sigma_1 \\ \sigma_2 \end{bmatrix} \right. \\ &\quad \left. + \begin{bmatrix} -(2g_0 s\theta + \xi)\dot{\theta}^2 - \frac{g_0^2}{K}c^2\theta \\ 2(g_0 c\theta \dot{\theta} + \eta)\dot{\theta} - \frac{g_0}{K}(\xi + g_0 s\theta)c\theta \end{bmatrix} \right\} \\ &\triangleq R(\theta) \{A(\theta, \xi)\sigma + b(\theta, \dot{\theta}, \xi, \eta)\}.\end{aligned}$$

Under the *regularity assumption* that matrix $A(\theta, \xi)$ is non-singular or, equivalently, that

$$\rho \triangleq \xi + g_0 s\theta \neq 0,\quad (20)$$

the inversion-based control

$$\sigma = A^{-1}(\theta, \xi) \left(R^T(\theta)v - b(\theta, \dot{\theta}, \xi, \eta) \right),\quad (21)$$

with $v = (v_1, v_2)$ as new input vector, yields

$$\begin{bmatrix} y_1^{[4]} \\ y_2^{[4]} \end{bmatrix} = \begin{bmatrix} v_1 \\ v_2 \end{bmatrix},\quad (22)$$

i.e., two decoupled chains of four input-output integrators.

Since the dimension (i.e., 6) of the robot state (q, \dot{q}) plus the dimension (i.e., 2) of the compensator state (ξ, η) equals the sum of the relative degrees ($4 + 4 = 8$) of the two outputs in eq. (22), exact state linearization has been achieved. This implies that the system has *trivial* zero dynamics with respect to the chosen output (13). Equivalently, one may say that the

1. This step is different from our original approach in De Luca and Oriolo (2000a, b) and leads to a simpler controller.

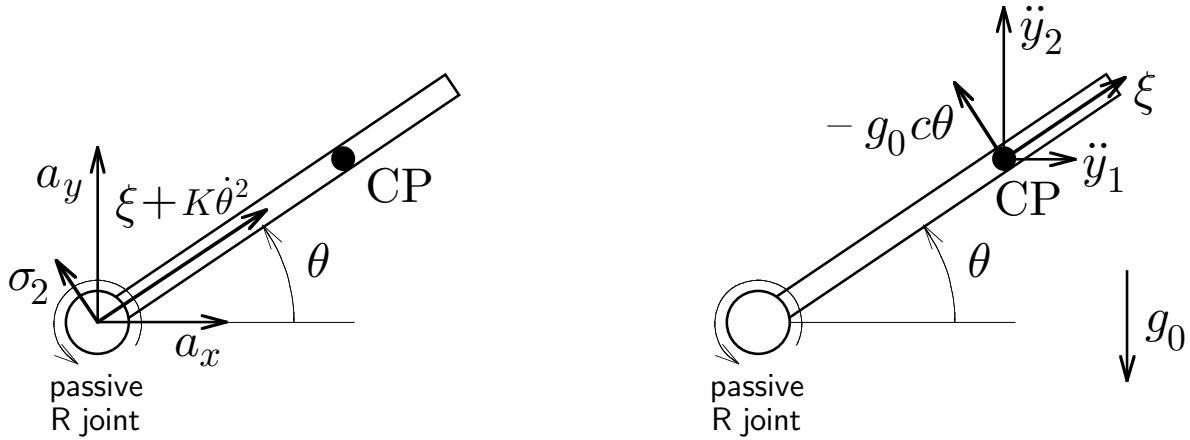


Fig. 2. Physical interpretation of eqs. (15) (left) and (16) (right).

CP position of the last link is a *flat* output (Fliess et al. 1995). As already mentioned, this property has already been used in various contexts, e.g., the control of a VTOL aircraft (Martin, Devasia, and Paden 1996), the trajectory planning for a planar rigid body with a single thruster (Faiz and Agrawal 1998), and within a general study of flatness in systems with a single degree of underactuation (Rathinam and Murray 1998).

For the PFL system (12), the linearizing dynamic controller is obtained combining eqs. (15), (17), (18), and (21). The block diagram of the controller is shown in Figure 3. The initialization of the compensator state at time $t = 0$, i.e., $(\xi(0), \eta(0))$, is arbitrary (more on this in Section 5).

As a byproduct of the linearizing algorithm, a new set of state coordinates can be defined consisting of the output function (13), together with its first-, second-, and third-order derivatives (respectively, eqs. (14), (16), and (19)). The inverse transformation from these linearizing coordinates to the robot and compensator states can be computed in closed form as:

$$\begin{aligned}
 \theta &= \text{ATAN2} \{ \text{sign}(\rho)(\ddot{y}_2 + g_0), \text{sign}(\rho)\ddot{y}_1 \} \\
 \xi &= \ddot{y}_1 c \theta + \ddot{y}_2 s \theta \\
 \dot{\theta} &= \frac{y_2^{[3]} c \theta - y_1^{[3]} s \theta}{\rho} \\
 \eta &= y_1^{[3]} c \theta + y_2^{[3]} s \theta - g_0 c \theta \dot{\theta} \\
 \begin{bmatrix} x \\ y \end{bmatrix} &= \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} - K \begin{bmatrix} c \theta \\ s \theta \end{bmatrix} \\
 \begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} &= \begin{bmatrix} \dot{y}_1 \\ \dot{y}_2 \end{bmatrix} - K \dot{\theta} \begin{bmatrix} -s \theta \\ c \theta \end{bmatrix},
 \end{aligned} \tag{23}$$

where $\text{ATAN2}(y, x)$ is the four-quadrant inverse tangent function. The coordinate transformation is well defined if and only if the regularity condition (20) holds.

From eq. (16), it is easy to show that

$$\rho^2 = \ddot{y}_1^2 + (\ddot{y}_2 + g_0)^2, \tag{24}$$

and thus the regularity condition can be checked based only on the output trajectory, without actually computing θ and ξ . Physically, $\rho \neq 0$ means that the linear acceleration ξ of the center of percussion of the last link along the link axis should not cancel the projection of the gravitational acceleration along the same axis.

We finally remark that all the previous derivations are valid for any value of g_0 and thus, in particular, also for robot motion in an horizontal plane ($g_0 = 0$); in this case, we have $\rho = \xi$ and the regularity condition becomes simply $\xi \neq 0$. This means that with our scheme pure rotational motion around the CP is not allowed in the absence of gravity.

5. Trajectory Planning and Control

We now consider the problem of planning a feasible trajectory joining an arbitrary initial robot state with a desired final state in a given time. In view of the existence of a non-integrable differential constraint on the underactuated robot, this problem is an instance of *nonholonomic trajectory planning*. The dynamic feedback linearization technique of Section 4 suggests a simple and systematic approach to its solution. In fact, planning a feasible motion for the underactuated robot is equivalent to planning a state-to-state transfer for the equivalent linear representation (22). The latter can be formulated as an *interpolation* problem using smooth parametric functions $y_1(s)$ and $y_2(s)$, with a timing law $s = s(t)$. For simplicity, we shall directly generate trajectories $y_1(t)$ and $y_2(t)$.

In particular, assume that at time $t = 0$ the robot starts from a generic state $(q_s, \dot{q}_s) = (x_s, y_s, \theta_s, \dot{x}_s, \dot{y}_s, \dot{\theta}_s)$ to reach a goal state $(q_g, \dot{q}_g) = (x_g, y_g, \theta_g, \dot{x}_g, \dot{y}_g, \dot{\theta}_g)$ at time $t = T$. In view of the transposition of the planning problem to the feedback linearized system (22), one needs to associate to (q_s, \dot{q}_s) and (q_g, \dot{q}_g) the appropriate boundary conditions for the new state variables, i.e., y_1, y_2 , and their derivatives up to the third order,

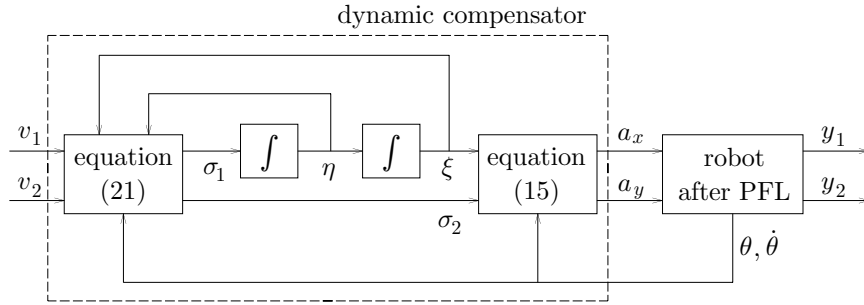


Fig. 3. Scheme of the linearizing dynamic controller.

at time $t = 0$

$$\begin{bmatrix} y_1(0) \\ \dot{y}_1(0) \\ \ddot{y}_1(0) \\ y_1^{[3]}(0) \end{bmatrix} = \begin{bmatrix} y_{1s} \\ \dot{y}_{1s} \\ \ddot{y}_{1s} \\ y_{1s}^{[3]} \end{bmatrix}, \quad \begin{bmatrix} y_2(0) \\ \dot{y}_2(0) \\ \ddot{y}_2(0) \\ y_2^{[3]}(0) \end{bmatrix} = \begin{bmatrix} y_{2s} \\ \dot{y}_{2s} \\ \ddot{y}_{2s} \\ y_{2s}^{[3]} \end{bmatrix},$$

and at time $t = T$

$$\begin{bmatrix} y_1(T) \\ \dot{y}_1(T) \\ \ddot{y}_1(T) \\ y_1^{[3]}(T) \end{bmatrix} = \begin{bmatrix} y_{1g} \\ \dot{y}_{1g} \\ \ddot{y}_{1g} \\ y_{1g}^{[3]} \end{bmatrix}, \quad \begin{bmatrix} y_2(T) \\ \dot{y}_2(T) \\ \ddot{y}_2(T) \\ y_2^{[3]}(T) \end{bmatrix} = \begin{bmatrix} y_{2g} \\ \dot{y}_{2g} \\ \ddot{y}_{2g} \\ y_{2g}^{[3]} \end{bmatrix}.$$

To this end, we use eqs. (13)–(14), (16) and (19), in which $\xi(0) = \xi_s$, $\xi(T) = \xi_g$, $\eta(0) = \eta_s$, and $\eta(T) = \eta_g$ are still free parameters.

A straightforward solution to the interpolation problem is to generate trajectories as polynomials of seventh degree:

$$y_i(t) = \sum_{j=0}^7 a_{ij} \lambda^j, \quad i = 1, 2, \quad (25)$$

with normalized time $\lambda = t/T$. Dropping for compactness the output index i , the expressions of the coefficients a_j are:

$$\begin{aligned} a_0 &= y_s \\ a_1 &= \dot{y}_s T \\ a_2 &= \frac{1}{2} \ddot{y}_s T^2 \\ a_3 &= \frac{1}{6} y_s^{[3]} T^3 \\ a_4 &= 35(y_g - y_s) - (20\dot{y}_s + 15\dot{y}_g)T \\ &\quad - (5\ddot{y}_s - \frac{5}{2}\ddot{y}_g)T^2 - (\frac{2}{3}y_s^{[3]} + \frac{1}{6}y_g^{[3]})T^3 \\ a_5 &= -84(y_g - y_s) + (45\dot{y}_s + 39\dot{y}_g)T \\ &\quad + (10\ddot{y}_s - 7\ddot{y}_g)T^2 + (y_s^{[3]} + \frac{1}{2}y_g^{[3]})T^3 \\ a_6 &= 70(y_g - y_s) - (36\dot{y}_s + 34\dot{y}_g)T \\ &\quad - (\frac{15}{2}\ddot{y}_s - \frac{13}{2}\ddot{y}_g)T^2 - (\frac{2}{3}y_s^{[3]} + \frac{1}{2}y_g^{[3]})T^3 \end{aligned}$$

$$\begin{aligned} a_7 &= -20(y_g - y_s) + 10(\dot{y}_s + \dot{y}_g)T + 2(\ddot{y}_s - \ddot{y}_g)T^2 \\ &\quad + \frac{1}{6}(y_s^{[3]} + y_g^{[3]})T^3. \end{aligned}$$

The robot+compensator state trajectory associated to the linearizing output trajectory (25) that solves the planning problem is obtained by pure algebraic computations using eqs. (23). Moreover, the open-loop commands that realize this trajectory are

$$\begin{aligned} v_i(t) &= \frac{1}{T^4} (840a_{i7}\lambda^3 + 360a_{i6}\lambda^2 + 120a_{i5}\lambda + 24a_{i4}), \\ i &= 1, 2, \end{aligned}$$

which represent the nominal inputs to system (22) or, equivalently, to the dynamic feedback compensator (see Figure 3), which in turn produces the acceleration inputs (a_x, a_y) . Since $n = 3$, eq. (7) with $a = (a_x, a_y)$ can be directly² used to generate F_a . Finally, the original generalized forces τ driving the robot are obtained from eq. (8).

The selection of initial and final compensator states (ξ_s, η_s) and (ξ_g, η_g) affects the boundary conditions, and thus the generated motion inside the chosen class of interpolating functions. In particular, the compensator states should be chosen so as to avoid the singularity $\rho = 0$ during the motion. This problem will be considered in detail in the next two subsections, referring separately to the cases of robot motion in the absence ($g_0 = 0$) and presence ($g_0 = 9.81 \cdot \cos \psi \neq 0$) of gravity.

First, however, we briefly discuss the problem of *tracking* the generated trajectories. The feedforward commands resulting from a trajectory planning algorithm yield the desired robot reconfiguration only in nominal conditions, i.e., initial state matched with the desired reference trajectory and absence of disturbances during motion. Feedback control must be used to alleviate the effects of an initial state error and of different kinds of perturbations. The presented dynamic feedback linearization approach leads to a simple (linear) solution

2. For $n > 3$, we need to know also the accelerations (a_1, \dots, a_{n-3}) , which are computed according to eq. (11) from the trajectories independently assigned to (q_1, \dots, q_{n-3}) .

to the tracking problem, with the appealing feature of global exponential convergence.

The design of a tracking controller is performed on the equivalent system (22). Given a desired smooth trajectory $(y_{1d}(t), y_{2d}(t))$ for the CP (e.g., generated through eq. (25)), we let

$$v_i = y_{id}^{[4]} + F_i \begin{bmatrix} y_{id}^{[3]} - y_i^{[3]} \\ \ddot{y}_{id} - \ddot{y}_i \\ \dot{y}_{id} - \dot{y}_i \\ y_{id} - y_i \end{bmatrix}, \quad i = 1, 2, \quad (26)$$

where the gain matrices $F_i = [f_{i3} \ f_{i2} \ f_{i1} \ f_{i0}]$ can be chosen so as to assign arbitrary stable eigenvalues to the tracking dynamics. A desirable side effect of exponential convergence of the linearizing output errors is the *predictability* of the mechanism motion, in view of the fact that the CP point is fixed on the passive link. In particular, the motion of the latter is contained in a “tube” of width $2K$ around the CP.

The actual states $(y_1, y_2, \dots, y_1^{[3]}, y_2^{[3]})$ in eq. (26) are computed on-line from the measured joint positions and velocities of the robot and from the available dynamic compensator state (ξ, η) , by using the forward kinematics (9), its Jacobian matrix, and transformations (13)–(14), (16), and (19). Provided that the regularity condition $\rho \neq 0$ is satisfied during motion (i.e., along the actual state trajectory of the closed-loop system), the tracking error converges to zero with the prescribed exponential rate, independently from the presence or absence of gravity.

5.1. The Zero Gravity Case

Consider the case of a rest-to-rest motion ($\dot{x}_s = \dot{y}_s = \dot{\theta}_s = \dot{x}_g = \dot{y}_g = \dot{\theta}_g = 0$). From eqs. (13)–(14), (16), and (19), we obtain the boundary conditions for the first output

$$\begin{bmatrix} y_{1s} \\ \dot{y}_{1s} \\ \ddot{y}_{1s} \\ y_{1s}^{[3]} \end{bmatrix} = \begin{bmatrix} x_s + K c\theta_s \\ 0 \\ \xi_s c\theta_s \\ \eta_s c\theta_s \end{bmatrix}, \quad \begin{bmatrix} y_{1g} \\ \dot{y}_{1g} \\ \ddot{y}_{1g} \\ y_{1g}^{[3]} \end{bmatrix} = \begin{bmatrix} x_g + K c\theta_g \\ 0 \\ \xi_g c\theta_g \\ \eta_g c\theta_g \end{bmatrix},$$

and for the second output

$$\begin{bmatrix} y_{2s} \\ \dot{y}_{2s} \\ \ddot{y}_{2s} \\ y_{2s}^{[3]} \end{bmatrix} = \begin{bmatrix} y_s + K s\theta_s \\ 0 \\ \xi_s s\theta_s \\ \eta_s s\theta_s \end{bmatrix}, \quad \begin{bmatrix} y_{2g} \\ \dot{y}_{2g} \\ \ddot{y}_{2g} \\ y_{2g}^{[3]} \end{bmatrix} = \begin{bmatrix} y_g + K s\theta_g \\ 0 \\ \xi_g s\theta_g \\ \eta_g s\theta_g \end{bmatrix}.$$

Being in this case $\rho = \xi$, in order to avoid the singularity ξ_s and ξ_g should be nonzero and of the same sign. This condition is however only necessary, for ξ may nevertheless cross zero during the motion. The occurrence of this event may be predicted in advance on the basis of the following analysis. Equation (24) becomes in this case

$$\xi^2 = \ddot{y}_1^2 + \ddot{y}_2^2,$$

so that the singularity is met if and only if the two output accelerations (both fifth-order polynomials) vanish at a same time instant $\bar{t} \in [0, T]$. Elementary algebra states that this happens when the *resultant matrix* of the two polynomials \ddot{y}_1 and \ddot{y}_2 is singular. In our case, the regularity condition takes the following form:

$$\begin{aligned} &\xi_g \sin(\theta_g - \theta_s) T^2 + 10 [(y_{1s} - y_{1g}) \sin \theta_s \\ &\quad - (y_{2s} - y_{2g}) \cos \theta_s] \neq 0, \end{aligned} \quad (27)$$

provided that either

$$-10(y_{1g} - y_{1s}) + (\xi_s \cos \theta_s - \xi_g \cos \theta_g) T^2 \neq 0 \quad (28)$$

or

$$-10(y_{2g} - y_{2s}) + (\xi_s \sin \theta_s - \xi_g \sin \theta_g) T^2 \neq 0. \quad (29)$$

It is relatively easy to show that one of the two inequalities (28) and (29) can be always satisfied by an appropriate choice of (ξ_s, ξ_g) . The same is true for inequality (27) with the only notable exception of straight line transfers, i.e.,

$$\theta_s = \theta_g \pmod{180^\circ} \quad \text{and} \quad \frac{y_{2s} - y_{2g}}{y_{1s} - y_{1g}} = \tan \theta_s.$$

In the latter case, the motion task can be split in two phases by the addition of an intermediate configuration (outside the straight line).

5.1.1. Numerical Results

To illustrate the performance of the planner, we present a typical result obtained for the rest-to-rest task

$$\begin{bmatrix} x_s \\ y_s \\ \theta_s \end{bmatrix} = \begin{bmatrix} 0.5 \text{ m} \\ 1 \text{ m} \\ 0^\circ \end{bmatrix} \longrightarrow \begin{bmatrix} x_g \\ y_g \\ \theta_g \end{bmatrix} = \begin{bmatrix} 1.5 \text{ m} \\ 2 \text{ m} \\ 45^\circ \end{bmatrix},$$

with $T = 10$ s, $K = 2/3$ ($\ell_3 = 1$ m), $\xi_s = \xi_g = -0.1$ m/s², and $\eta_s = \eta_g = 0$.

The cartesian motion of the third link corresponding to the planned trajectory for the CP is given in Figure 4. The motion of a complete 3R arm (with $\ell_1 = \ell_2 = 1.5$ m), obtained by kinematic inversion, is given in Figure 5. The nominal torques in Figure 6 are obtained³ from the inverse dynamics (6) with the following mass data (links are thin rods of uniform mass): $m_1 = 10$, $m_2 = 5$, $m_3 = 1$ (kg). Note the high joint velocities and peak torques around $t = 6$ s, corresponding to a rapid rotation of the third link approximately around its CP—and thus to a decrease of the regularity index ρ .

We have also simulated the tracking controller along the rest-to-rest trajectory generated above. The robot starts from

3. This is a purely algebraic computation. Alternatively, the torques could have been computed using eq. (10) and (8) in conjunction with a simulation of the robot dynamics.

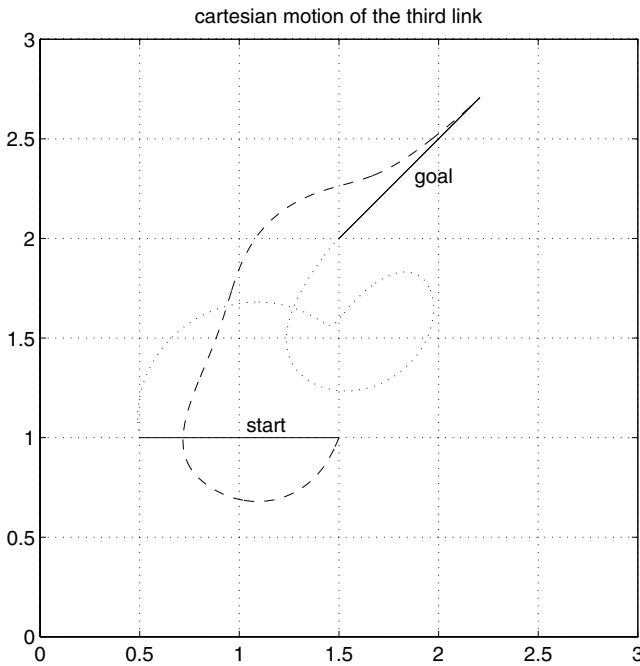


Fig. 4. Rest-to-rest planning in zero gravity: Third link motion.

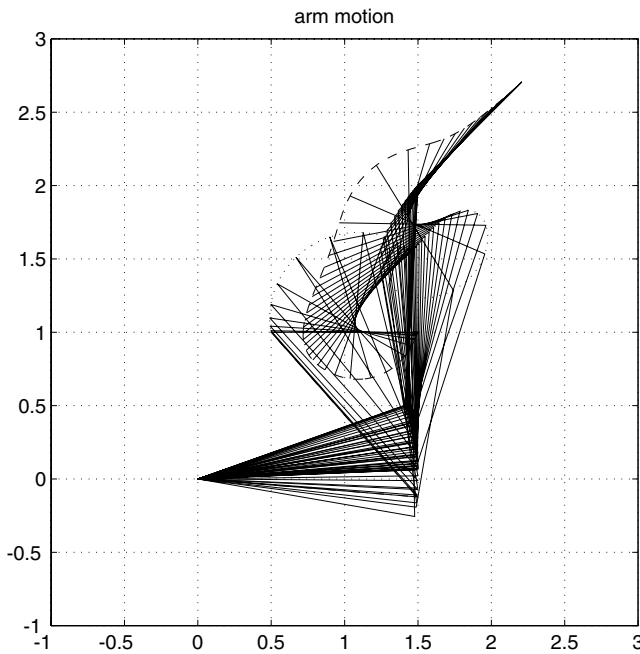


Fig. 5. Rest-to-rest planning in zero gravity: 3R arm motion.

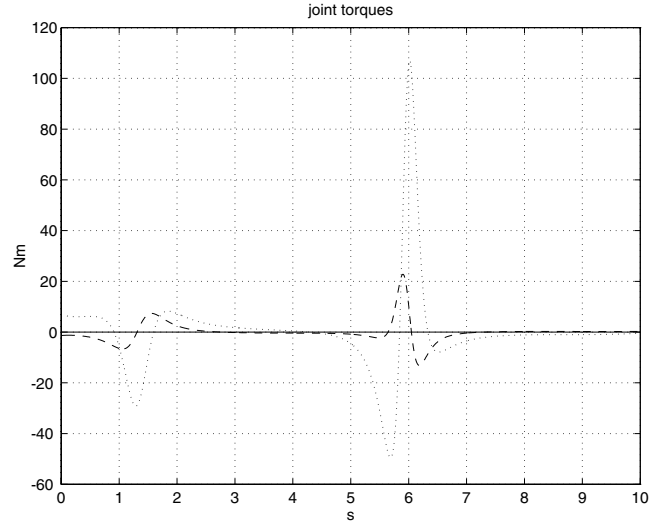


Fig. 6. Rest-to-rest planning in zero gravity: τ_1 (\cdots), τ_2 ($---$).

the off-path configuration $x(0) = 0.5$ m, $y(0) = 0.9$ m, $\theta(0) = 15^\circ$, and with zero initial velocity. The compensator state is initialized at the same value used for the nominal trajectory generation: $\xi(0) = -0.1$ m/s², $\eta(0) = 0$. The eigenvalues of the closed-loop tracking error dynamics have been all placed in -2 , for both components of the output trajectory. The resulting gain matrices are

$$F_1 = F_2 = \begin{bmatrix} 8 & 24 & 32 & 16 \end{bmatrix}.$$

The actual motion of the third link is shown in Figure 8 (to be compared with Figure 4). The evolution of the CP position errors $e_i = y_{id} - y_i$ ($i = 1, 2$) in Figure 7 shows the prescribed exponential rate of decay.

5.2. The Nonzero Gravity Case

To analyze this case, we consider first a rest-to-rest motion between inverted equilibria ($\theta_s = \theta_g = 90^\circ$, with zero initial and final velocities) in the vertical plane ($\psi = 0$). From eqs. (13)–(14), (16), and (19), we obtain the boundary conditions for the first output

$$\begin{bmatrix} y_{1s} \\ \dot{y}_{1s} \\ \ddot{y}_{1s} \\ y_{1s}^{[3]} \end{bmatrix} = \begin{bmatrix} x_s \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad \begin{bmatrix} y_{1g} \\ \dot{y}_{1g} \\ \ddot{y}_{1g} \\ y_{1g}^{[3]} \end{bmatrix} = \begin{bmatrix} x_g \\ 0 \\ 0 \\ 0 \end{bmatrix},$$

and for the second output

$$\begin{bmatrix} y_{2s} \\ \dot{y}_{2s} \\ \ddot{y}_{2s} \\ y_{2s}^{[3]} \end{bmatrix} = \begin{bmatrix} y_s + K \\ 0 \\ \xi_s \\ \eta_s \end{bmatrix}, \quad \begin{bmatrix} y_{2g} \\ \dot{y}_{2g} \\ \ddot{y}_{2g} \\ y_{2g}^{[3]} \end{bmatrix} = \begin{bmatrix} y_g + K \\ 0 \\ \xi_g \\ \eta_g \end{bmatrix}.$$

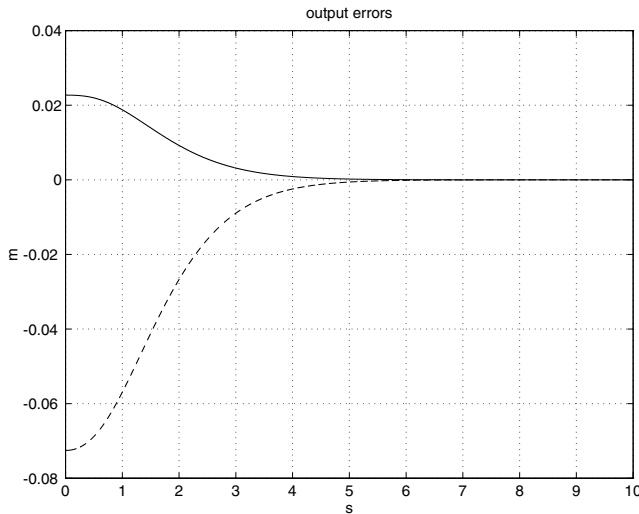


Fig. 7. Trajectory tracking: e_1 (—), e_2 (---).

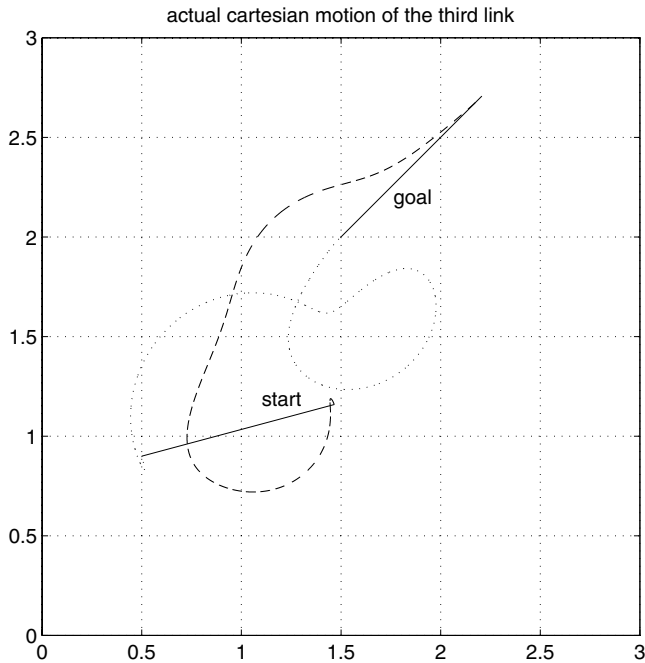


Fig. 8. Trajectory tracking: Actual cartesian motion of the third link.

Instead, for a swing-up maneuver between the downward equilibrium $\theta_s = -90^\circ$ and the upward (inverted) equilibrium $\theta_g = 90^\circ$ of the third link, we have the same boundary conditions for the first output, while for the second output they are replaced by

$$\begin{bmatrix} y_{2s} \\ \dot{y}_{2s} \\ \ddot{y}_{2s} \\ y_{2s}^{[3]} \end{bmatrix} = \begin{bmatrix} y_s - K \\ 0 \\ -\xi_s \\ -\eta_s \end{bmatrix}, \quad \begin{bmatrix} y_{2g} \\ \dot{y}_{2g} \\ \ddot{y}_{2g} \\ y_{2g}^{[3]} \end{bmatrix} = \begin{bmatrix} y_g + K \\ 0 \\ \xi_g \\ \eta_g \end{bmatrix}.$$

We note that, since the singularity occurs now for $\rho = \xi + g_0 s \theta = 0$, zero values or opposite signs for ξ_s and ξ_g are not a priori forbidden. Similarly to Section 5.1, the analysis of potential occurrence of singularities can be performed on the basis of the resultant matrix of the two polynomials \dot{y}_1 and $\ddot{y}_2 + g_0$.

In particular, for the swing-up maneuver, when the x coordinate of the start and goal configurations are the same ($x_s = x_g$) the interpolation scheme (25) will generate $y_1(t) \equiv 0$, not allowing the third link to undergo the required rotation of 180° and leading the system to the singularity $\rho = 0$. In this case, an intermediate state ($x_m \neq x_s, y_m, \theta_m, \dot{x}_m, \dot{y}_m, \dot{\theta}_m, \xi_m, \eta_m$) should be added, to be reached at some instant $T_m \in (0, T)$. For example, at the horizontal position $\theta_m = 0$ for the third link, we would have as intermediate boundary conditions for the two outputs

$$\begin{bmatrix} y_{1m} \\ \dot{y}_{1m} \\ \ddot{y}_{1m} \\ y_{1m}^{[3]} \end{bmatrix} = \begin{bmatrix} x_m + K \\ \dot{x}_m \\ \xi_m \\ \eta_m + g_0 \dot{\theta}_m \end{bmatrix}, \quad \begin{bmatrix} y_{2m} \\ \dot{y}_{2m} \\ \ddot{y}_{2m} \\ y_{2m}^{[3]} \end{bmatrix} = \begin{bmatrix} y_m \\ \dot{y}_m + K \dot{\theta}_m \\ -g_0 \\ \xi_m \dot{\theta}_m \end{bmatrix},$$

so that trajectory planning is split in two similar interpolation problems (called phase I and II).

5.2.1. Numerical Results

The performance of the planner is evaluated on two case studies: a transfer between two inverted equilibria and a swing-up maneuver. We model the third link as before.

Figures 9 and 10 refer to a motion task between the inverted equilibria

$$\begin{bmatrix} x_s \\ y_s \\ \theta_s \end{bmatrix} = \begin{bmatrix} 0.75 \text{ m} \\ 1 \text{ m} \\ 90^\circ \end{bmatrix} \longrightarrow \begin{bmatrix} x_g \\ y_g \\ \theta_g \end{bmatrix} = \begin{bmatrix} 1.75 \text{ m} \\ 1 \text{ m} \\ 90^\circ \end{bmatrix},$$

with $T = 1$ s and $\xi_s = \xi_g = \eta_s = \eta_g = 0$.

From the third link cartesian motion shown in Figure 9, we note that the CP is always kept at the same height. The motion of the same complete 3R arm of Section 5.1, obtained by kinematic inversion, is given in Figure 9. The nominal torques in Figure 10 are obtained from inverse dynamics (6).

Results on the same task executed slower ($T = 5$ s) are shown in Figures 11 and 12. The third link hardly leaves the

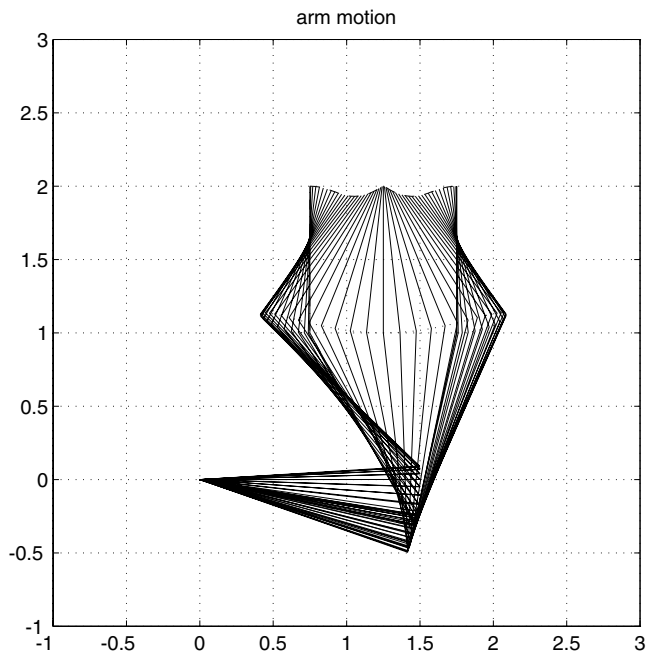


Fig. 9. Transfer between inverted equilibria: 3R arm motion.

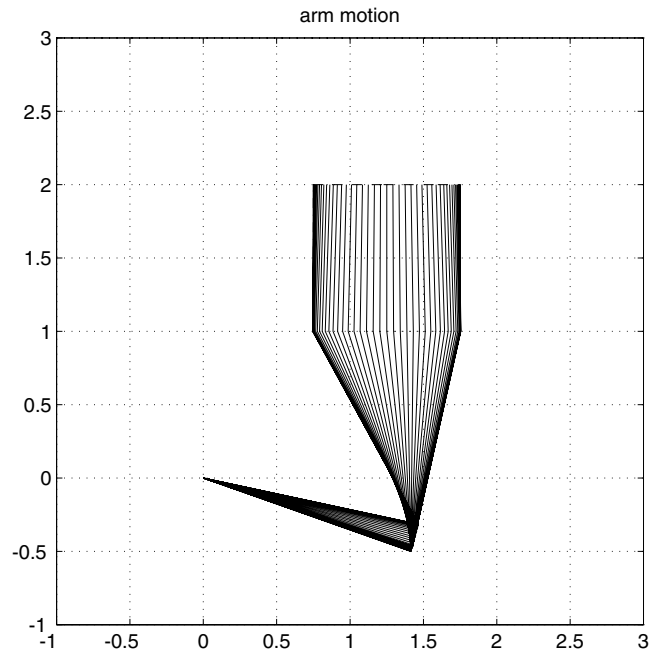


Fig. 11. Slower transfer between inverted equilibria: 3R arm motion.

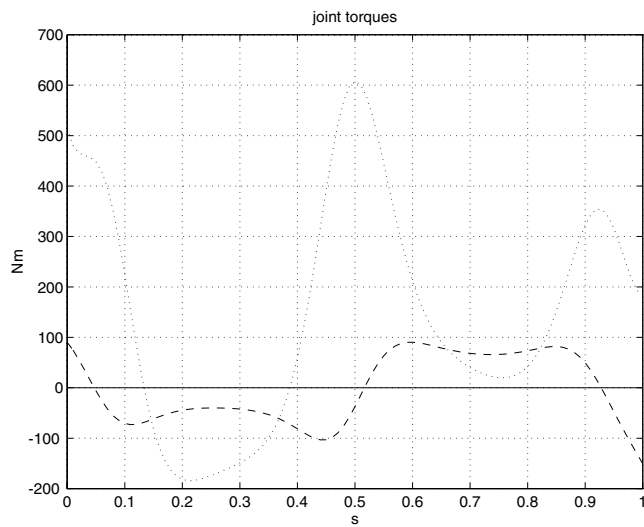


Fig. 10. Transfer between inverted equilibria: Torques τ_1 (—) and τ_2 (---).

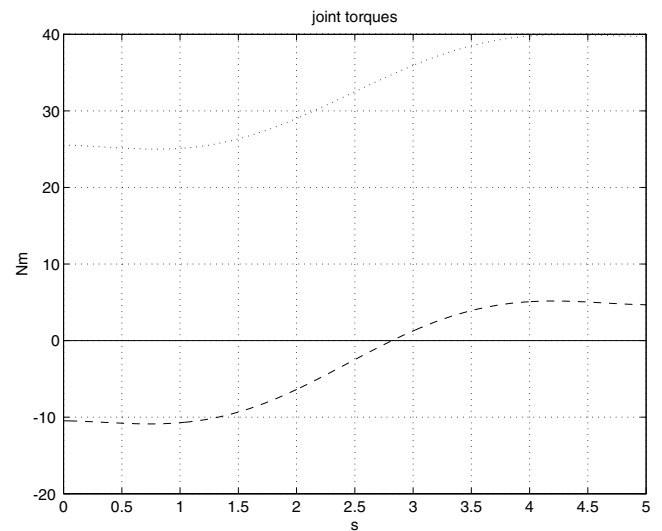


Fig. 12. Slower transfer between inverted equilibria: Torques τ_1 (—) and τ_2 (---).

vertical position, mimicking the natural balancing of a stick during a slow translation. The necessary torques are also quite reduced.

In Figures 13–15, we report the results for the two-phase swing-up maneuver

$$\begin{bmatrix} x_s \\ y_s \\ \theta_s \end{bmatrix} = \begin{bmatrix} 0.75 \text{ m} \\ 1 \text{ m} \\ -90^\circ \end{bmatrix} \rightarrow \begin{bmatrix} x_g \\ y_g \\ \theta_g \end{bmatrix} = \begin{bmatrix} 0.75 \text{ m} \\ 1 \text{ m} \\ 90^\circ \end{bmatrix},$$

with $T = 1.6$ s, $\xi_s = 0$, $\xi_g = -20$ m/s², and $\eta_s = \eta_g = 0$. As intermediate zero-velocity state we used $(x_m, y_m, \theta_m) = (0.75, 1, 0^\circ)$, with $T_m = 1$ s, $\xi_m = -10$ m/s², and $\eta_m = 0$.

Analyzing Figure 13 in detail, we notice in phase I an initial small oscillation, in which the third link builds up kinetic energy, before the counterclockwise rotation to the horizontal position obtained with a slight upward motion. At the beginning of phase II, the third link inverts its angular velocity and executes a large clockwise swing of 270°.

The nominal torques of the 3R robot in Figure 15, obtained for $m_1 = m_2 = 1$, $m_3 = 0.5$ (kg), show a discontinuity at the phase transition instant. This is because the interpolating polynomials (25) can guarantee boundary continuity only up to the third derivative, while the nominal torque depends on the fourth time derivatives of $y_1(t)$ and $y_2(t)$ —see eqs. (21) and (22).

6. Conclusions

The trajectory planning and control problems for the class of n -link planar robots with passive rotational last joint have been solved in the unifying framework of dynamic feedback linearization, both in the absence and presence of gravity. The use of a preliminary partial linearization of the system equations via static state feedback enables one to reduce the problem to its essence, namely planning and controlling the motion of the last link alone, driven by two independent acceleration inputs at its base. The whole approach relies on the basic property that the position of the center of percussion of the last link is a linearizing (or flat) output for the system.

Using the dynamic feedback transformation, the system dynamics is reduced to decoupled input-output chains of integrators. Therefore, planning feasible motions boil down to separate interpolations for the two components of the linearizing output. In particular, rest-to-rest reconfiguration tasks can be always executed with a single-phase trajectory when the robot is moving on a horizontal plane (except for the special case of straight line transfer). Under gravity, this is possible only when the transfer takes place between two upward or downward equilibria. Swing-up maneuvers must be accomplished through a two-phase trajectory.

In any case, during planning it is necessary to avoid the singularity which occurs when the acceleration imposed to the center of percussion along the last link axis vanishes or

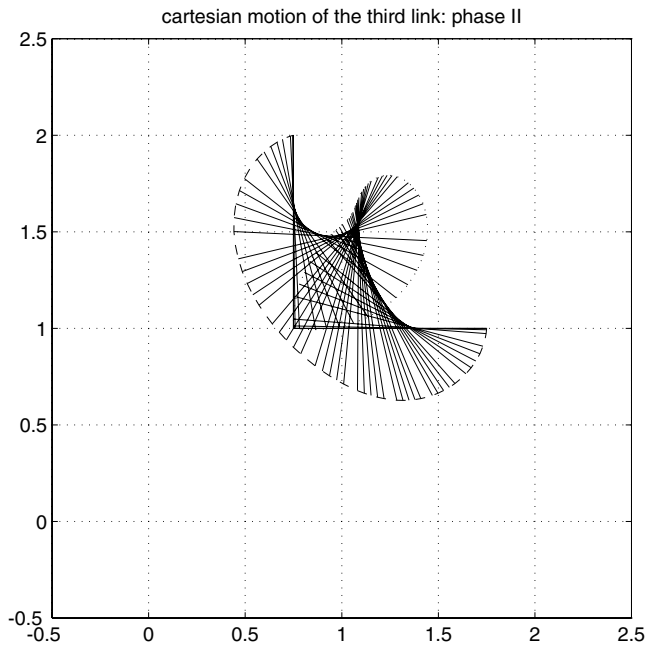
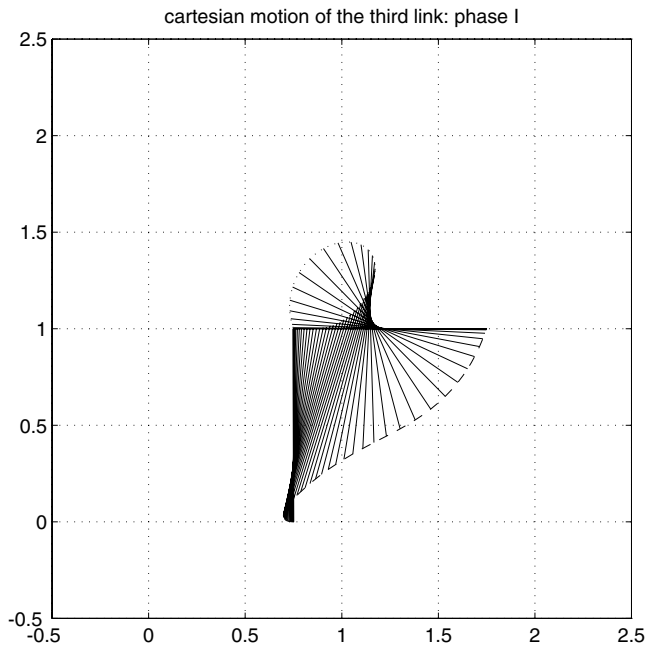


Fig. 13. Swing-up maneuver: Third link motion in phase I (above) and in phase II (below).

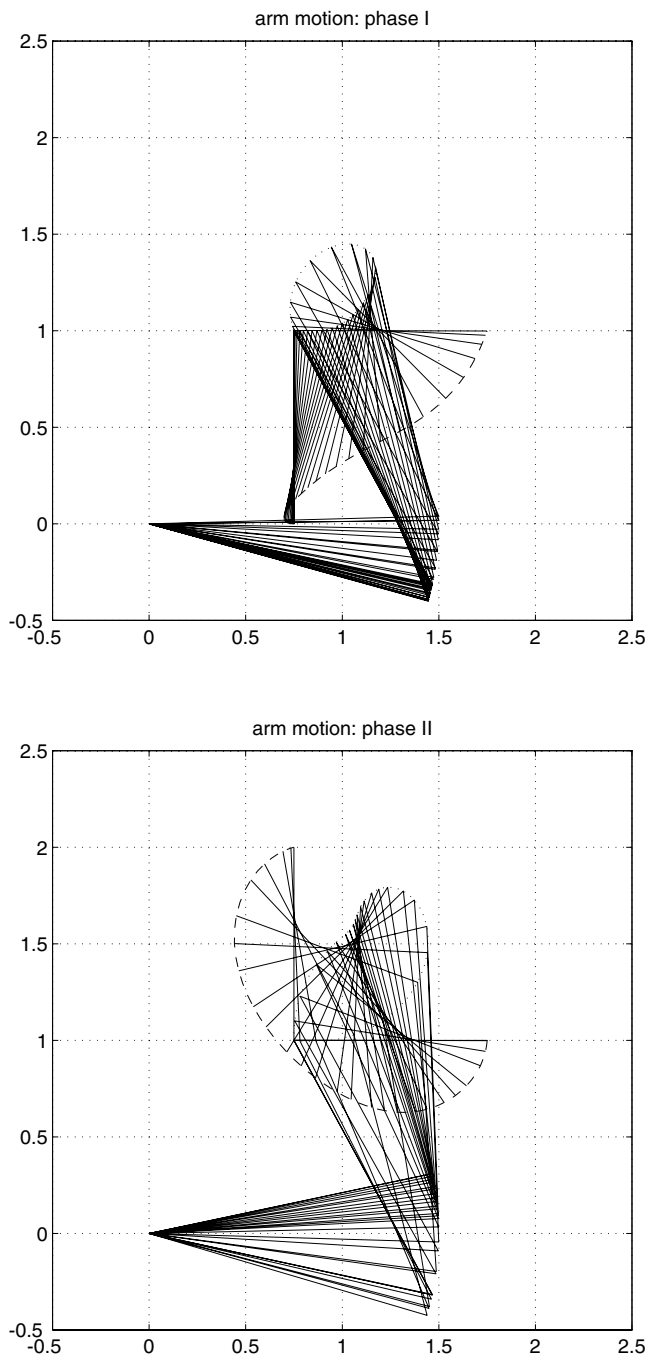


Fig. 14. Swing-up maneuver: 3R arm motion in phase I (above) and in phase II (below).

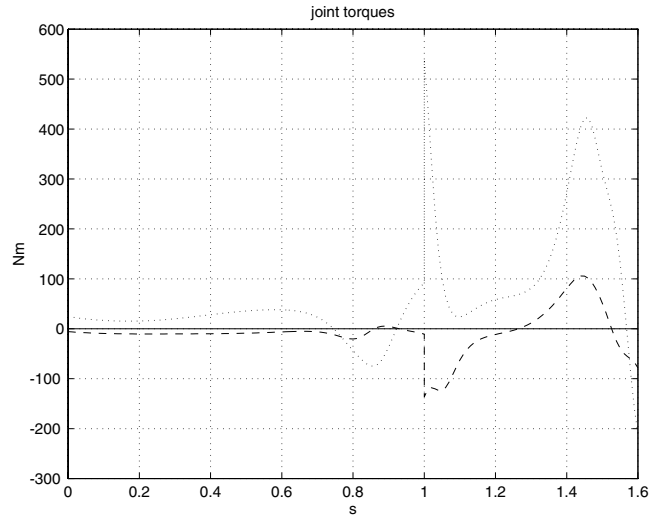


Fig. 15. Swing-up maneuver: Torques τ_1 (—) and τ_2 (---)

cancels the acceleration in the same direction due to gravity, when present. This can be avoided by a proper choice of the boundary conditions for the interpolation (i.e., by selecting appropriate initial and final values for the compensator state), or also by switching these conditions at an intermediate point (i.e., resetting the compensator state).

As for the trajectory tracking problem, an additional outcome of the dynamic feedback linearization approach is the possibility of designing exponentially stabilizing controllers along the planned robot trajectories, based on standard linear techniques.

6.1. Extensions

The present work can be improved in several directions. A possible drawback of the chosen interpolation method stands in the “swinging” last link motion induced by the use of seventh-order polynomials as trajectories assigned to its center of percussion. A remarkable benefit should be obtained by separating path synthesis from timing law generation. Instead of using high-order time polynomials $y(t)$, which are prone to wandering problems, one can fit suitable low-order parametrized functions $y(s)$ between the start and goal configurations, satisfying geometric (directional) boundary conditions, and then use a scalar time function $s(t)$ in order to satisfy the remaining differential conditions at the start and goal states. Indeed, the idea of time-scaling has already been successfully applied to motion optimization of underactuated robots (Arai, Tanie, and Shiroma 1998). On the other hand, the number of free design choices available in the proposed approach leaves room for many optimization issues, e.g., in terms of energy or torque minimization during the maneuver. Also, the possible addition of intermediate configurations as

via-points may yield more natural trajectories and/or singularity avoidance, as seen in Section 5.2.

We have analyzed the case in which the passive joint is the last in the kinematic chain, but it can be easily recognized that the presented planning method is applicable also when an intermediate joint is passive, as long as the latter is preceded by at least two active joints proximal to the base. In fact, the distal active joints can be frozen so as to recover the situation considered in this paper. When the second joint in the chain is passive (and rotational), the situation is similar to the 2R or PR underactuated robot; thus, no planning method is available but reconfiguration may be performed by feedback as in Spong (1998) (with gravity) or as in De Luca, Mattone, and Oriolo (2000) (without gravity). Finally, when the first joint is passive, the associated dynamic equation is at least partially integrable in the absence of gravity (see Oriolo and Nakamura (1991)).

The case of multiple degree of underactuation is still open in general, even in the planar case. In the special situation where the last $n - 2$ joints are rotational and passive, the center of percussion of the last link is still a linearizing output provided that each of the last $n - 3$ links is hinged at the center of percussion of the previous link. For this case, a motion planner based on a sequence of elementary rotation or translation maneuvers has been developed in Shiroma, Arai, and Tanie (1998), while the extension of the present dynamic feedback linearization technique has been considered in De Luca and Iannitti (2001).

Acknowledgments

This work was supported by the MURST project *MISTRAL*.

References

- Arai, H., and Tachi, S. 1991. Position control of a manipulator with passive joints using dynamic coupling. *IEEE Trans. on Robotics and Automation* 7(4):528–534.
- Arai, H., Tanie, K., and Shiroma, N. 1998a. Time-scaling control of an underactuated manipulator. *IEEE Int. Conf. on Robotics and Automation*, pp. 2619–2626.
- Arai, H., Tanie, K., and Shiroma, N. 1998b. Nonholonomic control of a three-DoF planar underactuated manipulator. *IEEE Trans. on Robotics and Automation* 14(5):681–695.
- Bicchi, A. 2000. Hands for dexterous manipulation and robust grasping: A difficult road toward simplicity. *IEEE Trans. on Robotics and Automation* 16(6):652–662.
- Bicchi, A., and Goldberg, K. (orgs.). 1996. *Minimalism in Robot Manipulation*. Workshop at the 1996 *IEEE Int. Conf. on Robotics and Automation*.
- Brockett, R. W. 1983. Asymptotic stability and feedback stabilization. In *Differential Geometric Control Theory*, eds. R. W. Brockett, R. S. Millman, and H. J. Sussmann, 181–191, Birkhäuser.
- Bullo, F., and Lynch, K. M. 2001. Kinematic controllability for decoupled trajectory planning in underactuated mechanical systems. *IEEE Trans. on Robotics and Automation* 17(4):402–412.
- Charlet, B., Lévine, J., and Marino, R. 1989. On dynamic feedback linearization. *Systems and Control Lett.*, vol. 13, pp. 143–152.
- d'Andrea-Novel, B., Bastin, G., and Campion, G. 1995. Control of nonholonomic wheeled mobile robots by state feedback linearization. *Int. J. of Robotics Research* 14(6):543–559.
- De Luca, A., and Iannitti, S. 2001. Dynamic feedback linearization of an XYnR planar underactuated robot with n passive joints. *2001 Journées Doctorales d'Automatique*, pp. 281–287.
- De Luca, A., Iannitti, S., and Oriolo, G. 2001. Stabilization of a PR planar underactuated robot. *2001 IEEE Int. Conf. on Robotics and Automation*, pp. 2090–2095.
- De Luca, A., Iannitti, S., Mattone, R., Oriolo, G. 2001. Control problems in underactuated manipulators. *2001 IEEE/ASME Int. Conf. on Advanced Mechatronics*, pp. 855–928.
- De Luca, A., and Lucibello, P. 1998. A general algorithm for dynamic feedback linearization of robots with elastic joints. *1998 IEEE Int. Conf. on Robotics and Automation*, pp. 504–510.
- De Luca, A., Mattone, R., and Oriolo, G. 2000. Stabilization of an underactuated planar 2R manipulator. *Int. J. of Robust and Nonlinear Control*, vol. 10, pp. 181–198.
- De Luca, A., and Oriolo, G. 2000a. Motion planning and trajectory control of an underactuated three-link robot via dynamic feedback linearization. *2000 IEEE Int. Conf. on Robotics and Automation*, pp. 2789–2795.
- De Luca, A., and Oriolo, G. 2000b. Motion planning under gravity for underactuated three-link robots. *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp. 139–144.
- De Luca, A., and Oriolo, G. 1998. Stabilization of the Acrobot via iterative state steering. *1998 IEEE Int. Conf. on Robotics and Automation*, pp. 3581–3587.
- Descusse, J., and Moog, C. H. 1985. Decoupling with dynamic compensation for strong invertible affine nonlinear systems. *Int. J. of Control*, vol. 43, pp. 1387–1398.
- Fliess, M., Lévine, J., Martin, Ph., and Rouchon, P. 1995. Flatness and defect of nonlinear systems: Introductory theory and examples. *Int. J. of Control* 61(6):1327–1361.
- Faiz, N., and Agrawal, S. K. 1998. Optimal planning of an under-actuated planar body using higher-order method. *1998 IEEE Int. Conf. on Robotics and Automation*, pp. 736–741.
- Funda, J., Taylor, R. H., Eldridge, B., Gomory, S., and Gruben, K. G. 1996. Constrained cartesian motion control for teleoperated surgical robots. *IEEE Trans. on Robotics and Automation* 12(3):453–465.

- Imura, J., Kobayashi, K., and Yoshikawa, T. 1996. Nonholonomic control of a three-link planar manipulator with a free joint. *35th IEEE Conf. on Decision and Control*, pp. 1435–1436.
- Isidori, A. 1995. *Nonlinear Control Systems*, 3rd Edition, Springer-Verlag.
- Laumond, J.-P. (ed.) 1998. *Robot Motion Planning and Control*, Lecture Notes in Control and Information Sciences, vol. 229. London: Springer-Verlag.
- Lewis, A. D., and Murray, R. M. 1997. Configuration controllability of simple mechanical control systems. *SIAM J. on Control and Optimization* 35(3):766–790.
- Lynch, K. M., and Mason, M. T. 1999. Dynamic nonprehensile manipulation: Controllability, planning, and experiments. *Int. J. of Robotics Research* 18(1):64–92.
- Martin, Ph., Devasia, S., and Paden, B. 1996. A different look at output tracking: Control of a VTOL aircraft. *Automatica* 32(1):101–107.
- Martin, Ph., Murray, R. M., and Rouchon, P. 1997. Flat systems. Notes for the Mini-Course held at the *4th European Control Conf.*
- Murray, R. M., Li, Z., and Sastry, S. S. 1994. *A Mathematical Introduction to Robotic Manipulation*. CRC Press.
- Nakamura, Y., Suzuki, T., and Koinuma, M. 1997. Nonlinear behavior and control of nonholonomic free-joint manipulator. *IEEE Trans. on Robotics and Automation* 13(6):853–862.
- Nakanishi, J., Fukuda, T., Koditschek, D. 2000. A brachiating robot controller. *IEEE Trans. on Robotics and Automation* 16(2):109–123.
- Oriolo, G., and Nakamura, Y. 1991. Control of mechanical systems with second-order nonholonomic constraints: Underactuated manipulators. *30th IEEE Conf. on Decision and Control*, pp. 2398–2403.
- Rathinam, M., and Murray, R. M. 1998. Configuration flatness of Lagrangian systems underactuated by one control. *SIAM J. on Control and Optimization* 36(1):164–179.
- Shiroma, N., Arai, H., and Tanie, K. 1998. Nonholonomic motion planning for coupled planar rigid bodies. *3rd Int. Conf. on Advanced Mechatronics*, pp. 173–178.
- Spong, M. W. 1999. Bipedal locomotion, robot gymnastics, and robot air hockey: A rapprochement. *Super-Mechano Systems Workshop (SMS'99)*, Tokyo.
- Spong, M. W. 1998. Underactuated mechanical systems. In *Control Problems in Robotics and Automation*, eds. B. Siciliano and K. P. Valavanis, LNCIS, vol. 230, pp. 135–150. London: Springer Verlag.
- Spong, M. W. 1995. The swing up control problem for the Acrobot. *IEEE Control Systems* 15(1):49–55.
- Spong, M. W., and Block, D. 1995. The Pendubot: A mechatronic system for control research and education. *34th IEEE Conf. on Decision and Control*, pp. 555–557.
- Vafa, Z., and Dubowsky, S. 1990. The kinematics and dynamics of space manipulators: The virtual manipulator approach. *Int. J. of Robotics Research* 9(4):3–21.
- van Nieuwstadt, M. J., and Murray, R. M. 1998. Real-time trajectory generation for differentially flat systems. *Int. J. of Robust and Nonlinear Control* 8(11):995–1020.
- Yoshikawa, T., Kobayashi, K., and Watanabe, T. 2000. Design of a desirable trajectory and convergent control for 3-DoF manipulator with a nonholonomic constraint. *2000 IEEE Int. Conf. on Robotics and Automation*, pp. 1805–1810.