Geometric motion control for a kinematically redundant robotic chain: application to a holonomic mobile manipulator

Claudio Altafini*
SISSA/ISAS Int. School for Advanced Studies,
via Beirut 2-4, 34014 Trieste, Italy.
E-mail altafini@sissa.it

January 20, 2003

Abstract

For kinematically redundant robotic manipulators, the extra degrees of freedom available allows freedom in the generation of the trajectories of the end-effector. In this paper, for this scope, we use techniques for motion control of rigid bodies on Riemannian manifolds (and Lie groups in particular) to design workspace control algorithms for the end-effector of the robotic chain and then to pull them back to joint space, all respecting the different geometric structures of the two underlying model spaces. The trajectory planner makes use of geometric splines. Examples of the different kinds of curves that are obtained via the De Casteljau algorithm in correspondence of different metric structures in SE(3) are reported. The feedback module, instead, consists of a Lyapunov based PD controller defined from a suitable notion of error distance on the Lie group. The motivating application of our work is a holonomic mobile manipulator for which simulation results are described in detail.

Keywords: redundant robotic chains, motion control, holonomic mobile robot.

1 Introduction

The main scope of this paper is to explore the use of the geometric methods for trajectory generation [1, 2, 3, 4, 5, 6] and tracking [7, 8, 9] to design more effective workspace motion control algorithms for holonomic mobile manipulators. Like the platform under consideration here, most mobile manipulators will have kinematically redundant degrees of freedom, making it more important to generate the best possible trajectory for the end-effector. Furthermore, while the model space for the joint space is essentially the Euclidean space, the workspace is intrinsically a Lie group, the so-called Special Euclidean group, so that coordinate based methods fail to satisfy any intrinsic criterion of optimality (beside having other well-known problems, like singularities, which are not intrinsic but induced by the parameterization chosen). The tools described in the following are needed to perform motion control in a way coherent with the geometry of the Lie group. In particular, the error between a reference trajectory and a true one has to be defined according to the group operation. Unlike the joint space which is abelian, $SE(3)$ is noncommutative and

*Work done while the author was with the Division of Optimization and Systems Theory, Department of Mathematics, Royal Institute of Technology, SE-10044, Stockholm, Sweden, and supported by the Swedish Foundation for Strategic Research through the Center for Autonomous Systems at KTH

therefore the group operation is "multiplication" (here matrix product). Using exponential maps, it is possible to describe the mapping between the two different model spaces, i.e. to map the "additive" group structure of the joint space to the "multiplicative" one of the workspace. The formalism employed here essentially allows to go in the opposite direction without having to resort to local coordinates for $SE(3)$.

The basic task that is required to the system is to be able to move the end-effector (both rotate and translate) from two given points belonging to the reachable workspace. For an open kinematic chain, the problem of planning a motion is composed of two parts: the first is to find a suitable trajectory for the end effector, the second is to translate this trajectory into a corresponding trajectory for the joints of the manipulator, to be used as reference input to the system via the Euler-Lagrange equations. This is the inverse kinematics in joint space. Alternatively, one could consider the workspace Euler-Lagrange equations (or their reduced version, the Euler-Poincaré equations obtained by "factorizing out" the group symmetry), compute the corresponding workspace generalized torques and map those back to joint space. This last scheme is the one examined in detail. Since a mechanical system is a second order system, its nominal trajectory, in order to be feasible for a control input in the class of piecewise continuous signals, has to be at least C^1. The trajectories considered here for the nominal path of the end-effector are C^1 geometric splines that can be generated either from optimal control or from closed form algorithms. In particular one such closed form methods, called the De Casteljau algorithm, is analyzed in detail expanding previous work [10, 5] in which the algorithm was investigated for generic Riemannian manifolds and compact Lie groups, for which a natural (i.e. biinvariant) Riemannian structure exists. On the obtained trajectory in $SE(3)$, a feedforward workspace generalized torque and a Lyapunov based PD controller are calculated using the tools developed in [7]. The controllers are then pulled back to joint space. Problems of excessive magnitude of the joint torques, connected with passages of the joint variables in proximity of a singularity of the robotic chain, can be taken care of in standard ways.

As mentioned above, the motivating application for this work is a holonomic mobile manipulator, i.e. a robotic arm mounted on the top of a wheeled mobile platform, which provides extra reachability to the end-effector and for which we would like to design a classical two degree of freedom control structure in geometric terms. Compared to the arm, a mobile base is usually slower, coarser and its odometry is subject to drift that obviously propagates through the arm when doing end-effector pose estimate. Furthermore, a mobile manipulator is normally meant to be used for less repetitive and more diverse tasks than a static one. The consequence is that the widely used open loop control schemes based only on inverse kinematics/dynamics and relying only on the joints measurements are usually not enough, and one needs to integrate extra sensor information which is naturally available in workspace. Hence providing extra motivation for workspace control schemes.

The mathematical background is presented in Sec. 2. Section 3 discusses the generation of trajectories on $SE(3)$ and Section 4 the synthesis of a workspace controller, followed by a quick comparison with a joint space controller in Section 5. The application to a holonomic mobile manipulator is treated in Section 6.

2 Riemannian geometry on $SE(3)$

The *Special Euclidean Group* $SE(3)$ is the Lie group of isometric transformations of $\mathbb{R}^3$ and in homogeneous coordinates it is given by

$$SE(3) = \left\{ g \in Gl_4(\mathbb{R}),\ g = \begin{bmatrix} R & p \\ 0 & 1 \end{bmatrix} \text{ s.t. } R \in SO(3) \text{ and } p \in \mathbb{R}^3 \right\}$$

with $SO(3) = \{R \in Gl_3(\mathbb{R}) \ \text{ s.t. } RR^T = I_3 \ \text{ and } \det R = +1\}$ the group of rotations. The Lie algebra of $SE(3)$ is

$$\mathfrak{se}(3) = \left\{ X \in M_4(\mathbb{R}), \ \text{ s.t. } \ X = \begin{bmatrix} \widehat{\omega} & v \\ 0 & 0 \end{bmatrix} \ \text{ with } \widehat{\omega} \in \mathfrak{so}(3) \ \text{ and } \ v \in \mathbb{R}^3 \right\}$$

with $\mathfrak{so}(3) = \{\widehat{\omega} \in M_3(\mathbb{R}) \ \text{ s.t. } \widehat{\omega}^T = -\widehat{\omega}\}$ and $\widehat{\cdot} : \mathbb{R}^3 \to \mathfrak{so}(3)$ such that $\widehat{\omega}\sigma = \omega \times \sigma, \forall\, \sigma \in \mathbb{R}^3$.

The elements of $\mathfrak{se}(3)$ have the physical interpretation of velocities with respect to a choice of frame. In particular, deriving $g \in SE(3)$, the kinematic equations can assume two useful forms:

$$\dot{g} = X^s g \qquad \text{and} \qquad \dot{g} = g\, X^b \qquad X^s,\ X^b \in \mathfrak{se}(3) \tag{1}$$

called respectively *right* and *left invariant representations*. In the right invariant representation, the infinitesimal generator X^s is called the *spatial velocity*

$$X^s = \begin{bmatrix} \widehat{\omega}^s & v^s \\ 0 & 0 \end{bmatrix}$$

because it represents the velocity of g translated to the identity and expressed in an inertial frame. Invariance here is with respect to a matrix multiplication from the right and means invariance to the choice of the body fixed frame. Considering the rotation and translation components of the kinematic equations, the right invariant representation looks like

$$\begin{cases} \dot{R} &= \widehat{\omega}^s R \\ \dot{p} &= \widehat{\omega}^s p + v^s \end{cases}$$

Similarly, the left invariant representation expresses invariance to change of the inertial frame and $X^b \in \mathfrak{se}(3)$

$$X^b = \begin{bmatrix} \widehat{\omega}^b & v^b \\ 0 & 0 \end{bmatrix}$$

is called *body velocity*. The first order kinematic equations are then:

$$\begin{cases} \dot{R} &= R\,\widehat{\omega}^b \\ \dot{p} &= R\, v^b \end{cases} \tag{2}$$

The relation between spatial and body velocity at $g \in SE(3)$ is expressed by the *adjoint map* $\mathrm{Ad}_g(Y) = gYg^{-1}\ \forall\, Y \in \mathfrak{se}(3)$ that gives the change of basis on the Lie algebra: $X^s = \mathrm{Ad}_g(X^b) = gX^bg^{-1}$ and $X^b = \mathrm{Ad}_{g^{-1}}(X^s)$. The derivation of the adjoint map with respect to $g = e^{tX}$, $X \in \mathfrak{se}(3)$, at the identity of the group

$$\mathrm{ad}_X = \left.\frac{d}{dt}\left(\mathrm{Ad}_{e^{tX}}\right)\right|_{t=0}$$

gives the *Lie bracket* $\mathrm{ad}_X(Y) = [X,\, Y] = X\,Y - Y\,X$, i.e. the bilinear form defining the Lie algebra. The Lie brackets of the basis elements $A_1, \ldots, A_6$ of $\mathfrak{se}(3)$ are expressed in terms of the *structural constants* c_{ij}^k : $[A_i,\, A_j] = c_{ij}^k A_k$. The linear representations of the operators $\mathrm{Ad}_g(\cdot)$ and $\mathrm{ad}_X(\cdot)$ is:

$$\mathrm{Ad}_g = \begin{bmatrix} R & 0 \\ \widehat{p}R & R \end{bmatrix} \qquad \mathrm{ad}_X = \begin{bmatrix} \widehat{\omega} & 0 \\ \widehat{v} & \widehat{\omega} \end{bmatrix} \tag{3}$$

For a vector $Y \in \mathfrak{se}(3)$, if $Y \simeq (Y)^\vee = \xi_Y \in \mathbb{R}^6$, then $\mathrm{Ad}_g(Y) = \mathrm{Ad}_g \xi_Y$ and $\mathrm{ad}_X(Y) = \mathrm{ad}_X \xi_Y$, from which it is clear that Ad_g is an algebra homomorphism:

$$\mathrm{Ad}_g[X,\, Y] = [\mathrm{Ad}_g X,\, \mathrm{Ad}_g Y] \quad \text{and} \quad \mathrm{Ad}_{gh} = \mathrm{Ad}_g \mathrm{Ad}_h$$

2.1 Metric properties of $SE(3)$.

The Levi decomposition gives the semidirect product $SE(3) = SO(3) ⓢ \mathbb{R}^3$ with $SO(3)$ compact semisimple and $\mathbb{R}^3$ abelian. Semidirect product means that $SE(3)$, as a manifold, can be considered the direct product $SO(3) \times \mathbb{R}^3$, but its group structure includes the action of $SO(3)$ on $\mathbb{R}^3$ by isometries. For a Lie group, a metric is defined on the Lie algebra and then translated to the whole tangent bundle by left/right translation. Therefore it is automatically left or right invariant, but not necessarily both. Given two left invariant vector fields $X, Y \in \mathfrak{se}(3)$ and $g \in SE(3)$, $\langle gX, gY \rangle = \langle X, Y \rangle$, which means that the metric coefficients M_{ij} are constant with respect to left invariant frames.

On $SO(3)$, by Cartan 2nd criterion, the *Killing form*

$$\begin{aligned} K: \quad \mathfrak{so}(3) \times \mathfrak{so}(3) &\to \mathbb{R} \\ (X, Y) &\mapsto K(X, Y) = \mathrm{tr}(\mathrm{ad}_X \, \mathrm{ad}_Y) \end{aligned}$$

is symmetric, negative definite and Ad-*invariant* (or bi-invariant):

$$K(X, Y) = K\left(\mathrm{Ad}_g X \cdot \mathrm{Ad}_g Y\right) \quad \forall\, g \in SO(3)$$

Therefore, $\langle \cdot, \cdot \rangle = -\alpha K(\cdot, \cdot)$ is a Riemannian metric.

The inner product in $\mathbb{R}^3$ characterizes $SE(3)$ as its uniquely defined group of motions, but does not determines univocally its metric structure. In fact, it is known that in $SE(3)$ there is no Ad-invariant Riemannian metric, which implies that there is no natural way of transporting vector fields between points of $SE(3)$ and that there is no natural concept of distance on $SE(3)$ [11, 12, 13]. The two most common approaches to tackle this obstruction are:

1. Ad-invariant pseudo-Riemannian structure
2. double geodesic.

The first consists in choosing an inner product which is nondegenerate but that can assume both negative and positive values. This corresponds to having curves with both negative and positive energy and gives as geodesics the so-called screw motions. In the second case, instead, the group structure of $SE(3)$ is disregarded in favor of a cartesian product of two distinct groups (rotations and translations). Either choice has advantages and disadvantages, according to the task in mind.

Both metric structures are characterized by a quadratic symmetric matrix so that if $X, Y \in \mathfrak{se}(3)$

$$\langle X, Y \rangle = \left((X)^\vee\right)^T M\, (Y)^\vee \tag{4}$$

From now on, we will often omit the "$\vee$" when considering the $\mathbb{R}^6$-representation of a vector field in $\mathfrak{se}(3)$. The Ad-invariant pseudo-Riemannian structure consists in insisting on the notion of one-parameter subgroups by using the quadratic form in $SE(3)$(combination of Killing form and Klein form)

$$M_{\mathrm{Ad}} = \begin{bmatrix} \alpha I & \beta I \\ \beta I & 0 \end{bmatrix} \tag{5}$$

whose eigenvalues are nondegenerate but can be either positive or negative according to the values of α and β. Its geodesics are the *screw motions.* The double geodesic approach is based on discarding the group structure of $SE(3)$ and consider separately the bi-invariant metric of $SO(3)$ and the Euclidean metric of $\mathbb{R}^3$. The corresponding quadratic form is

$$M_{\mathrm{dg}} = \begin{bmatrix} \alpha I & 0 \\ 0 & \beta I \end{bmatrix} \tag{6}$$

Although a representation of $\mathfrak{se}(3)$ based on disregarding the group structure is neither right nor left invariant, the right (or left) invariance of the metric is preserved. Consider for example the left-invariant representation of the system. Discarding the group structure, we have that the left-invariant equation for $g = (R, p)$ are:

$$\left\{ \begin{array}{rcl} \dot{R} & = & R\widehat{\omega}^b \\ \dot{p} & = & v^b \end{array} \right. \tag{7}$$

Changing body frame from g to $g_0 g$, where $g_0 = (R_0,\, p_0)$ we get

$$g_0 g = (R_0 R,\, R_0 p + p_0)$$

Deriving the two components,

$$\frac{d}{dt}(R_0 R,\, R_0 p + p_0) = R_0(\dot{R},\, \dot{p}) = (R_0 R\widehat{\omega}^b,\, R_0 v^b)$$

we reobtain (7). Applying the same change of body frame in the group structure, left invariance gives

$$\frac{d}{dt}(g_0 g) = g_0 g X^b$$

or eq. (2) in components. The only difference between (7) and (2) is the left action $SO(3) \to \mathrm{Aut}(\mathbb{R}^3)$ given by the rotation R in $\mathbb{R}^3$ which does not modify lengths.

2.2 Riemannian connection on $SE(3)$

The natural affine connection that can be associated to an Ad-invariant nondegenerate symmetric (0, 2)-tensor is called the (0)-connection and is studied by Cartan in [14]. However, since the quadratic form M_{Ad} is nondegenerate but not positive definite, it is not compatible with the standard definition of kinetic energy of a rigid body in $SE(3)$ because of the negative energy that can be associated to M_{Ad} along certain trajectories. Therefore we neglect it and concentrate instead on M_{dg}. Because of the lack of bi-invariance of M_{dg}, its Riemannian connection, call it $\overset{\mathrm{dg}}{\nabla}$, is not among the "canonical" ones studied in the classical literature [14, 15], but rather it can be seen as the torsion-free metric connection of a trivially reductive homogeneous space with respect to the left action on itself and studied accordingly (see for example [16], §13).

The Riemannian connection $\overset{\mathrm{dg}}{\nabla}$, being defined from a left-invariant metric, can be extended to left invariant vector fields on $TSE(3)$ in such a way that it retains the left-invariance property along the coordinate directions of an invariant basis. Calling A_i the elements of an orthonormal basis of left invariant vector fields:

$$\overset{\mathrm{dg}}{\nabla}_{gA_i}(gA_j) = g\overset{\mathrm{dg}}{\nabla}_{A_i} A_j = \Gamma_{ij}^k g A_k \tag{8}$$

for all $g \in SE(3)$. Since the Γ_{ij}^k are not tensorial, in general left invariance of the connection has to be intended with respect to affine transformations, i.e. if X is an infinitesimal affine transformation and ϕ_X the corresponding local one-parameter group of local transformations in $SE(3)$ generated by X (see Prop. 1.4, Ch. VI of [17]):

$$\phi_X\left(\overset{\mathrm{dg}}{\nabla}_Y Z\right) = \overset{\mathrm{dg}}{\nabla}_{\phi_X(Y)} \phi_X(Z) \qquad \forall\, Y,\, Z \in \mathfrak{se}(3)$$

For the left invariant basis, the Christoffel symbols can be expressed as:

$$\Gamma_{ij}^{k} = \frac{1}{2} M^{kl} \left(c_{lj}^{m} M_{mi} + c_{li}^{m} M_{mj} + c_{ij}^{m} M_{ml} \right)$$

From (4), a constant metric quadratic form like M_{dg}, interpreted as an inertia tensor, is a map $M_{\mathrm{dg}} : \mathfrak{se}(3) \to \mathfrak{se}^*(3)$ the dual of $\mathfrak{se}(3)$. Using ad_X^*, the dual of ad_X, defined as $(\mathrm{ad}_X Z; \eta) = (Z; \mathrm{ad}_X^* \eta)$, where $X, Z \in \mathfrak{se}(3)$, $\eta \in \mathfrak{se}^*(3)$ and $(\cdot\,;\,\cdot)$ indicates the $\mathbb{R}$-valued standard pairing between a Lie algebra and its dual, we get

$$\begin{aligned} \langle \mathrm{ad}_X Z, Y \rangle &= (\mathrm{ad}_X Z; M_{\mathrm{dg}} Y) = \\ &= (Z; \mathrm{ad}_X^* M_{\mathrm{dg}} Y) = \langle Z, M_{\mathrm{dg}}^{-1} \mathrm{ad}_X^* M_{\mathrm{dg}} Y \rangle \end{aligned} \tag{9}$$

Proposition 1 *The left-invariant covariant derivative for $(SE(3),\, M_{\mathrm{dg}})$ can be expressed as*

$$\overset{\mathrm{dg}}{\nabla}_X\, Y = \frac{1}{2} \Big([X, Y] - M_{\mathrm{dg}}^{-1} \left(\mathrm{ad}_X^* M_{\mathrm{dg}} Y + \mathrm{ad}_Y^* M_{\mathrm{dg}} X \right) \Big) \tag{10}$$

Proof See Appendix A.1. ■

From (3), the expressions for the coadjoint and infinitesimal coadjoint actions $\mathrm{Ad}_{g^{-1}}^*$ and ad_X^* are:

$$\begin{aligned} \mathrm{Ad}_{g^{-1}}^* &= (\mathrm{Ad}_g)^{-T} = \begin{bmatrix} R & \widehat{p}R \\ 0 & R \end{bmatrix} \\ \mathrm{ad}_X^* &= -\left.\frac{d}{dt} \mathrm{Ad}_{e^{-tX}}^* \right|_{t=0} = -\left(\mathrm{ad}_X \right)^T = \begin{bmatrix} -\widehat{\omega} & -\widehat{v} \\ 0 & -\widehat{\omega} \end{bmatrix} \end{aligned} \tag{11}$$

In (10), when we compute the covariant derivative of X along itself, due to the semidirect action of $SO(3)$ on $\mathbb{R}^3$ the terms $\mathrm{ad}_X^* M_{\mathrm{dg}} X$ are nonnull, even when the inertia tensor is diagonal with $\alpha = \beta$. In this case, M_{dg} can be pulled out and (with abuse of notation)

$$\mathrm{ad}_X^* M_{\mathrm{dg}} X = \alpha \mathrm{ad}_X^* X = \alpha \begin{bmatrix} 0 \\ -\widehat{\omega} v \end{bmatrix} = \alpha \begin{bmatrix} 0 \\ v \times \omega \end{bmatrix} \neq 0$$

Using (8), we can reduce the transport equation from $TSE(3)$ to the Lie algebra $\mathfrak{se}(3)$. The equations we obtain are the so-called *Euler-Poincaré equations* [18]. Consider the left-invariant trajectory $\dot{\gamma}(t) = \gamma(t) X(t)$. In the basis $A_1, \ldots, A_6$, the infinitesimal generator $X \in \mathfrak{se}(3)$ is expressed as $X(t) = \alpha^i(t) A_i$ and its derivative as $\dot{X}(t) = \dot{\alpha}(t)^i A_i$.

Proposition 2 *In $SE(3)$, the Euler-Poincaré equations corresponding to the metric M_{dg} are*

$$\begin{aligned} \dot{\gamma} &= \gamma X \\ \dot{X} &= M_{\mathrm{dg}}^{-1} \mathrm{ad}_X^* M_{\mathrm{dg}} X \end{aligned}$$

Proof See Appendix A.2. ■

3 Motion generation in $SE(3)$

Assuming that the forward kinematics map is surjective, the forced Euler-Poincaré equations, i.e. the second order equations for the robotic chain as seen from the end-effector are fully actuated, i.e. 6 independent control inputs are available for such 6-dimensional mechanical system. This simplifies considerably the trajectory generation problem, as any smooth enough trajectory is feasible for the system and the Euler-Poincaré equations give almost directly the corresponding nominal input f_{FF} (see below).

Here we review two methods used to generate C^1 trajectories on $SE(3)$. The typical situation is as follows: given a number of waypoints in $TSE(3)$, we would like to generate a feasible curve in $TSE(3)$ that interpolates them. The class of admissible inputs considered here is piecewise continuous, bounded measurable signals. Consequently, the minimal requirement for a feasible trajectory is to be globally C^1 (C^0 in the velocity phase space) and piecewise smooth. The loss of smoothness occurs only at the junction points. Let us concentrate on the trajectory generation between two consecutive waypoints characterized by the boundary data $g_0, \; g_f \in SE(3)$ and $\dot{g}_0 \in T_{g_0}SE(3)$, $\dot{g}_f \in T_{g_f}SE(3)$.

3.1 Optimal control approach

The optimal control approach to such problem is to consider a second order variational problem with cost functional the square of the L^2 norm of the acceleration:

$$J = \int_0^T \langle \overset{\mathrm{dg}}{\nabla}_{\dot{\gamma}} \dot{\gamma}, \, \overset{\mathrm{dg}}{\nabla}_{\dot{\gamma}} \dot{\gamma} \rangle \, dt$$

The use of variational techniques for trajectory generation on matrix Lie groups is investigated for example in [19, 4, 6]. The main result for generic Riemannian manifolds is the following theorem.

Theorem 1 ([2, 4]) *A necessary condition for a C^1 curve $\gamma(t) \in SE(3)$, $t \in [0, T]$, interpolating the data above to be an extremum of J is that*

$$\overset{\mathrm{dg}}{\nabla}_{\dot{\gamma}} \overset{\mathrm{dg}}{\nabla}_{\dot{\gamma}} \overset{\mathrm{dg}}{\nabla}_{\dot{\gamma}} \dot{\gamma} + R(\dot{\gamma}, \, \overset{\mathrm{dg}}{\nabla}_{\dot{\gamma}} \dot{\gamma})\dot{\gamma} = 0$$

In the above mentioned literature, it is shown how to construct optimal trajectories from this necessary condition.

3.2 The De Casteljau algorithm on $SE(3)$

Working in $\mathbb{R}^n$, the resulting optimal curves obtained from Theorem 1 correspond to polynomial splines. The same trajectories can also be generated in an analytic way by considering the polygonal support of the curve, using constructions like the Bézier polynomials or schemes like the De Casteljau algorithm [20]. When one tries to extend such methods from $\mathbb{R}^n$ to a noncommutative Lie group, the situation becomes more complicated, as the different constructions no longer coincide. In [5, 10], the generalization of closed form methods like the De Casteljau algorithm is investigated for compact Lie groups, for which a "natural" (i.e. completely frame-independent) Riemannian structure exists. The lack of a positive definite biinvariant metric on $SE(3)$ complicates things further, as the choice of the metric tensor is task (or designer) biased. The scope of this subsection is to extend to the two structures of $SE(3)$ introduced in Section 2.1, Ad-invariant pseudo-Riemannian and double-geodesic, the De Casteljau method. The algorithm used here is taken from [1] and produces a smooth curve connecting two poses in $SE(3)$ with given boundary velocities. The advantage of such an algorithm with respect to the variational approach is that it gives a curve in closed form, function only of the boundary data (and of the metric structure), so that it can be useful in applications in which a (noncausal) trajectory exactly matching the data is required. On the other hand, the obtained trajectories do not seem to be the optimum of any variational problem [10]. The idea is to find a closed form curve $\gamma_r(\cdot) : [0,\, 1] \mapsto SE(3)$ satisfying the boundary conditions:

$$\gamma_r(0) = g_0, \quad \left.\frac{d\gamma_r}{dt}\right|_{t=0} = \dot{g}_0, \quad \gamma_r(1) = g_f, \quad \left.\frac{d\gamma_r}{dt}\right|_{t=1} = \dot{g}_f \tag{12}$$

Here, choosing $T = 1$ as final time is meant to simplify the calculations. If $T \neq 1$, the time axis can be rescaled appropriately afterwards. The basic idea in $\mathbb{R}^n$ is to transform the boundary conditions on the velocity into intermediate points (called "control points"). The combination of the straight line segments connecting the extreme points to the control points gives the desired polynomial. The generalization to a Riemannian manifold consists in substituting the line segments, used for the construction in the Euclidean version, with geodesic arcs. Likewise, a couple of iterated combinations of the geodesics gives a C^∞ curve, which shares the same boundary conditions with the original patching of geodesic arcs.

A sketch of the (left-invariant version of the) algorithm is as follows:

- transform the 1st order boundary values in infinitesimal generators V_0^1 and V_2^1 (see [10] for the details)
$$\left.\frac{d\gamma_r}{dt}\right|_{t=0} = 3 g_0 V_0^1 \qquad \text{and} \qquad \left.\frac{d\gamma_r}{dt}\right|_{t=1} = 3 g_f V_2^1$$
- get the "control points"
$$g_1 = g_0 e^{V_0^1} \qquad \text{and} \qquad g_2 = g_f e^{-V_2^1}$$
  i.e. the points reached by the time-one one-parameter arcs from the extremes.
- using the logarithmic map, find the velocity V_1^1 s.t.
$$g_2 = g_1 e^{V_1^1}$$
  The three velocities obtained so far $V_0^1,\ V_1^1,\ V_2^1\ \in \mathfrak{se}(3)$ are constant. Their combination (through the exponential) gives rise to curves which not anymore correspond to one-parameter subgroups, but keep the same boundary values as the C^0 patch of geodesic arcs.
- construct $V_0^2(t)$ and $V_1^2(t)$ s.t.
$$e^{V_0^2(t)} = e^{(1-t)V_0^1} e^{tV_1^1} \qquad e^{V_1^2(t)} = e^{(1-t)V_1^1} e^{tV_2^1} \qquad t \in [0,\ 1]$$
- construct $V_0^3(t)$ s.t.
$$e^{V_0^3(t)} = e^{(1-t)V_0^2(t)} e^{tV_1^2(t)} \qquad t \in [0,\ 1]$$
  The velocities $V_0^2(t),\ V_1^2(t),\ V_0^3(t)$ are not constant but correspond to "polynomial" generators
- the interpolating curve is
$$\gamma_r(t) = g_0 e^{tV_0^1(t)} e^{tV_0^2(t)} e^{tV_0^3} \qquad t \in [0,\ 1]$$

As in $SO(3)$ and $SE(3)$ the exponential and logarithmic maps have closed form expressions, and as long as the data are given in a symmetric fashion (i.e. we do not have to compute covariant derivatives), the procedure above requires only linear algebra tools plus exp and log maps.

By repeating the procedure, C^1 (piecewise smooth) trajectories can be generated from a sequence of waypoints in $SE(3)$.

4 Workspace controller

While geometric formulations of joint space control of rigid robotic manipulators are quite common, workspace equivalent ones seem to be much more rare. An emblematic example is the book [21]: when it comes to workspace control (Section 5.4) it falls short of all the remarkable geometric methods employed up to that point and develops only a coordinate-based approach. In fact, intrinsic workspace control requires some extra tools from Riemannian geometry and the theory of Lie group to be utilized.

The problem of constructing a controller for a fully actuated system on a Riemannian manifold is treated in [7]. We recall in this Section the main concepts needed and then do the calculations in detail for our case. The resulting controller is composed of a feedforward term corresponding to the reference trajectory $(\gamma_r, \dot{\gamma}_r) \in SE(3)$, as seen from the true one, plus a feedback term constructed from a suitably defined symmetric positive definite function $\phi(\gamma, \gamma_r) : SE(3) \times SE(3) \to \mathbb{R}$, which plays the role of the Lyapunov function. If $(\gamma, \dot{\gamma})$ is the real trajectory followed by the end-effector, the "error distance" ϕ induces a class of maps $\mathcal{T}$ between sections of the tangent bundle, named "transport maps" in [7]

$$\mathcal{T}_{(\gamma,\gamma_r)} : T_\gamma SE(3) \to T_{\gamma_r} SE(3)$$

that allow to map vector fields between $T_\gamma SE(3)$ and $T_{\gamma_r} SE(3)$ in a manner compatible with the one form induced by ϕ on γ and on γ_r

$$\mathrm{d}\phi(\gamma, \gamma_r)|_\gamma = -\mathcal{T}^*_{(\gamma,\gamma_r)}\, \mathrm{d}\phi(\gamma, \gamma_r)|_{\gamma_r}$$

where $\mathcal{T}^*$ is the dual of $\mathcal{T}$. By raising the indexes on both sides, the transport map $\mathcal{T}$ is used to transform the total derivative of ϕ into derivative w.r.t. the first factor

$$\begin{aligned} \frac{d}{dt}\phi(\gamma, \gamma_r) &= \mathrm{d}\phi(\gamma, \gamma_r)|_\gamma\, \dot{\gamma} + \mathrm{d}\phi(\gamma, \gamma_r)|_{\gamma_r}\, \dot{\gamma}_r \\ &= \mathrm{d}\phi(\gamma, \gamma_r)|_\gamma \left(\dot{\gamma} - \mathcal{T}_{(\gamma,\gamma_r)}\dot{\gamma}_r\right) = \mathrm{d}\phi(\gamma, \gamma_r)|_\gamma\, \dot{e} \end{aligned}$$

where $\dot{e} \in T_\gamma SE(3)$ is called (body frame) *velocity error*. The feedforward controller is then given by the covariant derivative along γ

$$f^B_{\mathrm{FF}} = \frac{D}{dt}\left(\mathcal{T}_{(\gamma,\gamma_r)}\dot{\gamma}_r\right) = M_{\mathrm{dg}} \left(\overset{\mathrm{dg}}{\nabla}_{\dot{\gamma}}\, \mathcal{T}_{(\gamma,\gamma_r)}\dot{\gamma}_r \Big|_{\gamma_r \text{ fixed}} + \frac{d}{dt}\left(\mathcal{T}_{(\gamma,\gamma_r)}\dot{\gamma}_r\right)\Big|_{\gamma \text{ fixed}} \right) \tag{13}$$

where the first term is the usual covariant derivative for a vector field (depending on γ) and the second term expresses how the change in γ_r is viewed along γ. A Lyapunov based PD controller for the feedback part is

$$f^B_{\mathrm{PD}} = -\,\mathrm{d}\phi(\gamma, \gamma_r)|_\gamma - K_D\left(\dot{\gamma} - \mathcal{T}_{(\gamma,\gamma_r)}\dot{\gamma}_r\right) \tag{14}$$

4.1 Tracking control on $SE(3)$

The formulation above for the controller does not take advantage of the Lie group structure of the workspace $SE(3)$. In Section 2.2, we saw that when the configuration space is a Lie group, the Euler-Lagrange equations can be reduced to Euler-Poincaré equations by using left invariance. One can always assume that also the reference trajectory γ_r is left invariant: $\dot{\gamma}_r = \gamma_r X_r$ for some $X_r \in \mathfrak{se}(3)$ (obtained, tautologically, by writing $X_r = \gamma_r^{-1}\dot{\gamma}_r$). Due to the semidirect product structure of $SE(3)$, it is not straightforward to express directly the Lie algebra evaluated infinitesimal generators from the motion generation methods illustred in Section 3. Given left invariant reference trajectory and true trajectory $\gamma_r(t)$, $\gamma(t) \in C^\infty(SE(3))$, the *right group error* [8, 9] can be defined as the curve

$$\gamma_e(t) \triangleq \gamma_r^{-1}(t)\gamma(t) \tag{15}$$

The problem of comparing the derivatives of left invariant curves $\dot{\gamma}_r = \gamma_r X_r$ and $\dot{\gamma} = \gamma X$ is simpler than the Riemannian treatment above and is discussed in [8] for compact groups and in [7] for the $SE(3)$ case. It can be formulated as follows: $\dot{\gamma}_r$ and $\dot{\gamma}$ live in $T_{\gamma_r}SE(3)$ and $T_\gamma SE(3)$ respectively. Through left invariance they are both pulled back to the Lie algebra. However, this does not allow to compare them directly. In order to do that, one has to use the adjoint map of γ_e which expresses the change of basis in the Lie algebra, i.e. how one infinitesimal generator X_r "looks like" when the corresponding derivative is parallel transported to γ and then Lie algebra evaluated. For a generic Riemannian manifold, this condition corresponds to the existence of the transport map $\mathcal{T}$. Deriving (15)

$$\begin{aligned}
\dot{\gamma}_e &= \frac{d}{dt}\left(\gamma_r^{-1}\right)\gamma + \gamma_r\frac{d}{dt}\left(\gamma_r\right) = \gamma_r^{-1}\gamma X - \gamma_r^{-1}\dot{\gamma}_r\gamma_r^{-1}\gamma \\
&= \gamma_r^{-1}\gamma\left(X - \mathrm{Ad}_{\left(\gamma_r^{-1}\gamma\right)^{-1}}X_r\right) = \gamma_e\left(X - \mathrm{Ad}_{\gamma_e^{-1}}X_r\right) \triangleq \gamma_e X_e
\end{aligned} \tag{16}$$

i.e. also the trajectory for the right error is left invariant with respect to a suitable infinitesimal generator $X_e \in \mathfrak{se}(3)$. Using the adjoint map as in (16) to transport vectors is, for example, compatible with the following quadratic error function

$$\phi(\gamma, \gamma_r) = \phi(\gamma_e, 0) \triangleq \frac{1}{2}\mathrm{tr}\left(K_1\left(I_3 - R_e\right)\right) + \frac{1}{2}p_e^T K_2 p_e \tag{17}$$

where $\gamma_e = (R_e, p_e) = \left(R_r R,\, R_r^T(p - p_r)\right)$. The eigenvalues k_i of $K_1 = K_1^T$ are chosen such that $k_i + k_j > 0$ for $i \neq j$ and $K_2 > 0$. For a matrix A, call $\mathrm{skew}(A) = A - A^T$ (see [8] or formula (35) and Appendix of [7]).

Proposition 3 *For the Riemannian manifold* $(SE(3), I)$ *the* $\mathfrak{se}(3)$ *evaluated controller* (13)-(14) *corresponding to the right group error* (15) *and the error function* (17) *can be expressed as:*

$$f_{\mathrm{FF}} = -\frac{1}{2}\left(\mathrm{ad}_X \mathrm{Ad}_{\gamma_e^{-1}}X_r + \mathrm{ad}_X^* \mathrm{Ad}_{\gamma_e^{-1}}X_r + \mathrm{ad}^*_{\mathrm{Ad}_{\gamma_e^{-1}}X_r}X\right) + \mathrm{Ad}_{\gamma_e^{-1}}\dot{X}_r \tag{18}$$

$$f_{\mathrm{PD}} = -\begin{bmatrix} \mathrm{skew}(K_1 R_e)^T & R_r^T K_2 p_e \\ 0_{(1\times 3)} & 0 \end{bmatrix} - K_D X_e \tag{19}$$

Proof See Appendix A.3. ■

The necessary condition for the existence of such controller is that the system is fully actuated on $\mathfrak{se}(3)^*$, so that there exists a one-parameter subgroup corresponding to γ_e for all γ and γ_r in $SE(3)$.

4.2 Pull-back of the controller to joint space

When considering the forward kinematics as a map from the joint space, call it $\mathbf{Q}$, to $SE(3)$, the rectangular Jacobian $J(\mathbf{q})$ has the interpretation of velocity gain from the joint space velocity $\dot{\mathbf{q}} \in T\mathbf{Q}$ to the workspace velocities $X_{\rho(\mathbf{q})} \in \mathfrak{se}(3)$. Similarly, the differential inverse kinematics can be thought of as the gain from end-effector generalized forces to joint torques/forces. Pulling back the obtained controller using the Moore-Penrose pseudoinverse, the energy along a workspace trajectory is preserved in joint space. Since $M_{\mathrm{dg}} = I$, we have:

$$\tau_{\mathbf{q}} = J^{\dagger}(\mathbf{q})\,(f_{\mathrm{FF}} + f_{\mathrm{PD}}) \tag{20}$$

In order to deal with large forces exerted in proximity of a singularity, generalized inverse types of solutions can be used. The resulting joint space controller is then

$$\tau_{\mathbf{q}} + \left(I - J^{\dagger}(\mathbf{q})J(\mathbf{q})\right)\tau'_{\mathbf{q}} \tag{21}$$

with $\tau'_{\mathbf{q}} = \mathrm{d}\psi$ the differential of an auxiliary function ψ. Since the inverse kinematics is an immersion map, problems of instability of the joint space zero dynamics can arise in correspondence of functions ψ that take into account only the joint/link variables but not their velocities [22]. One standard solution is then to add an extra damping term in $\tau'_{\mathbf{q}}$ function of the error of the joint space velocities, with respect to a reference joint velocity profile [23]. More on redundacy resolution in geometric terms is discussed in [24].

5 Joint space controller

Alternatively to the scheme above, a controller can be synthesized directly in joint space. Since the joint space $\mathbf{Q}$ has the structure of a vector space, the derivation of a joint space controller is more intuitive and is well-known in the literature. In this case, the reference trajectory $(\gamma_r, \dot{\gamma}_r) \in TSE(3)$ can be pulled back to joint space through the differential inverse kinematics, so that a reference trajectory $(\mathbf{q}_r, \dot{\mathbf{q}}_r) \in T\mathbf{Q}$ is obtained. The joint space acceleration is $\nabla_{\dot{\mathbf{q}}_r}\dot{\mathbf{q}}_r$, with ∇ the Riemannian connection associated to the generalized intertia matrix $M(\mathbf{q})$ of the robotic chain. The corresponding feedforward controller is

$$\tau_{\mathbf{q}_r} = M(\mathbf{q})\nabla_{\dot{\mathbf{q}}_r}\dot{\mathbf{q}}_r \tag{22}$$

or, in coordinates, $(\tau_r)_l = M_{lk}\ddot{\mathbf{q}}_r^k + \Gamma^k_{ij}\dot{\mathbf{q}}_r^i\dot{\mathbf{q}}_r^j$. Unlike $SE(3)$, an Euclidean space comes with an abelian group structure, with addition as group operation. In the case at hand here, it implies that the joint space error looks like $\mathbf{q} - \mathbf{q}_r$ and a feedback controller is easily constructed out of it. Check the existing literature for the feedback schemes normally adopted in this cases. This design is more suited to control methods that rely exclusively on the joint space measurements.

6 Application to a mobile manipulator

The motivating application for this work is a holonomic mobile manipulator, see for example the special issue of the *Journal of Robotic Systems*, November 1996 or the papers [25, 26, 27, 28, 29, 30, 31] for a survey of the state of art on mobile manipulation. The use of a mobile base extends considerably the workspace volume employable in ordinary operations, but it worsens its precision of execution with respect to a static manipulator. Furthermore, if non repetitive tasks are to be performed, it is much more conjecturable that extra sensors for the measurement of workspace quantities have to be available. These facts, together with the practical considerations that mobile manipulators are more likely to have redundant degrees of freedom than their static cousins, make the development of workspace control algorithms more desirable than before.

The mobile manipulator under examination in this work is composed of a 6 degree of freedom robotic arm (Puma 560) mounted on the top of a Nomad XR4000 holonomic mobile base, see Fig. 1. The platform Nomad XR4000 has four independently controlled castor wheels that allow

Figure 1: The mobile manipulator under investigation: Puma 560 arm and Nomad XR4000 mobile platform.

holonomic movements like following a straight line while rotating and, as it is often the case with the existing mobile platforms, the low level control of the wheels (each of them has two actuators) as well as their coordination is not accessible to the user. If one adds that the platform works with a bandwidth of one or two orders of magnitude slower than the arm, the overall consequence is that the dynamics of the base are less accurate than in the arm. Also this fact calls for extra sensors to be integrated in the motion control algorithms and the natural place to do this is the workspace. The control structure that we want to use for the system is a two degree of freedom scheme composed of an open loop module and a tracking controller. The module for the trajectory generation of the whole coordinated system has to work at the lower frequency, as at higher bandwidth only the 6 joints of the arm are accessible. When generating input trajectories for the joints, in order to have a feasible trajectory for the joint forces/torques and not to excite the structural vibrations of the composed system it is important to have smooth enough trajectories for the end-effector, along which to apply inverse kinematic/dynamic schemes. This is accomplished here by using geometric splines on $SE(3)$. The curve is produced by applying the De Casteljau algorithm studied in Section 3.2 which gives a smooth combination of exponentials on the group that is C^1 at the junctions and admits a closed form expression. The numerical construction of a geometric spline is treated in Section 6.2, where a simple method to sample the curve is shown. While the geometric spline is meant to represent the "macroscopic" motion, in order to perform the numerical integration it is convenient to transform it into a composition of piecewise screw motions at the sampling frequency of the controller. In fact, while the spine itself corresponds to a time-varying infinitesimal generator, screw motions represent one-parameter curves, i.e. integral curves of constant vector fields from given initial conditions and are more suited for straightforward numerical integration. Indeed, this is the situation in which the formalism of exponential coordinates fits naturally, since it is based on representing any motion as an (explicit) integral of some linear differential equation from a specified initial condition. Using a similar type of discretization, also the workspace feedback controller can be synthesized in a numerical scheme. This is done in Section 6.3.

6.1 Kinematic structure of the mobile manipulator

Kinematically, a mobile manipulator is an open chain. If the base is holonomic, the motion of the end effector can be easily factorized into the product of rigid body transformations of one degree of freedom joints or links. The base of a holonomic mobile robot can be modeled as two prismatic links plus a rotational joint and the arm with 6 joints. The forward kinematics of a robotic arm is represented by the map

$$\begin{array}{rrcl} \rho: & \mathbf{Q} = (\mathbb{R})^2 \times (\mathbb{T})^7 & \to & SE(3) \\ & \mathbf{q} = [q_1 \dots q_9] & \mapsto & g = \rho(\mathbf{q}) \end{array} \tag{23}$$

where the seven dimensional torus $\mathbb{T}^7$ is composed of seven rotational joints. The first joint of the arm and the rotational joint of the platform are coaxial, therefore, at the kinematic level, only the difference of the two joint variables is considered. Overall, the joint space of the system has 8 *independent* degrees of freedom. Notice, however, that as the two coaxial joints belong to different components, for control purposes it is sometimes necessary to keep them separate (when, for example, the arm and base are controlled at different bandwidths, not in this study).

Since $\dim\mathbf{Q} > \dim SE(3) = 6$, in a generic pose the surjectivity requirement is not a restriction. Furthermore, since the metric M_{dg} splits rotations from translations, it is also possible to define a manifold with boundary, contained in $SE(3)$, in which ρ is always onto. The boundaries concern the vertical directions of $\mathbb{R}^3$, upper and lower limited by the extendibility of the arm.

6.2 Open loop controller along a geometric spline

Given the boundary data (12), we want to produce a vector of control inputs that steers the manipulator from g_0 to g_f respecting the desired velocity conditions $\dot{g}_0$ and $\dot{g}_f$, in case of perfect tracking and of no model error. The algorithm used is then:

1. by means of the De Casteljau algorithm, compute a closed form smooth trajectory for the end-effector in $SE(3)$ from the boundary conditions;

2. interpolate the global trajectory according to a given sampling rate $1/T$;

3. on each sampling interval, compute the corresponding screw motion and the constant velocity that produces it;

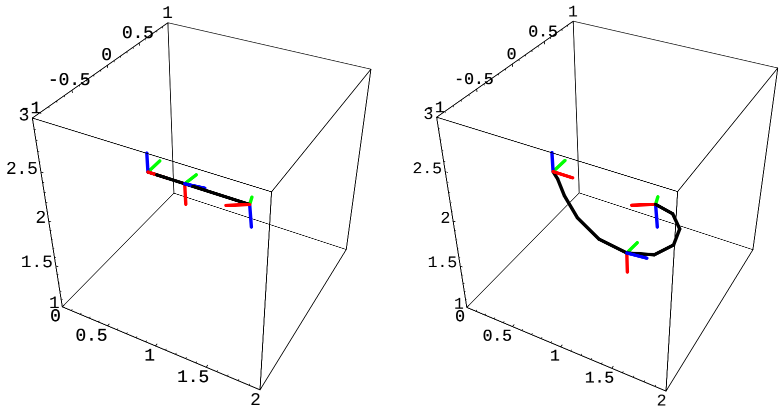
Once we have piecewise constant infinitesimal generators, the trajectory can be used indifferently in workspace or joint space feedforward schemes, obtaining the corresponding generalized torques by means of a discrete derivative. As an example, we discuss the joint space feedforward controller in Section 6.2.3.

6.2.1 Trajectory generation with the De Casteljau algorithm

Simulations results for the $SE(3)$ trajectory resulting from the variational problem of Section 3.1 are extensively treated in [6, 13] and will not be repeated here. Instead, we show some examples of trajectories that can be obtained with the De Casteljau algorithm. The two cases above for the metric of $SE(3)$ lead to different curves because the geodesics are different. In the pseudo-Riemannian case, the velocities V_0^1, V_1^1 and V_2^1 correspond to twists in $\mathfrak{se}(3)$ and are obtained through the exponential and logarithmic maps. In the double-geodesic case, instead, we need to split all the data $g_i \in SE(3)$ into $(R_i, p_i) \in SO(3) \times \mathbb{R}^3$, use the metrics in $SO(3)$ and $\mathbb{R}^3$ to construct the various $\widehat{\omega}_j^i \in \mathfrak{so}(3)$ and $v_j^i \in \mathbb{R}^3$, and then recombine them in homogeneous representations

$$\left(\widehat{\omega}_j^i, v_j^i\right) \mapsto \begin{bmatrix} \widehat{\omega}_j^i & v_j^i \\ 0 & 0 \end{bmatrix} \in \mathfrak{se}(3)$$

The curves one obtain in the two cases are different. In particular, in the double geodesic case, it is possible to maintain the idea of straight line independent of the amount of the rotation, provided that the difference of the two end positions p_0 and p_f is aligned with the direction of both boundary tangent vectors of the translational part, see Fig. 2 (a). The price to pay is that the curve is not left-invariant with respect to a coordinate transformation, since we "forget" about a rotation in the Euclidean part. In general, the more consistent is the rotation component of the desired motion (with respect to the translational part), the more the two curves will look different (compare Fig. 3 and Fig. 4).

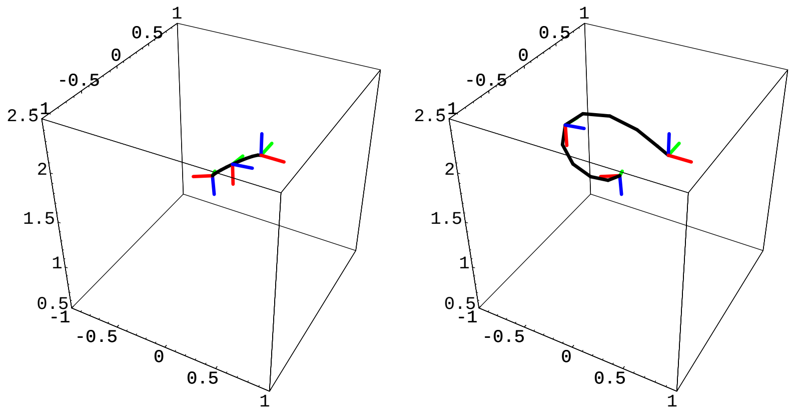


(a) Double geodesic curve (b) Ad-invariant curve

Figure 2: The two curves generated by the De Casteljau algorithm in the special situation in which the translational components of the boundary data lie on the same direction of the translational boundary velocities.

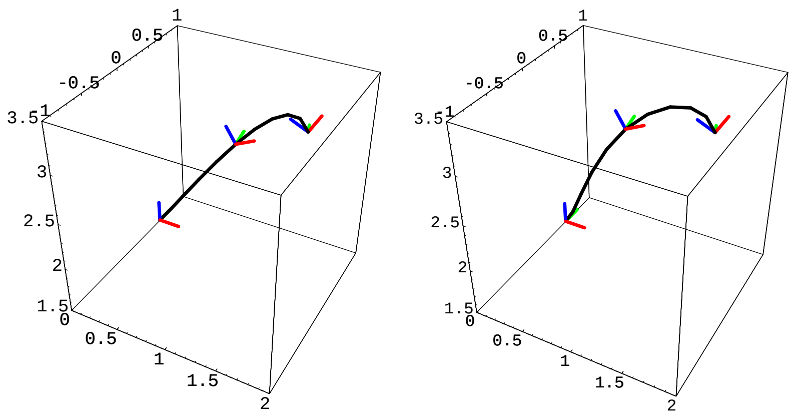
6.2.2 Sample and Hold in $SE(3)$: piecewise screw motion curve

Sampling the workspace kinematics with frequency $\frac{1}{T}$ means setting a tangent vector at the sampling time kT and maintaining it constant until $(k+1)T$. The "macroscopic" trajectory $\gamma_r(t)$ has to be decomposed accordingly, taking into account the group structure of $SE(3)$. At kT, the reference end-effector pose $\gamma_r(kT)$ is known from the previous iteration and the desired pose at $(k+1)T$, $\gamma_r(kT+T)$, can be calculated from the closed form trajectory generator algorithm. The constant infinitesimal generator of the one-parameter curve connecting $\gamma_r(kT)$ and $\gamma_r(kT+T)$ can



(a) Double geodesic curve (b) *Ad*-invariant curve

Figure 3: The two curves generated by the De Casteljau algorithm in a generic case.



(a) Double geodesic curve (b) *Ad*-invariant curve

Figure 4: The two curves generated by the De Casteljau algorithm in another generic case.

be obtained through the logarithmic map:

$$V_k = \log_{SE(3)} \left(\gamma_r^{-1}(kT) \gamma_r(kT+T) \right) \tag{24}$$

so that, parameterizing the screw motion by $\theta \in [0,\ 1]$ we get (see Figure 5):

$$\gamma_r(kT + \theta T) = \gamma_r(kT) e^{V_k \theta T} \quad \theta \in [0,\ 1] \tag{25}$$

V_k can be thought of as a Lie algebra evaluated discrete derivative operator and the "sampling" scheme applied here resembles a first order sample and hold. The main difference, however, with respect to the classical first order sample and hold is that as $\gamma_r(kT+T)$ is given, the "prediction" step is not open loop but it is controlled by the desired reference trajectory. The trajectory resulting after the sampling is a composition of parts of one-parameter curves i.e. a continuous piecewise screw motion trajectory.

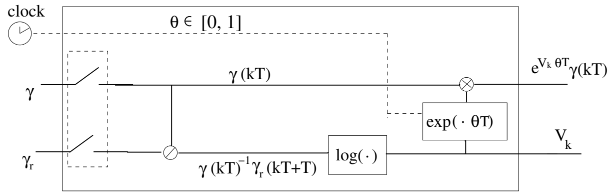


Figure 5: First order sample and hold on the Lie group generating a piecewise screw motion.

6.2.3 Numerical algorithm for the joint space feedforward controller

The inverse kinematic and dynamic maps between joint space and workspace need to be solved numerically for redundant manipulators. The question of how to discretize a geometric structure requires special care and is treated for example in the papers [32, 33] for mechanical structures on Lie groups. Here we discuss briefly only the simplest case, the discretization of the differential inverse kinematics. For the joint space open loop control scheme, the Algorithm of Section 6.2 must be completed by the following steps:

4. invert the system along each screw motion.

5. compute the feedforward generalized torques from (22).

Once the initial conditions $\gamma(0)$ and the corresponding zeros of the joint angle vector $\mathbf{q}$ are fixed, the joint coordinates can be obtained by numerical integration

$$\mathbf{q}_{k+1} = \mathbf{q}_k + (J(\mathbf{q}))^{\dagger} V_k T \tag{26}$$

An extra weight can be added to the joints: the resulting weighted pseudoinverse solution can for example reflect the different inertia of the two parts of the system. All the well-known redundancy resolution methods can be used to deal with the rectangular Jacobian, see [34, 35].

The joint velocity corresponding to V_k is $\dot{\mathbf{q}}_k = J^{\dagger}(\mathbf{q})V_k$ and the corresponding acceleration can be obtained by a further discretization (this time on the vector space $T_{\mathbf{q}}\mathbf{Q}$) and the reference torques then from (22).

6.2.4 Simulations

We saw above that the different metric structures result in different splines curves. A similar example to the one of Fig. 2 of the different γ_r obtained for the same boundary data in the two cases is shown in Figure 7 and Figure 8. Both inverse kinematics are computed considering the general least square version of (26), with the condition number as secondary objective function. Notice the difference in the base trajectories between Figure 7 and Figure 8.

6.3 Workspace control

In this Section we want to show how the workspace feedback controller of Proposition 3 behaves for the mobile manipulator, using as open loop trajectory the same type of spline of Section 6.2.1.

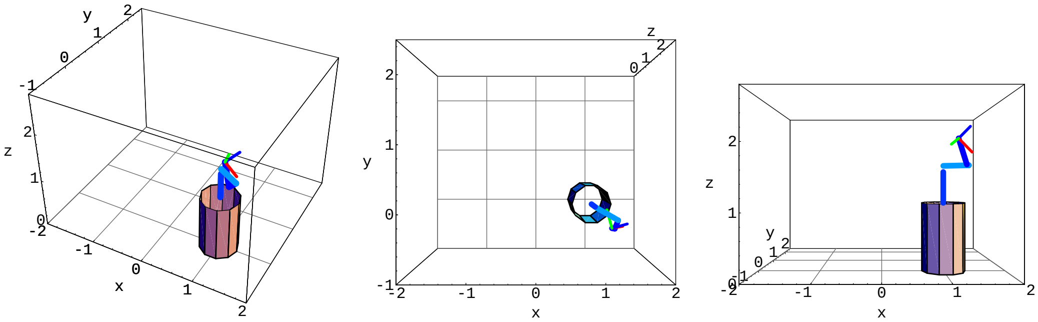


Figure 6: Initial configuration.

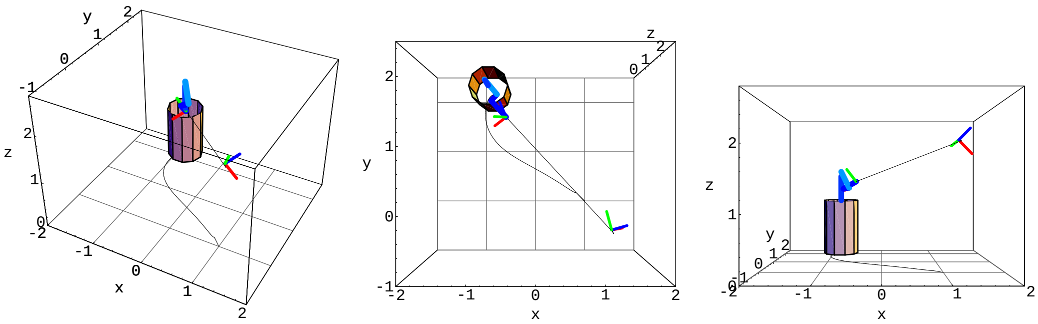


Figure 7: Nominal motion corresponding to the double geodesic metric.

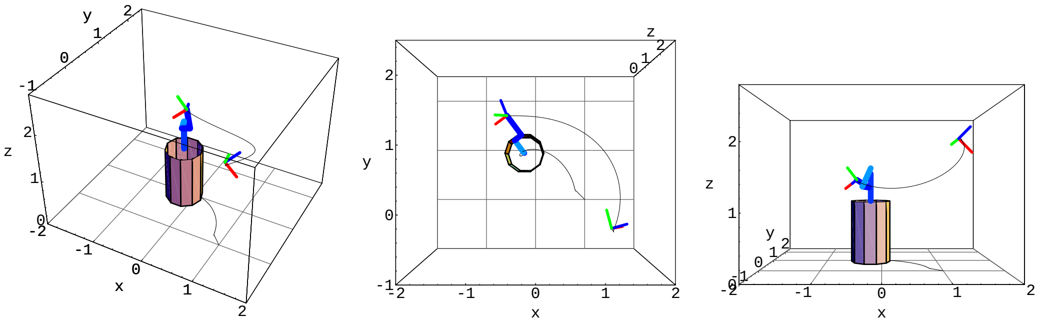


Figure 8: Nominal motion corresponding to the Ad-invariant metric.

Figure 9 gives the pictorial representation of the group error (15) and of why we need to use the transport map. In order to transform (18) and (19) into a numerical algorithm, the same type of discretization explained in Section 6.2.2 can be used for the vector fields on $SE(3)$. Formulae (24) and (25) give already the Lie algebra evaluated infinitesimal generators. The computation of $\dot{X}_r$ in (18) requires the second order discretization to be carried out on the "Lie group side" of the forward kinematics. Since $\mathfrak{se}(3)$ is a vector space, then a naive discretization is possible and this is what we use here. However, as mentioned above, since we have a mechanical system on a Lie group, a more geometric numerical algorithm should be used, like those described in [32, 33]. Once f_{FF} and f_{PD} are discretized, they can be mapped back to joint space via (20)-(21) without extra complications.

6.3.1 Simulations

In Figure 10, the combined effect of f_{FF} and f_{PD} is shown along the same type of spline used in Figure 7. Notice that the high gain chosen here makes the error to be recovered very promptly with quite an abrupt movement of the end-effector that propagates down to the platform. Afterwards, only the feedforward controller essentially drives the system and in fact also the trajectory of the base becomes smooth as it is desirable. While the reaction of the feedback component can be reduced on the end-effector by tuning K_1, K_2 and K_D, its effect on the base can be independently reduced or eliminated by changing the weights on the pseudoinversion.

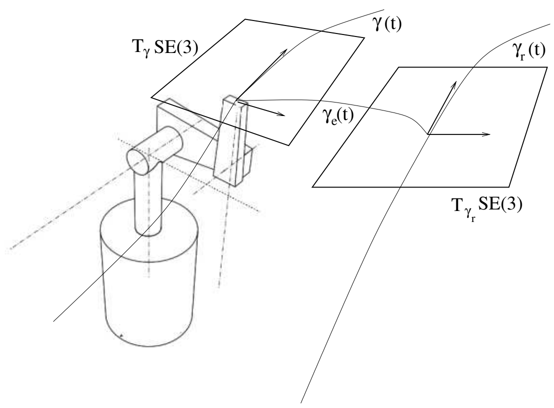


Figure 9: Reference, true and error trajectory for a workspace feedback controller.

7 Conclusion

Given a redundant robotic chain, a workspace controller can be synthesized for the Euler-Lagrange equations that can be associated to the end-effector on $SE(3)$. The global geometric perspective implies that the use of local coordinates in the description of rigid body motion of the end-effector can be avoided. The use of workspace control algorithms is meant to exploit more in depth the Riemannian structure of the workspace, including its Lie group symmetry, for both motion planning and stabilization purposes. Furthermore, from a more practical point of view, it provides a framework in which extra sensor information (other than joint measurements) can be easily inserted and force control easily performed. The main advantage of the geometric workspace control is, however, that it allows to design (and stabilize) in a natural way any type of end-effector trajectories, thus making full use of the redundant degrees of freedom of the robot.

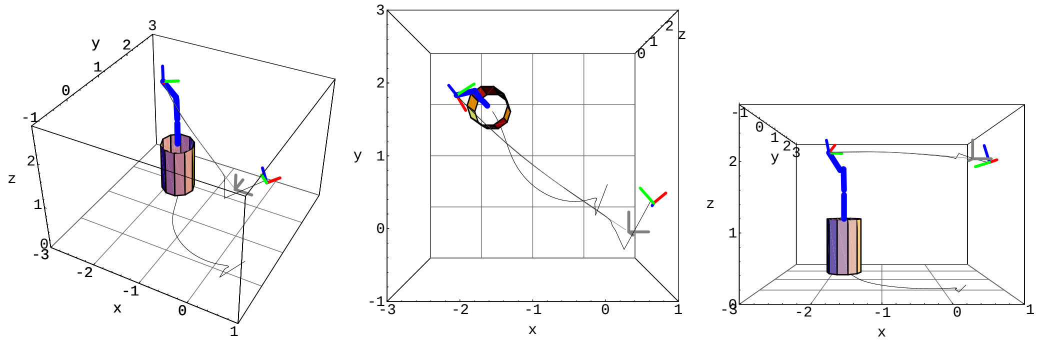


Figure 10: Feedforward and PD controls for a reference trajectory (dotted line) like that of Fig. 2.

A Proofs

A.1 Proof of Proposition 1

Adapting Theorem 3.3, Chapter X of [17] to the case of trivially reductive homogeneous spaces given by the left action of a Lie group on itself, the Riemannian connection for M_{dg} is expressed as

$$\nabla_X Y = \frac{1}{2}[X,\, Y] + U(X,\, Y)$$

where $U(X,\, Y)$ is the symmetric bilinear mapping $\mathfrak{se}(3) \times \mathfrak{se}(3) \to \mathfrak{se}(3)$ defined by

$$\langle U(X,\, Y),\, Z\rangle = \frac{1}{2}\left(\langle \mathrm{ad}_Z X,\, Y\rangle + \langle X,\, \mathrm{ad}_Z Y\rangle\right) \tag{27}$$

for all $X,\, Y,\, Z\, \in \mathfrak{se}(3)$. Using (9) to extract Z on both terms on the right hand side expression (27) the result follows (see also [36] Appendix 2). ■

A.2 Proof of Proposition 2

Call $B_i = L_{\gamma *} A_i = \gamma A_i$, $i = 1, \ldots, 6$ the basis of $T_\gamma SE(3)$ obtained by left translation of $A_1, \ldots, A_6$. Since $X = X(t)$ is time varying, (8) cannot be applied directly. The expression for the transport equation along $\gamma(t)$ is

$$\begin{aligned}
0 = \frac{D}{dt}(\dot\gamma) &= \overset{\mathrm{dg}}{\nabla}_{\dot\gamma(t)}\, \dot\gamma(t) = \overset{\mathrm{dg}}{\nabla}_{\gamma X(t)}\, \gamma X(t) = \overset{\mathrm{dg}}{\nabla}_{\alpha^i(t)B_i}\, \alpha^j(t) B_j \\
&= \left(\mathcal{L}_{\dot\gamma}\alpha^j\right) B_j + \alpha^i\alpha^j\, \overset{\mathrm{dg}}{\nabla}_{B_i}\, B_j \\
&= \dot\alpha^j B_j + \alpha^i\alpha^j\gamma\, \overset{\mathrm{dg}}{\nabla}_{A_i}\, A_j && \text{by (8)} \\
&= \gamma\dot\alpha^j A_j + \frac{1}{2}\gamma\alpha^i\alpha^j \Big([A_i, A_j] - M_{\mathrm{dg}}^{-1}\mathrm{ad}^*_{A_i} M_{\mathrm{dg}} A_j \\
&\qquad - M_{\mathrm{dg}}^{-1}\mathrm{ad}^*_{A_j} M_{\mathrm{dg}} A_i \Big) && \text{by (10)} \\
&= \gamma\dot\alpha^j A_j + \frac{1}{2}\gamma \Big([\alpha^i A_i, \alpha^j A_j] - M_{\mathrm{dg}}^{-1}\mathrm{ad}^*_{\alpha^i A_i} M_{\mathrm{dg}}\alpha^j A_j - \\
&\qquad - M_{\mathrm{dg}}^{-1}\mathrm{ad}^*_{\alpha^j A_j} M_{\mathrm{dg}}\alpha^i A_i \Big)
\end{aligned}$$

Therefore

$$\overset{\mathrm{dg}}{\nabla}_{\dot\gamma(t)}\,\dot\gamma(t)=\gamma\left(\dot X+\overset{\mathrm{dg}}{\nabla}_X\,X\right)=0 \tag{28}$$

Expression (28) is well-known in the literature, see ex. Hermann [37] p.429. The result follows from it. ■

A.3 Proof of Proposition 3

The expression for the controller (18) - (19) could be obtained directly from [7]. However, for sake of clarity and continuity of exposition, we rather reobtain such result. While $\dot\gamma_e\in T_{\gamma_e}SE(3)$, the transport map has to be evaluated along the γ curve:

$$\dot e=\dot\gamma-\mathcal{T}_{(\gamma,\gamma_r)}\dot\gamma_r=\gamma X_e=\gamma_r\dot\gamma_e\in T_\gamma SE(3)$$

Therefore, from (16), $\mathcal{T}\dot\gamma_r=\gamma\mathrm{Ad}_{\gamma_e^{-1}}X_r$ and

$$\overset{\mathrm{dg}}{\nabla}_{\dot\gamma}\,\mathcal{T}\dot\gamma_r\Big|_{\gamma_r\text{ fixed}}=\overset{\mathrm{dg}}{\nabla}_{\gamma X}\,\gamma\mathrm{Ad}_{\gamma_e^{-1}}X_r$$

Calling $Y_r=\mathrm{Ad}_{\gamma_e^{-1}}X_r\in\mathfrak{se}(3)$ the infinitesimal generator X_r as seen from γ and considering the left invariant basis $A_1,\dots,A_6$ of $\mathfrak{se}(3)$, X and Y_r admit the coordinate expressions $X=a^iA_i$ and $Y_r=b^iA_i$ or, in the basis $B_1,\dots B_6$ of $T_\gamma SE(3)$, $\gamma X=\gamma a^iA_i=a^iB_i$ and $\gamma Y_r=\gamma b^iA_i=b^iB_i$.

$$\begin{aligned}\overset{\mathrm{dg}}{\nabla}_{\dot\gamma}\,\mathcal{T}\dot\gamma_r\Big|_{\gamma_r\text{ fixed}}&=\left(\mathcal{L}_{\dot\gamma}b^k+a^ib^j\Gamma^k_{ij}\right)B_k=\gamma\left(\frac{db^k}{dt}A_k+\overset{\mathrm{dg}}{\nabla}_X\,Y_r\right)=\gamma\left(\dot Y_r+\overset{\mathrm{dg}}{\nabla}_X\,Y_r\right)\\&=\gamma\left(\frac{d}{dt}\left(\mathrm{Ad}_{\gamma_r^{-1}\gamma}X_r\right)+\overset{\mathrm{dg}}{\nabla}_X\,\mathrm{Ad}_{\gamma_r^{-1}\gamma}X_r\right)\Big|_{\gamma_r\text{ fixed}}\end{aligned}$$

Using formula (9.3.6) of [18]

$$\frac{d}{dt}\left(\mathrm{Ad}_{\gamma_r^{-1}\gamma}X_r\right)\Big|_{\gamma_r\text{ fixed}}=\mathrm{Ad}_{\gamma^{-1}}\left(-[\mathrm{Ad}_\gamma X,\,\mathrm{Ad}_{\gamma_r}X_r]+0\right)=-\mathrm{ad}_X\mathrm{Ad}_{\gamma_e^{-1}}X_r$$

and using equation (10):

$$\overset{\mathrm{dg}}{\nabla}_X\,\mathrm{Ad}_{\gamma_r^{-1}\gamma}X_r\Big|_{\gamma_r\text{ fixed}}=\frac{1}{2}\left(\mathrm{ad}_X\mathrm{Ad}_{\gamma_e^{-1}}X_r-M_{\mathrm{dg}}^{-1}\mathrm{ad}^*_XM_{\mathrm{dg}}\mathrm{Ad}_{\gamma_e^{-1}}X_r-M_{\mathrm{dg}}^{-1}\mathrm{ad}^*_{\mathrm{Ad}_{\gamma_e^{-1}}M_{\mathrm{dg}}X_r}X\right)$$

The second term in (13) is

$$\frac{d}{dt}\left(\mathcal{T}\dot\gamma_r\right)\Big|_{\gamma\text{ fixed}}=\frac{d}{dt}\left(\dot\gamma_r\gamma_e\right)=\frac{d}{dt}\left(\gamma\mathrm{Ad}_{\gamma_e^{-1}}X_r\right)=\gamma\mathrm{Ad}_{\gamma_e^{-1}}\dot X_r$$

from (9.3.4) of [18]. Concerning the feedback controller, using left invariance of $\dot e=\gamma X_e$, the real valued total derivative $\frac{d}{dt}\phi$ uniquely defines the covector $\eta\in\mathfrak{se}^*(3)\simeq\mathbb{R}^6$ corresponding to $\mathrm{d}\phi|_\gamma$ as $\eta=\mathrm{d}\phi|_\gamma\,\gamma$ so that

$$\begin{aligned}\frac{d}{dt}\phi&=\mathrm{d}\phi|_\gamma\,\dot e=\eta X_e=\frac{1}{2}\mathcal{L}_{\dot\gamma}\left(\mathrm{tr}\left(K_1(I_3-R_r^TR)\right)+(p-p_r)^TR_rK_2R_r^T(p-p_r)\right)\\&=\mathrm{skew}\left(K_1R_r^TR\right)^T\widehat\omega_e+p_e^TK_2R_rv_e=\left[\left(\mathrm{skew}\left(K_1R_r^TR\right)^T\right)^\vee\quad p_e^TK_2R_r\right]X_e^\vee\end{aligned}$$

From (14), the derivative part of the controller $K_D\dot e=\gamma K_DX_e$ is already left invariant. Since $M_{\mathrm{dg}}=I$, and f^B_{FF} is left invariant the expression (18) follows. Furthermore, lowering indices on η leaves η invariant therefore also (19) follows. ■

References

[1] C. Altafini. The De Casteljau algorithm in SE(3). In A. Isidori, F. Lamnabhi-Lagarrigue, and W. Respondek, editors, *Nonlinear Control in Year 2000*, Lecture notes in control and information sciences. Springer-Verlag, London, UK, 2000.

[2] P. E. Crouch and F. Silva Leite. The dynamic interpolation problem on Riemannian manifolds, Lie groups and symmetric spaces. *Journal of Dynamical and Control Systems*, 1(2):177–202, 1995.

[3] Q.J. Ge and B. Ravani. Geometric construction of Bézier motions. *ASME Journal of mechanical design*, 116:749–755, 1994.

[4] L. Noakes, G. Heinzinger, and B. Paden. Cubic splines on curved spaces. *IMA Journal of Mathematics, Control and Information*, 12(6):465–473, 1989.

[5] F.C. Park and B. Ravani. Bézier curves on Riemannian manifolds and Lie groups with kinematic applications. *ASME Journal of mechanical design*, 117(1):36–40, 1995.

[6] M. Zefran, V. Kumar, and C.B. Croke. On the generation of smooth three-dimensional rigid body motions. *IEEE Trans. on Robotics and Automation*, 14(4):576–589, 1998.

[7] F. Bullo and R. Murray. Tracking for fully actuated mechanical systems: a geometric framework. *Automatica*, 35:17–34, 1999.

[8] D Koditschek. Application of a new Lyapunov function to global adaptive attitude tracking. In *Proc. of the 27th IEEE Conf on Decision and Control*, pages 186–191, Austin, TX, 1988.

[9] J.T. Wen and K. Kreutz-Delgado. The attitude control problem. *IEEE Trans. on Automatic Control*, 36(10):1148–1162, 1991.

[10] P.E. Crouch, F. Silva Leite, and G. Kun. Geometric splines. In *Proc. 14th IFAC World Congress*, volume D, pages 533–538, Beijing, China, July 1999.

[11] R.W. Brockett. Some mathematical aspects of robotics. In J. Baillieul, editor, *Robotics*, Proceedings of Symposia in Applied Mathematics. American Mathematical Society, Providence, 1990.

[12] F.C. Park. Distance metrics on the rigid body motions with applications to mechanisms design. *ASME Journal of mechanical design*, 117(1):48–54, 1995.

[13] M. Zefran, V. Kumar, and C.B. Croke. Metrics and connections for rigid body kinematics. *International Journal of Robotics Research.*, 18(2):243–258, 1999.

[14] E. Cartan. La géometrie des groupes de transformations. In *Œuvres Complètes*, volume 2, part I, pages 673–792. Gauthier-Villars, Paris, France, 1953.

[15] L.P. Eisenhart. *Riemannian geometry.* Princeton University Press, Princeton, 1966.

[16] K. Nomizu. Invariant affine connections on homogeneous spaces. *American Journal of Mathematics*, 76:33–65, 1954.

[17] S. Kobayashi and K. Nomizu. *Foundations of differential geometry*, volume I and vol. II. Interscience Publisher, New York, 1963 and 1969.

[18] J.E. Marsden and T.S. Ratiu. *Introduction to mechanics and symmetry*, volume 17 of *Texts in Applied Mathematics*. Springer-Verlag, 2nd edition, 1999.

[19] M. Camarinha, F. Silva Leite, and P.E. Crouch. Second order optimality conditions for an higher order variational problem on a Riemannian manifold. In *Proc. 35th Conf. on Decision and Control*, pages 1636–1641, Kobe, Japan, December 1996.

[20] G. Farin. *Curves and surfaces for computer-aided geometric design: a practical guide*. Academic Press, 4th edition, 1997.

[21] R.M. Murray, Z. Li, and S. Sastry. *A Mathematical Introduction to Robotic Manipulation*. CRC Press, 1994.

[22] P. Hsu, J. Hauser, and S. Sastry. Dynamic control of redundant manipulators. *Journal of Robotic Systems*, 6:133–148, 1989.

[23] A. De Luca. Zero dynamics in robotic systems. In C.I. Byrnes and A. Kurzhanski, editors, *Nonlinear Synthesis*. Birkhäuser, Boston, 1991.

[24] C. Altafini. Redundant robotic chains on Riemannian manifolds. In *Proc. of the 41st Conference on Decision and Control*, pages 1534–1539, Las Vegas, NV, December 2002.

[25] M. Egerstedt and X. Hu. Coordinated trajectory following for mobile manipulation. In *Proc. of the 2000 IEEE Int. Conf. on Robotics and Automation*, pages 3479–3484, San Francisco, CA, 2000.

[26] O. Khatib. Mobile robotic manipulation. In *Proc. 26th International Symposium on Industrial Robots*, pages 7–12, Edmonds, UK, 1995.

[27] H. Seraji. A unified approach to motion control of mobile manipulators. *International Journal of Robotics Research*, 17(2):107–118, 1998.

[28] T. G. Sugar and V. Kumar. Control of cooperating mobile manipulators. *IEEE Trans. on Robotics and Automation*, 18(1):94–103, 2002.

[29] H. G. Tanner and K. J. Kyriakopoulos. Nonholonomic motion planning for mobile manipulators. In *Proc. of the 2000 IEEE Int. Conf. on Robotics and Automation*, pages 1233–1238, San Francisco, CA, 2000.

[30] K. Watanabe, K. Sato, K. Izumi, and Y. Kunitake. Dynamic control for an holonomic omnidirectional mobile manipulator. In S.G. Tzafestas and G. Schmidt, editors, *Progress in System and Robot Analysis and Control Design*, Lecture notes in control and information sciences. Springer-Verlag, London, UK, 1999.

[31] Y. Yamamoto and X. Yun. Coordinating locomotion and manipulation of a mobile manipulator. *IEEE Trans. on Automatic Control*, 39(6):1326–1332, 1994.

[32] J.E. Marsden, S. Pekarsky, and S. Shkoller. Discrete Euler-Poincaré and Lie-Poisson equations. *Nonlinearity*, 12:1647–1662, 1999.

[33] J.E. Marsden and J.M. Wendlandt. Mechanical systems with symmetry, variational principles and integration algorithms. In M. Alber, B. Hu, and J. Rosenthal, editors, *Current and Future Directions in Applied Mathematics*, pages 219–261. Birkhäuser, 1997.

[34] Y. Nakamura. *Advanced Robotics: Redundancy and Optimization.* Addison-Wesley, Reading, MA, 1991.

[35] L. Sciavicco and B. Siciliano. *Modeling and Control of Robot Manipulators.* McGraw-Hill, New York, NY, 1996.

[36] V. I. Arnold. *Mathematical methods of Classical Mechanics*, volume 60 of *Graduate texts in mathematics.* Springer-Verlag, 2nd edition, 1989.

[37] R. Hermann. *Differential geometry and the calculus of variations.* Academic Press, New York, NY, 1968.