

Kinematic Modeling and Redundancy Resolution for Nonholonomic Mobile Manipulators

Alessandro De Luca Giuseppe Oriolo Paolo Robuffo Giordano

Dipartimento di Informatica e Sistemistica

Università di Roma "La Sapienza"

Via Eudossiana 18, 00184 Roma, Italy

{deluca,oriolo,robuffo}@dis.uniroma1.it

Abstract—We consider robotic systems made of a nonholonomic mobile platform carrying a manipulator (nonholonomic mobile manipulator, NMM). By combining the manipulator differential kinematics with the admissible differential motion of the platform, a simple and general kinematic model for NMMs is derived. Assuming that the robotic system is kinematically redundant for a given task, we present the extension of redundancy resolution schemes originally developed for standard manipulators, in particular the Projected Gradient (PG) and the Reduced Gradient (RG) optimization-based methods. The case of a configuration-dependent task specification is also discussed. The proposed modeling approach is illustrated with reference to representative NMMs, and the performance of the PG and RG methods for redundancy resolution is compared on a series of numerical case studies.

I. INTRODUCTION

A mobile manipulator consists of a mobile platform carrying a standard manipulator arm. This robotic system merges the dexterity of the manipulator with the increased workspace capabilities of the mobile platform. The platform may have different modes of locomotion, e.g., wheeled bases, walking robots, free-flying space robots. In this paper, we focus on the case of a manipulator carried by a nonholonomic (typically, wheeled) platform, addressing the problem of kinematic modeling and resolution of task-dependent redundancy for generic Nonholonomic Mobile Manipulators (NMMs).

Nonholonomic wheeled platforms are subject to nonintegrable differential constraints due to the rolling without slipping of the wheels on the ground [1]. These constraints limit the instantaneous mobility but do not affect the global accessibility of the configuration space by suitable maneuvers. Several methods have been proposed in the literature for planning and controlling the motion of these robotic systems, see, e.g., [2], [3]. On the other hand, standard manipulators are kinematically unconstrained systems, i.e., the whole generalized velocity space can be accessed at any configuration. As for end-effector velocities, a local loss of mobility occurs at singular configurations. A common way to deal with singularities is to adopt a kinematically redundant manipulator, with a number n of degrees of freedom (i.e., joints) larger than the number s of variables needed to accomplish a given task. The $n - s$ extra dofs (degree of redundancy) may be used for the optimization of performance criteria (e.g., maximization of manipulability for singularity avoidance) or to satisfy ad-

ditional task requirements while executing a desired primary task [4]. Another interesting solution is to achieve redundancy by adding mobility to the (otherwise fixed) manipulator base, thereby obtaining an NMM.

Kinematic modeling and motion generation for NMMs has been addressed in the literature following two basic approaches. Some authors add the nonholonomic constraints in the description of the differential kinematics [5], [6]. A more efficient formulation, adopted also in this work, explicitly entails the differential motions that are feasible w.r.t. the nonholonomic constraints [7], [8]. The problem is usually tackled at a first-order kinematic level (i.e., with (pseudo-)velocities as command inputs) since planning the motion of NMMs is essentially a kinematic problem, but inclusion of dynamic aspects has also been considered [9]. When an NMM is redundant for a given task (this concept requires some caution, see Section II), redundancy can be exploited by extending schemes already available for standard manipulators, e.g., task space augmentation [5], [10], [11] or pseudoinversion of the Jacobian and use of its null-space motion [12].

In this paper, we present a simple and compact derivation of the kinematic model of general NMMs in terms of the available velocity command inputs for the combined platform/manipulator system. With respect to [8], we avoid in Section II the introduction of internal coordinates of the platform that are not relevant for the definition and execution of the task (e.g., for pose tracking with the manipulator end-effector). Based on this model, Section III presents the extension to NMMs of three approaches for redundancy resolution and kinematic control of standard manipulators, namely the Extended Jacobian (EJ) method, and the Projected Gradient (PG) and Reduced Gradient (RG) optimization methods. In particular, we detail how the constrained optimization of a configuration-dependent objective function can be achieved using the available command inputs. These schemes can be modified to cover also the interesting case of a configuration-dependent task specification, as shown in Section IV. Section V illustrates the derivation of the kinematic model for some representative examples of NMMs and analyzes the associated singularities. Finally, Section VI is devoted to the numerical comparison of task accuracy and optimization performance between the PG and the RG redundancy resolution methods for NMMs in a series of case studies.

II. KINEMATIC MODELING OF NMMs

Consider a robotic system made of a nonholonomic mobile platform carrying a manipulator. Let the configuration vector be $q = [q_p \ q_m]^T$, where $q_p \in \mathbb{R}^{n_p}$ and $q_m \in \mathbb{R}^{n_m}$ are the generalized coordinates of the platform and the manipulator, respectively. The variables needed to describe the task to be performed by the NMM are related to q by the kinematic map

$$r = f(q_p, q_m), \quad (1)$$

where $r \in \mathbb{R}^s$ is the task vector. Typically, nonholonomic constraints on platform motion arise from the pure rolling of its wheels on the ground. The platform kinematic model is then given by the driftless system

$$\dot{q}_p = G(q_p)u_p, \quad (2)$$

where $u_p \in \mathbb{R}^p$ are the velocity inputs (pseudovelocities), with $p < n_p$, and the columns of the $n_p \times p$ matrix G span the admissible velocity space at each platform configuration. From a kinematic point of view, the manipulator is instead a completely unconstrained system, i.e., vector \dot{q}_m can be arbitrarily specified at any arm configuration. Hence, we let

$$\dot{q}_m = u_m, \quad (3)$$

where $u_m \in \mathbb{R}^m$ are the velocity inputs for the manipulator. The velocity input vector available for kinematic control of the whole NMM is then $u = [u_p^T \ u_m^T]^T \in \mathbb{R}^{p+m}$.

For an NMM described by eqs. (2)–(3), we distinguish two redundancy concepts. When the total number of generalized coordinates exceeds the dimension of the task expressed by (1), i.e., $n_p + n_m > s$, a *static* (configuration) redundancy occurs. This definition is of interest, e.g., when an NMM configuration realizing a given end-effector placement is to be found. Instead, when the total number of velocity inputs (equivalently, of dof's) exceeds the dimension of the task, i.e., $p + n_m > s$, a *kinematic* (motion) redundancy occurs. This is the relevant definition used in the rest of this paper, where a kinematic inversion scheme is used for generating the velocity commands needed to execute a given task. Clearly, the two redundancy concepts collapse for standard manipulators.

Differentiating the relation (1) with respect to time gives

$$\begin{aligned} \dot{r} &= \frac{\partial f}{\partial q_p} \dot{q}_p + \frac{\partial f}{\partial q_m} \dot{q}_m = J_p(q)G(q_p)u_p + J_m(q)u_m \\ &= \begin{bmatrix} J_p(q)G(q_p) & J_m(q) \end{bmatrix} \begin{bmatrix} u_p \\ u_m \end{bmatrix} = J(q)u. \end{aligned} \quad (4)$$

The $s \times (p + n_m)$ matrix J will be simply called the *Jacobian* of the nonholonomic mobile manipulator, even if, strictly speaking, not all its elements are partial derivatives. In the following, we shall only consider kinematically redundant NMMs, where J has more columns than rows. The above formulation of the NMM kinematics is similar but simpler than that of [8]. In comparison with the approach of, e.g., [5], it uses the explicit form of the admissible platform velocities as expressed by eq. (2) instead of working with the nonholonomic constraints of the mobile base, i.e., $A(q_p)\dot{q}_p = 0$.

All classical problems addressed for standard redundant manipulators (study of singularities and their avoidance, augmentation of tasks and their priority, optimization of performance criteria, cyclicity of configuration motion, etc. — see [4]) can be directly reformulated for NMMs in terms of the Jacobian J in eq. (4). For example, a configuration \bar{q} is *singular* iff $\text{rank } J(\bar{q}) < s$. The addition of a mobile platform to a given manipulator can often delete many of the singularities that would affect the manipulator taken alone. This obvious fact is illustrated in Section V with two specific examples.

III. REDUNDANCY RESOLUTION METHODS FOR NMMs

We present here the extension to NMMs of three redundancy resolution approaches originally proposed for standard manipulators, i.e., the Extended Jacobian [4], the Projected Gradient [13], and the Reduced Gradient [14]–[16] methods. In the following, we shall assume that a desired time evolution $r_d(t)$ is specified for the task variables r . As usual [17], one should set $\dot{r} = \dot{r}_d$ for planning and $\dot{r} = \dot{r}_d + K(r_d - r)$, with a gain matrix $K > 0$, for kinematic control. The feedback term allows exponential recovery of initial task errors.

A. Extended Jacobian (EJ)

Assume that an additional constraint $y = h(q)$ of dimension $k = p + n_m - s$ is attached to eq. (1) in order to specify some desirable aspect of the solution. Differentiation yields

$$\begin{bmatrix} \dot{r} \\ \dot{y} \end{bmatrix} = \begin{bmatrix} J(q) \\ \frac{\partial h(q)}{\partial q_p} G(q_p) & \frac{\partial h(q)}{\partial q_m} \end{bmatrix} \begin{bmatrix} u_p \\ u_m \end{bmatrix} = J_e(q)u,$$

being J_e the square $(p + n_m) \times (p + n_m)$ extended Jacobian of the NMM. Whenever J_e is nonsingular, motion commands can be generated as

$$u = J_e^{-1}(q) \begin{bmatrix} \dot{r} \\ \dot{y} \end{bmatrix}.$$

In choosing h , special attention should be paid to *algorithmic singularities*, i.e., configurations where J_e is singular [4] in spite of the fact that $\text{rank } J = s$ and the last k rows of J_e are linearly independent. Another version of the EJ method for NMMs has been proposed in [5]. Since the solution realizing the given (square) extended task is unique, the two formulations are equivalent.

B. Projected Gradient (PG)

For a given \dot{r} , all solutions u to the differential kinematics (4) can be expressed as

$$u = J^\dagger(q)\dot{r} + (I - J^\dagger(q)J(q))u_0, \quad (5)$$

where J^\dagger is the pseudoinverse of matrix J , $I - J^\dagger J$ is the orthogonal projection operator into $\mathcal{N}(J)$, and $u_0 \in \mathbb{R}^{p+n_m}$ is arbitrary. A weighted pseudoinverse can be used in (5) in place of J^\dagger in order to guarantee a consistent solution in the presence of mixed linear/angular quantities.

For standard manipulators, where \dot{q} and u coincide, u_0 can be chosen so as to locally optimize a configuration-dependent

criterion $H(q)$ by setting $u_0 = \dot{q}_0 = \pm \alpha \nabla_q H(q)$. A suitable stepsize $\alpha > 0$ can be found by line search techniques [18]. Common choices for $H(q)$ are the manipulability index or the available joint range.

For NMMs, however, some care is required when devising a similar scheme, since the available commands are the pseudovelocities u rather than the generalized velocities \dot{q} . This means that a certain generalized velocity $\dot{q}_0 = \nabla_q H$ may not be feasible for the nonholonomic constraints of the platform, i.e., there may exist no choice of u_0 in eq. (5) which realizes it. One way to identify an appropriate command u_0 is proposed in the following. Differentiating in time $H(q)$ and using eqs. (2–3) gives

$$\begin{aligned} \dot{H}(q) &= \frac{\partial H(q)}{\partial q_p} G(q_p) u_p + \frac{\partial H(q)}{\partial q_m} u_m \\ &= \nabla_q^T H(q) \begin{bmatrix} G(q_p) & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} u_p \\ u_m \end{bmatrix}. \end{aligned} \quad (6)$$

Therefore, the command vector $u_H(q)$ that locally realizes the maximum improvement of the objective function $H(q)$ is

$$u_H(q) = \pm \alpha \begin{bmatrix} G^T(q_p) & 0 \\ 0 & I \end{bmatrix} \nabla_q H(q). \quad (7)$$

Motion commands are then generated by setting $u_0 = u_H(q)$ in (5). The corresponding generalized velocity of the platform

$$\dot{q}_{p,H} = \pm \alpha G(q_p) G^T(q_p) \nabla_{q_p} H(q)$$

represents a projection of $\pm \alpha \nabla_{q_p} H$ onto the subspace of generalized velocities that are admissible with respect to the nonholonomic constraints. In comparison with [8], the above derivation appears to be simpler.

C. Reduced Gradient (RG)

Both the PG and EJ resolution schemes require a large number of operations, either due to the computation of J^\dagger or to the inversion of the extended Jacobian J_e . An alternative strategy is to perform optimization of the objective function (or satisfaction of a secondary task) by directly working in the reduced $(p + n_m - s)$ -dimensional space of velocity inputs that satisfy the s -dimensional task (1). The Reduced Gradient method, originally introduced for manipulators in [14], implements this idea.

Assume that the Jacobian matrix $J(q)$ in eq. (4) has full rank at the current configuration q . Then, it is always possible to find a permutation matrix T such that

$$J(q)T = \begin{bmatrix} J_a(q) & J_b(q) \end{bmatrix},$$

with a nonsingular $s \times s$ matrix J_a . This induces a reordering of the velocity input vector since

$$\begin{bmatrix} u_p \\ u_m \end{bmatrix} = T \begin{bmatrix} u_a \\ u_b \end{bmatrix} = \begin{bmatrix} T_{pa} & T_{pb} \\ T_{ma} & T_{mb} \end{bmatrix} \begin{bmatrix} u_a \\ u_b \end{bmatrix}, \quad (8)$$

where $u_a \in \mathbb{R}^s$ and $u_b \in \mathbb{R}^{p+n_m-s}$. The differential kinematics (4) becomes accordingly (dropping dependencies)

$$\dot{r} = J \begin{bmatrix} u_p \\ u_m \end{bmatrix} = JT \begin{bmatrix} u_a \\ u_b \end{bmatrix} = \begin{bmatrix} J_a & J_b \end{bmatrix} \begin{bmatrix} u_a \\ u_b \end{bmatrix}.$$

The task motion constraint is automatically satisfied by letting

$$u_a = J_a^{-1} (\dot{r} - J_b u_b). \quad (9)$$

The remaining command u_b is chosen so as to locally optimize an objective function $H(q)$ as follows. Using eqs. (8) and (9) in (6) leads to

$$\begin{aligned} \dot{H}(q) &= \left[\frac{\partial H}{\partial q_p} G T_{pa} + \frac{\partial H}{\partial q_m} T_{ma} \right] J_a^{-1} \dot{r} \\ &\quad + \nabla_q^T H(q) \begin{bmatrix} G T_{pa} & G T_{pb} \\ T_{ma} & T_{mb} \end{bmatrix} \begin{bmatrix} -J_a^{-1} J_b \\ I \end{bmatrix} u_b. \end{aligned}$$

Therefore, the command vector u_b that locally realizes the maximum improvement of the objective function H is

$$\begin{aligned} u_b &= \pm \alpha \begin{bmatrix} -(J_a^{-1} J_b)^T & I \end{bmatrix} \begin{bmatrix} T_{pa}^T G^T & T_{ma}^T \\ T_{pb}^T G^T & T_{mb}^T \end{bmatrix} \nabla_q H \\ &= \pm \alpha \begin{bmatrix} -(J_a^{-1} J_b)^T & I \end{bmatrix} T^T \begin{bmatrix} G^T & 0 \\ 0 & I \end{bmatrix} \nabla_q H. \end{aligned} \quad (10)$$

By comparing eq. (10) with eq. (7), we note that u_b is the *reduction* (up to a permutation of components) of the velocity input u_H to the subspace of commands that automatically satisfy the task constraint.

When $\dot{r} = 0$ (self-motion) and the degree of kinematic redundancy of the NMM is 1, the RG and PG methods generate velocity commands in the same direction, although the former method allows longer steps to be taken and results in a faster optimization [14]. In any other case, the two methods generate different directions in the command space, and thus in the NMM configuration space. The computational load of the RG method is typically lower. However, it may be necessary to change the permutation matrix $T = T(q)$ in eq. (8) along the motion so as to extract a (different) nonsingular matrix $J_a(q)$.

In the special case when $n_m = s$ and the manipulator itself is not in a singular configuration, one can choose $J_a = J_m$ and $J_b = J_p G$ so that eqs. (9) and (10) simplify to

$$\begin{aligned} u_m &= J_m^{-1} (\dot{r} - J_p G u_p) \\ u_p &= \pm \alpha G^T \begin{bmatrix} -(J_a^{-1} J_b)^T & I \end{bmatrix} \nabla_q H. \end{aligned}$$

IV. REDUNDANCY RESOLUTION WITH A CONFIGURATION-DEPENDENT TASK

In some cases, the desired robot task at time t depends also on the current configuration $q(t)$. Hence, in the presence of kinematic redundancy, the task history cannot be specified in advance as a pure time-dependent function. This situation can be modeled by

$$r_d(t) = g(t, q(t)). \quad (11)$$

The time derivative of (11) is

$$\dot{r}_d(t) = \frac{\partial g(t, q)}{\partial t} + \frac{\partial g(t, q)}{\partial q} \begin{bmatrix} G(q_p) & 0 \\ 0 & I \end{bmatrix} u, \quad (12)$$

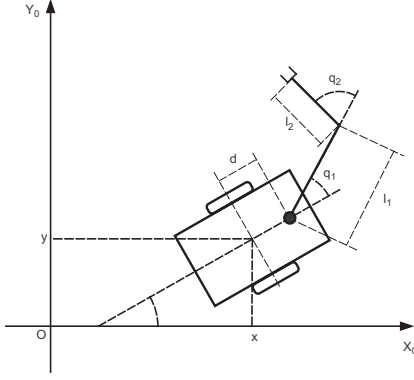


Fig. 1. A planar NMM with a 2R manipulator (top view)

where eqs. (2–3) have been used. Replacing \dot{r} in the differential kinematics (4) with \dot{r}_d given by eq. (12) leads to

$$\frac{\partial g(t, q)}{\partial t} + \frac{\partial g(t, q)}{\partial q} \begin{bmatrix} G(q_p) & 0 \\ 0 & I \end{bmatrix} u = J(q)u,$$

which can be reorganized as

$$\frac{\partial g(t, q)}{\partial t} = \left[J(q) - \frac{\partial g(t, q)}{\partial q} \begin{bmatrix} G(q_p) & 0 \\ 0 & I \end{bmatrix} \right] u = \bar{J}(t, q)u. \quad (13)$$

All the above redundancy resolution methods for NMMs can be extended to configuration-dependent tasks (11) on the basis of eq. (13). For motion planning, simply replace vector \dot{r}_d with $\partial g/\partial t$ and matrix J with \bar{J} in all formulas. For kinematic control purposes, the additional term $K(r_d - r)$ should be included in the left-hand side of (13). The above formulation slightly generalizes the standard case in the literature and can also be used component-wise. An example where the task is configuration-dependent will be considered in Section VI.

V. ANALYSIS OF EXAMPLES

We now give two modeling examples of NMM, showing how the combination nonholonomic platform/manipulator reduces the set of singular points. Some of the developments below are preliminary to the use of the RG method.

A. Unicycle + 2R planar manipulator

Consider a 2R manipulator in the horizontal plane, with link lengths l_1 and l_2 , mounted on a two-wheel differentially driven mobile platform with unicycle kinematics, as shown in Fig. 1. The base of the manipulator is placed along the main axis of the platform, at a distance d from the wheels' axis. The configuration vector of this NMM is $q = [q_p^T \ q_m^T]^T \in \mathbb{R}^5$, with $q_p = [x \ y \ \theta]^T \in \mathbb{R}^3$ and $q_m = [q_1 \ q_2]^T \in \mathbb{R}^2$.

The task of positioning the NMM end-effector in the plane has dimension $s = 2$, so that the degree of static redundancy is 3. The associated kinematic map is

$$\begin{bmatrix} r_x \\ r_y \end{bmatrix} = \begin{bmatrix} x + dc_\theta + l_1 c_{\theta q_1} + l_2 c_{\theta q_1 q_2} \\ y + ds_\theta + l_1 s_{\theta q_1} + l_2 s_{\theta q_1 q_2} \end{bmatrix}, \quad (14)$$

where c_{ijk} and s_{ijk} stand for $\cos(i+j+k)$ and $\sin(i+j+k)$. The Jacobians $J_p(q)$ and $J_m(q)$ are

$$J_p = \begin{bmatrix} 1 & 0 & -ds_\theta - l_1 s_{\theta q_1} - l_2 s_{\theta q_1 q_2} \\ 0 & 1 & dc_\theta + l_1 c_{\theta q_1} + l_2 c_{\theta q_1 q_2} \end{bmatrix}$$

$$J_m = \begin{bmatrix} -l_1 s_{\theta q_1} - l_2 s_{\theta q_1 q_2} & -l_2 s_{\theta q_1 q_2} \\ l_1 c_{\theta q_1} + l_2 c_{\theta q_1 q_2} & l_2 c_{\theta q_1 q_2} \end{bmatrix}.$$

The platform kinematics is described by

$$\dot{q}_p = \begin{bmatrix} \cos \theta & 0 \\ \sin \theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} = G(q_p)u_p,$$

with input u_p of dimension $p = 2$. Being $u_m = [\dot{q}_1 \ \dot{q}_2]^T \in \mathbb{R}^2$, the degree of kinematic redundancy is 2 for this motion task. The 2×4 Jacobian $J(q)$ in the differential kinematics (4) is

$$\begin{bmatrix} c_\theta & -ds_\theta - l_1 s_{\theta q_1} - l_2 s_{\theta q_1 q_2} & -l_1 s_{\theta q_1} - l_2 s_{\theta q_1 q_2} & -l_2 s_{\theta q_1 q_2} \\ s_\theta & dc_\theta + l_1 c_{\theta q_1} + l_2 c_{\theta q_1 q_2} & l_1 c_{\theta q_1} + l_2 c_{\theta q_1 q_2} & l_2 c_{\theta q_1 q_2} \end{bmatrix}. \quad (15)$$

To identify its singularities, consider the 2×2 minors Δ_{ij} obtained by selecting the i -th and j -th column from (15):

$$\begin{aligned} \Delta_{12} &= d + l_1 c_{q_1} + l_2 c_{q_1 q_2} & \Delta_{23} &= -d(l_1 s_{q_1} + l_2 s_{q_1 q_2}) \\ \Delta_{13} &= -l_1 c_{q_1} - l_2 c_{q_1 q_2} & \Delta_{24} &= -l_2(l_1 s_{q_2} + ds_{q_1 q_2}) \\ \Delta_{14} &= -l_2 c_{q_1 q_2} & \Delta_{34} &= l_1 l_2 s_{q_2}. \end{aligned}$$

Clearly, Δ_{12} and Δ_{13} cannot simultaneously vanish if $d \neq 0$. In this case, $\text{rank } J(q) = 2$ everywhere and the NMM has no singular configurations (as opposed to the manipulator taken alone). Moreover, if $d > l_1 + l_2$, Δ_{12} is always nonzero and the RG method of Section III-C can be used with the globally defined inverse of $J_a = J_p G$ (first two columns in (15)) without changing the permutation matrix T in eq. (8). In general, one may need to switch between the matrices corresponding to Δ_{12} and Δ_{13} .

Suppose now that the task is to follow a given cartesian trajectory while pointing at a fixed object with a camera mounted on the end-effector. To this end, we add the absolute orientation $\delta = \theta + q_1 + q_2$ of the second link to the end-effector positioning task (14), obtaining an extended task of dimension $s = 3$. The associated 3×4 Jacobian $J(q)$ becomes

$$\begin{bmatrix} c_\theta & -ds_\theta - l_1 s_{\theta q_1} - l_2 s_{\theta q_1 q_2} & -l_1 s_{\theta q_1} - l_2 s_{\theta q_1 q_2} & -l_2 s_{\theta q_1 q_2} \\ s_\theta & dc_\theta + l_1 c_{\theta q_1} + l_2 c_{\theta q_1 q_2} & l_1 c_{\theta q_1} + l_2 c_{\theta q_1 q_2} & l_2 c_{\theta q_1 q_2} \\ 0 & 1 & 1 & 1 \end{bmatrix}.$$

As before, by computing the minors Δ_{ijk}

$$\begin{aligned} \Delta_{123} &= d & \Delta_{134} &= l_1 c_{q_1} \\ \Delta_{124} &= d + l_1 c_{q_1} & \Delta_{234} &= l_1 d s_{q_1}, \end{aligned}$$

we conclude that, as long as $d \neq 0$, Δ_{123} never vanishes and the Jacobian has full rank everywhere.

B. Unicycle + 3R elbow-type manipulator

Consider a 3R elbow-type manipulator (with link lengths l_i , $i = 1, 2, 3$) mounted on the same previous platform of height h , with an offset $d \neq 0$ with respect to its center (see Fig. 2). Let $q_p = [x \ y \ \theta]^T \in \mathbb{R}^3$ and $q_m = [q_1 \ q_2 \ q_3]^T \in \mathbb{R}^3$ be the configuration vectors of the platform and manipulator,

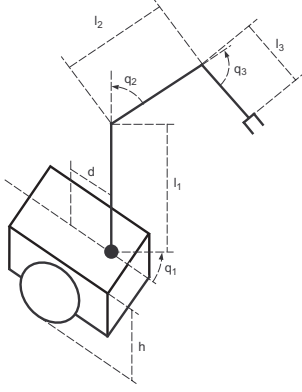


Fig. 2. A spatial NMM with a 3R elbow-type manipulator

respectively. For the task of positioning the end-effector in 3D-space ($s = 3$), the kinematic map is

$$\begin{bmatrix} r_x \\ r_y \\ r_z \end{bmatrix} = \begin{bmatrix} x + dc_\theta + l_2c_{\theta q_1}s_{q_2} + l_3c_{\theta q_1}s_{q_2q_3} \\ y + ds_\theta + l_2s_{\theta q_1}s_{q_2} + l_3s_{\theta q_1}s_{q_2q_3} \\ h + l_1 + l_2c_{q_2} + l_3c_{q_2q_3} \end{bmatrix},$$

and the associated Jacobian $J(q)$ of the NMM is

$$\begin{bmatrix} c_\theta & -ds_\theta - l_2s_{\theta q_1}s_{q_2} - l_3s_{\theta q_1}s_{q_2q_3} \\ s_\theta & dc_\theta + l_2c_{\theta q_1}s_{q_2} + l_3c_{\theta q_1}s_{q_2q_3} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} -l_2s_{\theta q_1}s_{q_2} - l_3s_{\theta q_1}s_{q_2q_3} & l_2c_{\theta q_1}c_{q_2} + l_3c_{\theta q_1}c_{q_2q_3} & l_3c_{\theta q_1}c_{q_2q_3} \\ l_2c_{\theta q_1}s_{q_2} + l_3c_{\theta q_1}s_{q_2q_3} & l_2s_{\theta q_1}c_{q_2} + l_3s_{\theta q_1}c_{q_2q_3} & l_3s_{\theta q_1}c_{q_2q_3} \\ 0 & -l_2s_{q_2} - l_3s_{q_2q_3} & -l_3s_{q_2q_3} \end{bmatrix}.$$

Since $u = [u_p^T u_m^T]^T = [v \omega \dot{q}_1 \dot{q}_2 \dot{q}_3]^T \in \mathbb{R}^5$, the degree of kinematic redundancy is 2. In order to study the rank of $J(q)$, we compute the $\binom{5}{3} = 10$ possible minors:

$$\begin{aligned} \Delta_{123} &= -l_3s_{q_2q_3}(\Lambda c_{q_1} + d) & \Delta_{145} &= l_2l_3\Lambda s_{q_3} \\ \Delta_{124} &= -l_3s_{q_2q_3}c_{q_1}\Lambda & \Delta_{234} &= 0 \\ \Delta_{125} &= -l_2l_3s_{q_1}s_{q_3} & \Delta_{235} &= -\Lambda(\Lambda c_{q_1} + d) \\ \Delta_{134} &= -dl_3s_{q_1}s_{q_2q_3}\Lambda & \Delta_{245} &= -\Lambda^2c_{q_1} \\ \Delta_{135} &= l_2l_3s_{q_3}(dc_{q_1} + \Lambda) & \Delta_{345} &= -d\Lambda^2s_{q_1}, \end{aligned}$$

where $\Lambda = l_2s_{q_2} + l_3s_{q_2q_3} = 0$ corresponds to the shoulder singularity for the elbow-type manipulator. No minor is always nonzero in this case. However, it is easy to verify that if $s_{q_2} \neq 0$ or $s_{q_3} \neq 0$ there is always a nonzero minor. Hence, $\text{rank } J(q) = 3$ for this NMM, except when the manipulator is stretched or folded along the vertical direction.

VI. SIMULATION RESULTS

We consider three case studies of kinematic control for the NMM in Fig. 1, and one for the NMM in Fig. 2. As redundancy resolution scheme, we adopt the RG method, whose optimization performance is compared with the PG method. The task control matrix gain K is set equal to the identity, while a constant step size $\alpha = 10$ has been used for simplicity in gradient computations.

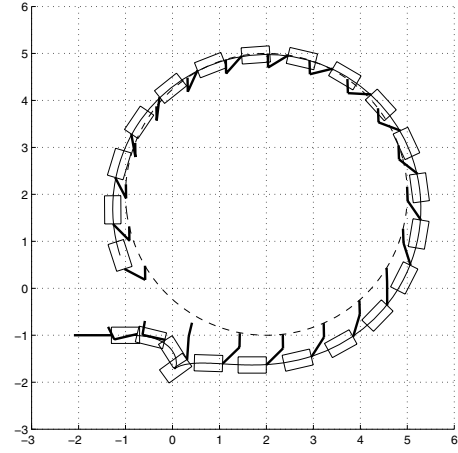


Fig. 3. Trajectory tracking for the planar NMM with RG method

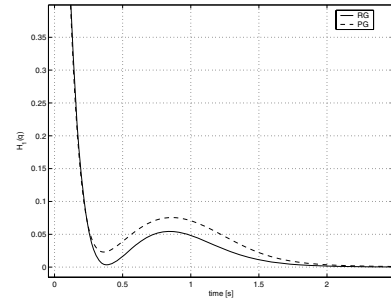


Fig. 4. Comparison between RG and PG methods in minimizing the objective function $H_1(q)$ (zoom on the initial transient)

1) *Position task for the NMM with planar manipulator:* The robot geometric data are $d = 0.3$, $l_1 = 0.5$, and $l_2 = 0.3$ m. Since $d < l_1 + l_2$, there is no minor guaranteed to be always nonzero. However, a simple switching strategy can be adopted for the RG method. Let j_i be the i -th column of the NMM Jacobian $J(q)$ in (15):

- 1) If $|\Delta_{13}| \geq |\Delta_{12}|$, start with the inversion of $J_a = [j_1 j_3]$, otherwise start with $J_a = [j_1 j_2]$;
- 2) if $|\Delta_{13}| < \Theta$ (or $|\Delta_{12}| < \Theta$) switch to the other minor.

In the simulations, a fixed threshold $\Theta = 10^{-2}$ has been used. The end-effector should follow the circular trajectory

$$r_d(t) = \begin{bmatrix} r_{dx}(t) \\ r_{dy}(t) \end{bmatrix} = \begin{bmatrix} 2 + 3 \cos(0.08\pi t + \frac{5}{4}\pi) \\ 2 + 3 \sin(0.08\pi t + \frac{5}{4}\pi) \end{bmatrix}, \quad (16)$$

for $T_s = 25$ s, while minimizing the objective function

$$H_1(q) = \frac{1}{2} \left(\theta + q_1 + q_2 - \frac{\pi}{2} \right)^2 + \frac{1}{2} \left(q_1 - \frac{\pi}{4} \right)^2.$$

The start configuration at $t = 0$ is $q_0 = [-1 \ -1 \ \pi \ 0 \ 0]^T$ and corresponds to the end-effector being out of the desired path.

The stroboscopic motion of the NMM in Fig. 3 shows the good tracking behavior, despite the initial backup of the platform. The performance comparison of the RG and PG methods in Fig. 4 indicates that RG is faster in approaching

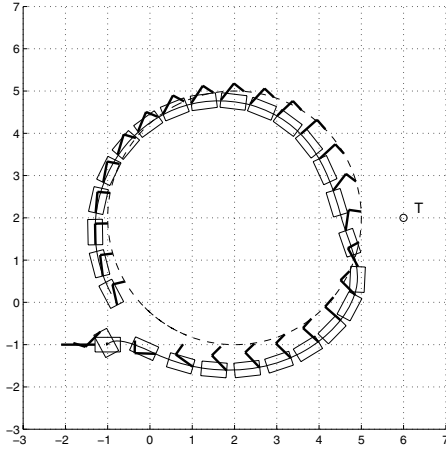


Fig. 5. Trajectory tracking and target pointing for the spatial NMM with RG method

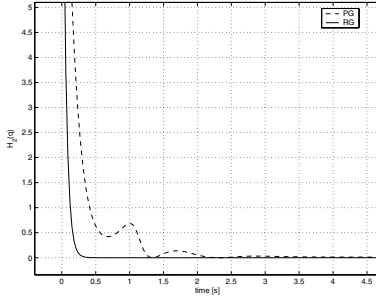


Fig. 6. Comparison between RG and PG methods in minimizing the objective function $H_2(q)$ (zoom on the initial transient)

the minimum value of H_1 (and also better in keeping a lower ripple during steady-state motion).

2) *Position/orientation task for the NMM with planar manipulator*: The former task is extended to include the absolute orientation of the second link of the manipulator. In particular, this should point towards a fixed target point T located at $(x_T, y_T) = (6, 2)$ m. Therefore, we append to the positioning task (16) a third component

$$r_{d\delta}(t) = \text{atan2}(y_T - r_{dy}(t), x_T - r_{dx}(t)).$$

The single degree of kinematic redundancy left is used to minimize the objective function

$$H_2(q) = \frac{1}{2} \left(q_2 + \frac{\pi}{2} \right)^2.$$

Figure 5 shows the stroboscopic motion of the NMM generated with the RG method. A waving behavior around the nominal end-effector trajectory is now realized by the platform in order to satisfy the additional task constraint. No switching strategy is needed here since submatrix $J_a = [j_1 \ j_2 \ j_3]$ is always nonsingular, as shown in Section V-A. Also in this case, the optimization with the RG method is more efficient than with the PG method (see Fig. 6), reaching faster and firmly keeping the absolute minimum of H_2 .

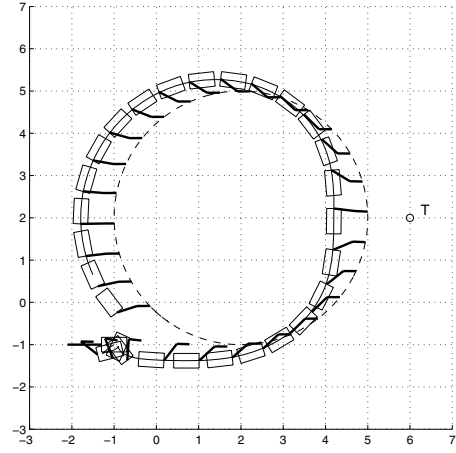


Fig. 7. Trajectory tracking and configuration-dependent target pointing for the planar NMM with RG method

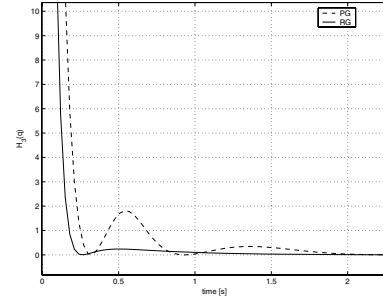


Fig. 8. Comparison between RG and PG methods in minimizing the objective function $H_3(q)$ (zoom on the initial transient)

3) *Configuration-dependent task for the NMM with planar manipulator*: Suppose that a task of dimension $s = 3$ is assigned for the NMM in Fig. 1, specifying the absolute orientation $\delta_1 = \theta + q_1$ of the *first* link of the manipulator in addition to the end-effector position given by eq. (14). The 3×4 NMM Jacobian $J(q)$ associated to this task vector is the matrix in (15) with an extra third row given by $[0 \ 1 \ 1 \ 0]$. It is easy to see that the minor $\Delta_{123} = d \neq 0$, and thus $\text{rank } J(q) = 3$ everywhere. The RG method is then built upon the always nonsingular block $J_a = [j_1 \ j_2 \ j_3]$.

A (mixed) configuration-dependent task specification is obtained extending the desired task (16) with a third component of the form

$$r_{d\delta_1}(q) = \text{atan2}(y_T - (y + ds_\theta + l_1 s_{\theta q_1}), x_T - (x + dc_\theta + l_1 c_{\theta q_1})).$$

This represents a situation where the first link of the manipulator should point towards a target T (chosen as in case 2), as if a camera mounted on the link was forced to look at a fixed point. This last task component is a configuration-dependent scalar function $g(q(t))$ (no explicit dependence on t in this case), as discussed in Section IV. The derivation of the modified Jacobian $\bar{J}(q)$ of eq. (13) is lengthy but straightforward.

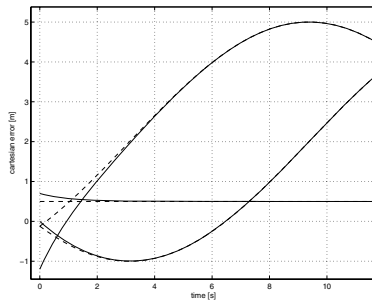


Fig. 9. Cartesian trajectory tracking error for the NMM in Fig. 2 with RG method (dashed lines represent the reference motion)

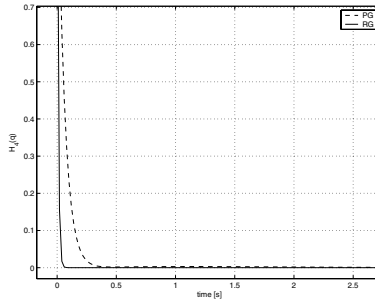


Fig. 10. Comparison between RG and PG methods in minimizing the objective function $H_4(q)$ (zoom on the initial transient)

The objective function chosen for this case is

$$H_3(q) = \frac{1}{2}(\theta + q_1 + q_2)^2,$$

and the results are shown in Figs. 7 and 8. Although the required task is quite demanding, the tracking performance is again very accurate and requires only a complex reconfiguration of the NMM during the initial instants of motion.

4) *Position task for the NMM with elbow-type manipulator:* The geometric data of the robot in Fig. 2 are $d = 0.3$, $h = 0.3$, $l_1 = 0.4$, $l_2 = 0.5$, and $l_3 = 0.4$ m. For the 3D task of end-effector spatial positioning, this NMM will use its two degrees of kinematic redundancy for minimizing the function

$$H_4(q) = \frac{1}{2}q_1^2 + \frac{1}{2}\left(q_3 + \frac{\pi}{2}\right)^2.$$

The end-effector should follow the circular trajectory (16) taking place at a constant height $r_{dz} = 0.5$ m. The initial configuration is $q_0 = [0 \ 0 \ \pi \ 0 \ \pi/2 \ 0]^T$, which corresponds again to the end-effector being out of path. Since there is no minor guaranteed to be always nonzero in this case, a switching strategy similar to the one discussed in case 1 has been implemented for the RG method. The results in Figs. 9 and 10 show a fast recovery of the initial cartesian error and of the optimal value for H_4 with both RG and PG methods.

Movie clips of numerical results, also including the case of manipulability optimization, can be found at the website <http://www.dis.uniroma1.it/labrob/research/NMM.html>.

VII. CONCLUSIONS

Based on the derivation of a simple but general kinematic model for nonholonomic mobile manipulators, we have extended to these systems the singularity analysis and redundancy resolution methods available for standard manipulators. In particular, for nonholonomic mobile manipulators that are kinematically redundant for the given task, extensive simulation has shown the superior optimization performance of the Reduced Gradient over the Projected Gradient method. The desired tasks for the robotic system are accurately executed by the combined motion of all the configuration variables.

Future work will deal with second-order kinematic models (with acceleration inputs) and associated redundancy resolution schemes, as well as dynamic models. Also, we are investigating the use of the proposed approach for visual servoing with a pan-tilt camera (seen as a manipulator).

REFERENCES

- [1] G. Campion, G. Bastin, and B. D'Andrea-Novell, "Structural properties and classification of kinematic and dynamic models of wheeled mobile robots," *IEEE Trans. on Robotics and Automation*, vol. 12, no. 1, pp. 47–62, 1996.
- [2] Z. Li and J. F. Canny (Eds), *Nonholonomic Motion Planning*, Kluwer Academic, 1993.
- [3] J.-P. Laumond (Ed), *Robot Motion Planning and Control*, vol. 229 of *LNCIS*, Springer, 1998.
- [4] Y. Nakamura, *Advanced Robotics: Redundancy and Optimization*, Addison-Wesley, 1991.
- [5] H. Seraji, "A unified approach to motion control of mobile manipulators," *Int. J. of Robotics Research*, vol. 17, no. 2, pp. 107–118, 1998.
- [6] F. G. Pin, K. A. Morgansen, F. A. Tulloch, C. J. Hacker, and K. B. Gower, "Motion planning for mobile manipulators with a non-holonomic constraint using the FSP method," *J. of Robotic Systems*, vol. 13, no. 11, pp. 723–736, 1996.
- [7] J. F. Gardner and S. A. Velinsky, "Kinematics of mobile manipulators and implications for design," *J. of Robotic Systems*, vol. 17, no. 6, pp. 309–320, 2000.
- [8] J.-Y. Fourquet, B. Bayle, and M. Renaud, "Manipulability of wheeled mobile manipulators: Application to motion generation," *Int. J. of Robotics Research*, vol. 22, no. 7-8, pp. 565–581, 2003.
- [9] Y. Yamamoto and X. Yun, "Unified analysis on mobility and manipulability of mobile manipulators," in *1999 IEEE Int. Conf. on Robotics and Automation*, 1999, pp. 1200–1206.
- [10] H. Seraji, "An on-line approach to coordinated mobility and manipulation," in *1993 IEEE Int. Conf. on Robotics and Automation*, 1993, vol. 1, pp. 28–35.
- [11] F. Lamirault, B. Bayle, J.-Y. Fourquet, and M. Renaud, "Kinematic control of wheeled mobile manipulators," in *2002 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2002, pp. 1572–1577.
- [12] B. Bayle, J.-Y. Fourquet, and M. Renaud, "Génération des mouvements des manipulateurs mobiles: Etat de l'art et perspectives," *J. Européen des Systèmes Automatisés*, vol. 35, no. 6, pp. 809–845, 2001.
- [13] A. Liegeois, "Automatic supervisory control of configurations and behavior of multibody mechanisms," *IEEE Trans. on Systems, Man, and Cybernetics*, vol. 7, no. 6, pp. 868–871, 1977.
- [14] A. De Luca and G. Oriolo, "The reduced gradient method for solving redundancy in robot arms," *Robotersysteme*, vol. 7, no. 2, pp. 117–122, 1991.
- [15] P. H. Chang, "A closed-form solution for inverse kinematics of robot manipulators with redundancy," *IEEE J. of Robotics and Automation*, vol. 3, no. 5, pp. 393–403, 1987.
- [16] A. Bettini, A. De Luca, and G. Oriolo, "An experimental comparison of redundancy resolution schemes," in *6th IFAC Symp. on Robot Control*, 2000, pp. 351–356.
- [17] L. Sciavicco and B. Siciliano, *Modelling and Control of Robot Manipulators*, Springer, 2000.
- [18] D. G. Luenberger, *Linear and Nonlinear Programming*, Addison-Wesley, 1984.