Short Papers

A New Trajectory Deformation Algorithm Based on Affine Transformations

Quang-Cuong Pham and Yoshihiko Nakamura

Abstract—We propose a new approach to deform robot trajectories based on affine transformations. At the heart of our approach is the concept of affine invariance: Trajectories are deformed in order to avoid unexpected obstacles or to achieve new objectives but, at the same time, certain definite features of the original motions are preserved. Such features include, for instance, trajectory smoothness, periodicity, affine velocity, or more generally, all affine-invariant features, which are of particular importance in human-centered applications. Furthermore, this approach enables one to "convert" the constraints and optimization objectives regarding the deformed trajectory into constraints and optimization objectives regarding the matrix of the deformation in a natural way, making constraints satisfaction and optimization substantially easier and faster in many cases. As illustration, we present an application to the transfer of human movements to humanoid robots while preserving equiaffine velocity, a well-established invariant of human hand movements. Building on the presented affine deformation framework, we finally revisit the concept of trajectory redundancy from the viewpoint of group theory.

Index Terms—Biomimetics, cognitive human–robot interaction, humanoid robots, kinematics, path planning for manipulators.

I. INTRODUCTION

In order to deal with unforeseen obstacles or perturbations of the target or of the robot's state, it is sometimes more advantageous to *deform* a previously planned trajectory rather than to recompute entirely a new one [2], [3].[1] In motion-capture-based applications, deforming previously recorded trajectories—e.g., to adapt them to a different environment, to retarget them to a different character [4]–[6], or to transfer them to a humanoid robot [7], [8]—is the only viable option, for one cannot reasonably record beforehand all the motions with the desired kinematic and dynamic properties.

A fundamental requirement for trajectory deformation methods is that they should *preserve the characteristic features* of the original trajectory. Such features may include trajectory smoothness, periodicity, optimality, etc., or—in human-centered

applications—*human-characteristic* properties so that robot motions obtained from motion transfer are perceived in a favorable way by human users, or that retargeted animations retain their natural, human-like, expressions.

A. Affine Invariance in Human Action and Perception

The inverse relationship between curvature and linear velocity is a remarkable property of human motions. In a drawing task, for instance, the velocity of the hand tends to be lower in curved portions of the trajectory and higher in the straight portions. This law has been quantified by the *two-third power law* : In planar drawing movements, the angular velocity (a) of the hand and the trajectory curvature (κ) were shown to be related by $a(t) = \gamma\kappa(t)^{2/3}$, with γ being a constant or piecewise constant [9]–[11]. Alternatively, this law can also be written as $v(t) = \gamma\kappa(t)^{1/3}$, where v is the *linear* velocity of the hand. Other types of movements, such as locomotion [12], [13] or smooth eye pursuit [14], have also been found to obey this law.

It was then observed that a planar motion obeying the two-third power law has in fact a *constant equiaffine linear velocity* [15]. This enabled extending the planar two-third power law to 3-D movements: It was shown that 3-D hand movements indeed display a roughly constant equiaffine velocity [16]–[18]. From a theoretical perspective, the two-third power law and its generalization, the law of constant equiaffine velocity, were conceptualized within the broader framework of *affine invariance*: Trajectories generated by humans were understood as being invariant with respect to certain groups of transformations— affine, equiaffine or Euclidean (the latter two being subgroups of the former) [13]. Specifically, it was argued that the three types of invariance—affine, equiaffine, and Euclidean—are present at specific degrees according to the type and context of the movements.

Affine invariance (from now on and unless otherwise specified, the term "affine invariance" shall also include equiaffine invariance, which can be seen as a more restricted case of affine invariance) has also been found in human perception: Motions obeying the power law (or with constant equiaffine velocity) are perceived as more uniform than motions with constant Euclidean velocity [11]; movement prediction in a handwriting task was shown to be facilitated by trajectories obeying the power law [19], etc. This may be related to the action–perception coupling hypothesized for human motor control and may have an underlying neural basis: For instance, the activity of neurons from the motor cortex of a rhesus monkey was shown to follow the two-third power law during a hand drawing task [20]. An

[1]This paper is a substantially revised and expanded version of [1], which was presented at the conference *Robotics: Science and Systems*, 2012.

fMRI study showed that the response of the brain to the visual perception of motions that obey the power law was stronger and more widespread than that of other types of motions [21], in particular in the areas involved in the perception of human motions such as the left premotor cortex [22].

B. Existing Approaches to Trajectory Deformation in Robotics

Several approaches exist for deforming robot trajectories. In *spline-based* approaches, a deformation is made by altering the coefficients multiplying the basis splines [23] or by adding to the original trajectory a displacement map—which is a sum of splines [5], [6]. These modifications can, furthermore, be done in a coarse-to-fine manner using wavelet bases [23] or through hierarchical approximations [6].

Another approach is based on the encoding of the original trajectory by an *autonomous nonlinear dynamical system* [8], [24]. A deformation is then made by altering the coefficients multiplying the basis functions that appear in the definition of the dynamical system. This approach yields a robust execution-time behavior thanks to the autonomous nature of the dynamical system. However, because of the very dynamical-system representation, inequality (such as joint limits, obstacle avoidance) or equality (such as specified final velocity, acceleration, etc.) constraints at specific time instants cannot be taken into account without integrating the whole trajectory up to these time instants, which can be costly.

The above two approaches are similar in that they make use of *exogenous* basis functions: splines in the spline-based approach and Gaussian kernel functions in the dynamical-system-based approach. A first difficulty then consists in choosing the appropriate bases for a particular task. Second, and more importantly, adding artificial functions to a natural movement can produce undesirable behaviors, such as large spline undulations in spline-based approaches [6], [23], lack of smoothness, etc., which call for supplementary and often costly efforts to correct.

A third approach, based on Euclidean transformations, was proposed recently [3], [25], pioneering the use of transformations groups for trajectory deformation. However, this approach requires reintegration of parts of the trajectory, and its versatility is limited by the small sizes of the Euclidean groups of transformations.

C. Proposed New Approach

Motivated by the large body of evidence of affine invariance in human action and perception, we propose here to deform a given trajectory by *applying affine transformations* on parts of it. Doing so presents several unique benefits. First, the deformed trajectory *inherently preserves* the affine-invariant features of the original trajectory (such as smoothness, periodicity, or, more specifically, *affine velocity*) which, in the light of the previous discussion, may be particularly relevant in the effort to make transferred or retargeted motions look more natural, human-like. Correlatively, since the only "basis functions" are the time-series of the original trajectory coordinates, no artifacts (such as large spline undulations in spline-based approaches) can be introduced. Second, this approach enables one to "convert" the constraints and optimization objectives regarding the

deformed trajectory into constraints and optimization objectives regarding the matrix of the deformation in a natural way, making constraints satisfaction and optimization substantially easier and faster in many cases.

In Section II, we present the core algorithm that deforms a trajectory to reach a new target configuration while satisfying smoothness and inequality constraints. In Section III, we show how to leverage the extra redundancy offered by the affine transformations to optimize geometric (such as closeness to the original trajectory or deformation rigidity) or dynamic costs (such as joint torques). In Section IV, we illustrate these results with a concrete application: transfer of human motions to a humanoid robot preserving the equiaffine velocity. In Section V, we give a characterization of *trajectory redundancy* by the group of admissible deformations, revisiting thereby the concept of kinematic redundancy and suggesting a new theoretical approach to motion planning. Finally, in Section VI, we conclude by discussing the advantages and limitations of the proposed approach, as well as directions for further developments.

II. AFFINE TRAJECTORY DEFORMATION: CORE FRAMEWORK

A. Affine Spaces, Affine Transformations, and Affine Deformations

An affine space is a set $\mathbb{A}$ together with a group action of a vector space $\mathbb{W}$. An element $\mathbf{w} \in \mathbb{W}$ transforms a point $\mathbf{q} \in \mathbb{A}$ into another point $\mathbf{q}'$ by

$$\mathbf{q}' = \mathbf{q} + \mathbf{w}$$

which can also be noted $\mathbf{q}' - \mathbf{q} = \mathbf{w}$ or $\overrightarrow{\mathbf{q}\mathbf{q}'} = \mathbf{w}$.

Given a point $\mathbf{q}_0 \in \mathbb{A}$ (the origin), an affine transformation $\mathcal{F}$ of the affine space can be defined by a couple $(\mathbf{w}, \mathcal{M})$, where $\mathbf{w} \in \mathbb{W}$ and $\mathcal{M}$ is a nonsingular linear map $\mathbb{W} \to \mathbb{W}$. The transformation $\mathcal{F}$ acts on $\mathbb{A}$ by

$$\forall \mathbf{q} \in \mathbb{A} \quad \mathcal{F}(\mathbf{q}) = \mathbf{q}_0 + \mathcal{M}(\overrightarrow{\mathbf{q}_0 \mathbf{q}}) + \mathbf{w}.$$

Note that, if $\mathbf{q}_0$ is a *fixed-point* of $\mathcal{F}$, then $\mathcal{F}$ can be written in the form

$$\forall \mathbf{q} \in \mathbb{A} \quad \mathcal{F}(\mathbf{q}) = \mathbf{q}_0 + \mathcal{M}(\overrightarrow{\mathbf{q}_0 \mathbf{q}}). \tag{1}$$

If $\mathbb{A}$ and $\mathbb{W}$ are in fact $\mathbb{R}^n$, then the set of affine transformations $\mathcal{F}$ form a Lie group of dimension $n^2 + n$, called the General Affine group and denoted GA(n).

We shall also consider two subgroups of GA(n):
1) the *special equiaffine group*, of dimension $n^2 + n - 1$ and denoted SEA(n), which consists of affine transformations whose $\mathcal{M}$ have determinant 1;
2) the *special Euclidean group*, of dimension $n(n + 1)/2$ and denoted SE(n), which consists of affine transformations whose $\mathcal{M}$ are orthogonal and have determinant 1.

Consider now a given trajectory $\mathbf{q}(t)_{t \in [0,T]}$, which may represent, e.g., the Cartesian coordinates of a manipulator's end-effector, the joint angles of a humanoid robot, or the position of a mobile robot in the plane. We say that a transformation $\mathcal{F}$ *deforms* $\mathbf{q}(t)_{t \in [0,T]}$ into $\mathbf{q}'(t)_{t \in [0,T]}$ at time instant τ if

$$\forall t < \tau \quad \mathbf{q}'(t) = \mathbf{q}(t)$$
$$\forall t \geq \tau \quad \mathbf{q}'(t) = \mathcal{F}(\mathbf{q}(t)).$$

If $\mathcal{F}$ is affine (respectively, equiaffine, Euclidean—we drop the term "special" for convenience), we say that the deformation is affine (respectively, equiaffine, Euclidean).

The idea is to manipulate the time instant τ and the transformation $\mathcal{F}$ to achieve the desired trajectory corrections while respecting smoothness constraints. This is discussed in the next sections.

B. Equality Constraints

1) Smoothness Constraints at the Deformation Time Instant: Assume that the trajectory $\mathbf{q}(t)_{t \in [0,T]}$ is of dimension n and is C^p, that is, differentiable p times with a continuous pth derivative. Consider an affine transformation $\mathcal{F}$ that deforms $\mathbf{q}(t)_{t \in [0,T]}$ into $\mathbf{q}'(t)_{t \in [0,T]}$ at a time instant τ. We say that $\mathcal{F}$ is C^p-*preserving* if the resulting $\mathbf{q}'(t)_{t \in [0,T]}$ is also C^p.

Since $\mathcal{F}$ is a smooth application, it is clear that $\mathbf{q}'(t)_{t \in (\tau,T]}$—note that the interval is open at τ—is also C^p. Regarding the time instant τ, the continuity (C^0) of $\mathbf{q}'(t)_{t \in [0,T]}$ imposes that $\mathcal{F}(\mathbf{q}(\tau)) = \mathbf{q}(\tau)$. Thus, $\mathcal{F}$ can be written in the form of (1) with $\mathbf{q}(\tau)$ replacing $\mathbf{q}_0$. Next, remark that the velocity of the deformed trajectory at τ is given by

$$\dot{\mathbf{q}}'(\tau) = \left. \frac{d\mathbf{q}'}{dt} \right|_{t=\tau} = \left. \frac{d\mathbf{M}(\mathbf{q} - \mathbf{q}(\tau))}{dt} \right|_{t=\tau} = \mathbf{M}\dot{\mathbf{q}}(\tau) \quad (2)$$

where $\mathbf{M}$ denotes the matrix of $\mathcal{M}$ in the canonical basis. Extending to higher order derivatives, the requirement that the deformed trajectory be C^p-continuous at τ is then given by

$$\mathbf{M}\dot{\mathbf{q}}(\tau) = \dot{\mathbf{q}}(\tau), \ \mathbf{M}\ddot{\mathbf{q}}(\tau) = \ddot{\mathbf{q}}(\tau), \ \dots \ \mathbf{M}\mathbf{q}^{(p)}(\tau) = \mathbf{q}^{(p)}(\tau). \quad (3)$$

Denote now by $\mathbf{m}$ the vector of dimension n^2 in which one has stacked the n^2 coefficients of $\mathbf{M}$ row by row, that is

$$\mathbf{m}_1 \stackrel{\text{def}}{=} \mathbf{M}_{11}, \ \mathbf{m}_2 \stackrel{\text{def}}{=} \mathbf{M}_{12}, \ \dots, \ \mathbf{m}_{n^2} \stackrel{\text{def}}{=} \mathbf{M}_{nn}.$$

Denote by $\mathbf{S}[\dot{\mathbf{q}}(\tau), \ddot{\mathbf{q}}(\tau), \dots, \mathbf{q}^{(p)}(\tau)]$ and $\mathbf{s}[\dot{\mathbf{q}}(\tau), \ddot{\mathbf{q}}(\tau), \dots, \mathbf{q}^{(p)}(\tau)]$, respectively, the matrix of dimension $np \times n^2$ and the vector of dimension np in which one has stacked the $\dot{\mathbf{q}}(\tau)$, $\ddot{\mathbf{q}}(\tau), \dots, \mathbf{q}^{(p)}(\tau)$ as follows:

$$\mathbf{S} \stackrel{\text{def}}{=} \begin{bmatrix} \dot{\mathbf{q}}(\tau)^\top & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \ddots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \dot{\mathbf{q}}(\tau)^\top \\ & \vdots & \\ \mathbf{q}^{(p)}(\tau)^\top & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \ddots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{q}^{(p)}(\tau)^\top \end{bmatrix}, \quad \mathbf{s} \stackrel{\text{def}}{=} \begin{bmatrix} \dot{\mathbf{q}}(\tau) \\ \vdots \\ \mathbf{q}^{(p)}(\tau) \end{bmatrix}.$$

The system of (3) is then equivalent to the following matrix equation:

$$\mathbf{S}[\dot{\mathbf{q}}(\tau), \ddot{\mathbf{q}}(\tau), \dots, \mathbf{q}^{(p)}(\tau)]\mathbf{m} = \mathbf{s}[\dot{\mathbf{q}}(\tau), \ddot{\mathbf{q}}(\tau), \dots, \mathbf{q}^{(p)}(\tau)].$$

a) Set of deformations satisfying C^p-continuity: If $\dot{\mathbf{q}}(\tau)$, $\ddot{\mathbf{q}}(\tau), \dots, \mathbf{q}^{(p)}(\tau)$ are linearly independent, then the set of

matrices $\mathbf{M}$ that respect C^p-continuity has dimension $n^2 - np = n(n - p)$. Furthermore, this set has an interesting group structure: If $\mathbf{M}_1$ and $\mathbf{M}_2$ both satisfy (2), then so do $\mathbf{M}_1^{-1}$ and $\mathbf{M}_1\mathbf{M}_2$. This shows that the space of affine deformations respecting C^p-continuity is a Lie subgroup of dimension $n(n - p)$ of GA(n).

b) Nonintegral degrees of smoothness: In some cases, the required degree of smoothness is *noninteger*. For instance, under reasonable assumptions on the control inputs, a nonhalting trajectory of a car-like robot in the plane is shown to be C^1 *and curvature-continuous* [26], [27], a requirement which can be considered as being strictly between C^1 and C^2. In this case, one can show that the space of admissible affine deformations at τ is a group of dimension 1 [26], which is strictly between $2 \times (2 - 1) = 2$ and $2 \times (2 - 2) = 0$.

2) Accuracy Constraints at the Final Time Instant: To reach a desired position $\mathbf{q}_d$ at the final time instant T, one needs to satisfy

$$\mathbf{q}'(T) = \mathcal{F}(\mathbf{q}(T)) = \mathbf{q}(\tau) + \mathbf{M}(\mathbf{q}(T) - \mathbf{q}(\tau)) = \mathbf{q}_d$$

which yields

$$\mathbf{M}(\mathbf{q}(T) - \mathbf{q}(\tau)) = \mathbf{q}_d - \mathbf{q}(\tau). \quad (4)$$

Next, following (2), to reach the desired velocity $\dot{\mathbf{q}}_d$, acceleration $\ddot{\mathbf{q}}_d$ or kth derivative $\mathbf{q}_d^{(k)}$, one needs to satisfy furthermore

$$\mathbf{M}\dot{\mathbf{q}}(T) = \dot{\mathbf{q}}_d, \ \mathbf{M}\ddot{\mathbf{q}}(T) = \ddot{\mathbf{q}}_d, \ \dots, \ \mathbf{M}\mathbf{q}^{(k)}(T) = \mathbf{q}_d^{(k)}. \quad (5)$$

Following the notations introduced previously, the system of (4) and (5) is then equivalent to the following matrix equation:

$$\mathbf{S}[\mathbf{q}(T) - \mathbf{q}(\tau), \dot{\mathbf{q}}(T), \ddot{\mathbf{q}}(T), \dots, \mathbf{q}^{(k)}(T)]\mathbf{m} =$$
$$\mathbf{s}[\mathbf{q}_d - \mathbf{q}(\tau), \dot{\mathbf{q}}_d, \ddot{\mathbf{q}}_d, \dots, \mathbf{q}_d^{(k)}].$$

Thus, satisfying C^p-continuity at the deformation time instant *and* achieving kth-order accuracy at the final time instant ($k = 0$ if only the final position is constrained, $k = 1$ if the final position and velocity are constrained, etc.) can be rendered by the condition that the coefficients of $\mathbf{M}$ satisfy

$$\mathbf{S}[\dot{\mathbf{q}}(\tau), \ \dots \ , \mathbf{q}^{(p)}(\tau), \mathbf{q}(T) - \mathbf{q}(\tau), \dot{\mathbf{q}}(T), \dots, \mathbf{q}^{(k)}(T)]\mathbf{m}$$
$$= \mathbf{s}[\dot{\mathbf{q}}(\tau), \dots, \mathbf{q}^{(p)}(\tau), \mathbf{q}_d - \mathbf{q}(\tau), \dot{\mathbf{q}}_d, \dots, \mathbf{q}_d^{(k)}]. \quad (6)$$

a) Set of admissible deformation: For convenience, a deformation that satisfies (6) is said *admissible*. As in Section II-B1, if $n \geq p + k + 1$ and that $\dot{\mathbf{q}}(\tau)$, $\ddot{\mathbf{q}}(\tau)$, $\dots, \mathbf{q}^{(p)}(\tau)$, $\mathbf{q}(T) - \mathbf{q}(\tau)$, $\dot{\mathbf{q}}(T), \dots, \mathbf{q}^{(k)}(T)$ are linearly independent, then the space of admissible deformations has dimension $n(n - p - k - 1)$. Note, however, that this set does not have a group structure, since composing two admissible deformations will bring $\mathbf{q}(T)$ again away from $\mathbf{q}_d$. The group structure would, however, apply if one assumes that $\mathbf{q}_d = \mathbf{q}(T)$, $\dot{\mathbf{q}}_d = \dot{\mathbf{q}}(T)$, etc.; cf., Section V.

Finally, enforcing constraints of order k at K different time instants would further reduce the dimension of the space of admissible deformations to $n(n - p - K(k + 1))$.

3) Degenerate Cases: Let us now discuss the degenerate cases. For clarity, we use the following notations:

$$V(\tau) \stackrel{\text{def}}{=} \{\ddot{\mathbf{q}}(\tau), \ldots, \mathbf{q}^{(p)}(\tau)\}$$

$$V(T) \stackrel{\text{def}}{=} \{\mathbf{q}(T) - \mathbf{q}(\tau), \dot{\mathbf{q}}(T), \ldots, \mathbf{q}^{(k)}(T)\}.$$

For simplicity, we restrict the discussion to the following particular cases.

1) One of the vector of $V(T)$ can be written as a linear combination of the other vectors of $V(T)$. For simplicity, assume e.g. that $\mathbf{q}^{(i)}(T) = \lambda \mathbf{q}^{(j)}(T)$, with $i \neq j$. In this case, if $\mathbf{q}_d^{(i)} \neq \lambda \mathbf{q}_d^{(j)}$, then (6) has no solution, independently of the choice of τ.

2) One of the vector of $V(T)$ can be written as a linear combination of the vectors of $V(\tau)$. For simplicity, assume, e.g., that $\mathbf{q}(T) - \mathbf{q}(\tau) = \lambda \dot{\mathbf{q}}(\tau)$. In this case, if $\mathbf{q}_d - \mathbf{q}(\tau) \neq \lambda \dot{\mathbf{q}}(\tau)$, then (6) has no solution. It is, thus, important to choose the deformation instant τ so that the vectors of $V(\tau)$ are independent from the vectors of $V(T)$. In this sense, straight line trajectories are "bad" for affine deformations since for any value of τ, $\mathbf{q}(T) - \mathbf{q}(\tau)$ is always collinear with $\dot{\mathbf{q}}(\tau)$. On the other hand, trajectories that undulate are "good" in the sense that the set $V(\tau)$ covers large sets of values as τ changes.

3) One of the vectors of $V(\tau)$ can be written as a linear combination of the other vectors of $V(\tau)$. This does not change the solvability of (6) since the left-hand side and the right-hand side of (2) are coordinated.

4) The robot stops at the end of the original trajectory, i.e., $\dot{\mathbf{q}}(T) = 0$, $\ddot{\mathbf{q}}(T) = 0$, etc. In most cases, the robot would be required to also stop at the end of the deformed trajectory, i.e., $\dot{\mathbf{q}}_d = 0$, $\ddot{\mathbf{q}}_d = 0$, etc. In such cases, (5) are always satisfied; therefore, there is no need to include them in (6), and one can set $k = 0$.

From the above development, one criterion to choose τ would be generally to avoid singular (i.e., non-full-rank) or near singular resulting matrices $\mathbf{S}$.

4) Subgroup Constraints: Restriction to a subgroups of the full affine group, e.g., to the equiaffine group or to the Euclidean group (cf., Section II-A), can also be treated as equality constraints. However, these constraints involve the coefficients of $\mathbf{M}$ in a *nonlinear* way. For instance, constraining the transformations to the equiaffine group amounts to the condition that $\det(\mathbf{M}) = 1$, while constraining them to the Euclidean group amounts to the condition that all the singular values of $\mathbf{M}$ be equal to 1. These constraints are more difficult to enforce than (6), requiring—in general—gradient-based methods. Concrete examples are given in Sections III-B and IV.

C. Inequality Constraints

In addition to equality constraints, many applications also require the satisfaction of *inequality* constraints, such as joint limits, upper-bounds on the velocities, accelerations or torques, avoidance of obstacles, etc. In many cases, these constraints can be expressed by

$$\forall i \in [1, K_{\text{ineq}}] \quad \mathbf{A}_i \mathbf{q}'(t_i) \leq \mathbf{b}_i \tag{7}$$

where $t_i \in [\tau, T]$ is a specific time instant, $\mathbf{A}_i$ is a $c \times n$ matrix, and $\mathbf{b}_i$ is a c-dimensional vector. To enforce joints limits, one can, for example, choose several t_i that sample the region where the joint values are expected to be large. Note that constraints

on higher-order derivatives such as $\dot{\mathbf{q}}$, $\ddot{\mathbf{q}}$, etc., can be similarly accommodated.

Next, observe that

$$\begin{aligned} \mathbf{A}_i \mathbf{q}'(t_i) &= \mathbf{A}_i (\mathbf{q}(\tau) + \mathbf{M}(\mathbf{q}(t_i) - \mathbf{q}(\tau))) \\ &= \mathbf{A}_i \mathbf{q}(\tau) + \mathbf{A}_i \mathbf{S}[\mathbf{q}(t_i) - \mathbf{q}(\tau)]\mathbf{m}. \end{aligned}$$

Thus, inequality (7) becomes

$$\mathbf{A}_i \mathbf{S}[\mathbf{q}(t_i) - \mathbf{q}(\tau)]\mathbf{m} \leq \mathbf{b}_i - \mathbf{A}_i \mathbf{q}(\tau).$$

Now, stacking vertically the $\mathbf{A}_i \mathbf{S}[\mathbf{q}(t_i) - \mathbf{q}(\tau)]$ into a matrix $\mathbf{A}$ and the $\mathbf{b}_i - \mathbf{A}_i \mathbf{q}(\tau)$ into a vector $\mathbf{b}$, the inequality constraints (7) amount to

$$\mathbf{Am} \leq \mathbf{b}. \tag{8}$$

Thus, satisfying smoothness constraints at the deformation time instant, accuracy constraints at the final time instant, and inequality constraints at intermediate time instants amount to an equality (6) and an inequality (8) in $\mathbf{m}$.

Example: As illustration, consider a planar three-link manipulator. The original trajectory of the end-effector is a straight line between the initial position and the final position. However, the corresponding joint-angle trajectory violates several joint limit constraints. A deformed C^1 trajectory is then computed that connects the initial and final positions ($n = 3$, $p = 1$, $k = 0$) while avoiding the joint limits (see Fig. 1). Note that, as the joint limits become more stringent, the deformed trajectory must depart farther from the original one (see, e.g., the difference between Fig. 1(a) and (b), where the upper limits on $q_2 + q_3$ was lowered from 2.9 to 2.8 rad), eventually leading to infeasible solutions in practice. While this issue is common to every deformation method, it can be more severe for affine deformations, since the deformation space has relatively low dimension $[n(n - p - k - 1)]$.

Bounds on velocities and accelerations: In applications related to human-to-robot motion transfer, fast human motions usually violate the bounds on the joint velocities and accelerations of existing humanoid robots. In these case, before performing deformations, one may uniformly time-scale the whole trajectory by a constant factor: Such an operation would not affect affine velocities/accelerations and would multiply equiaffine and Euclidean velocities/accelerations by a constant factor.

III. OPTIMIZATION

From Section II-B2, if $n > p + k + 1$, then multiple deformations may satisfy the desired smoothness and accuracy constraints. This section details how one can choose the deformations that furthermore *optimize* some criteria.

A. Closeness Maximization

One important optimization objective for the deformed trajectory is to be the "closest" possible to the original one, which can be equated to minimizing the distance

$$d(\mathbf{q}', \mathbf{q}) \stackrel{\text{def}}{=} \sup_{t \in [\tau, T]} \|\mathbf{q}'(t) - \mathbf{q}(t)\|$$
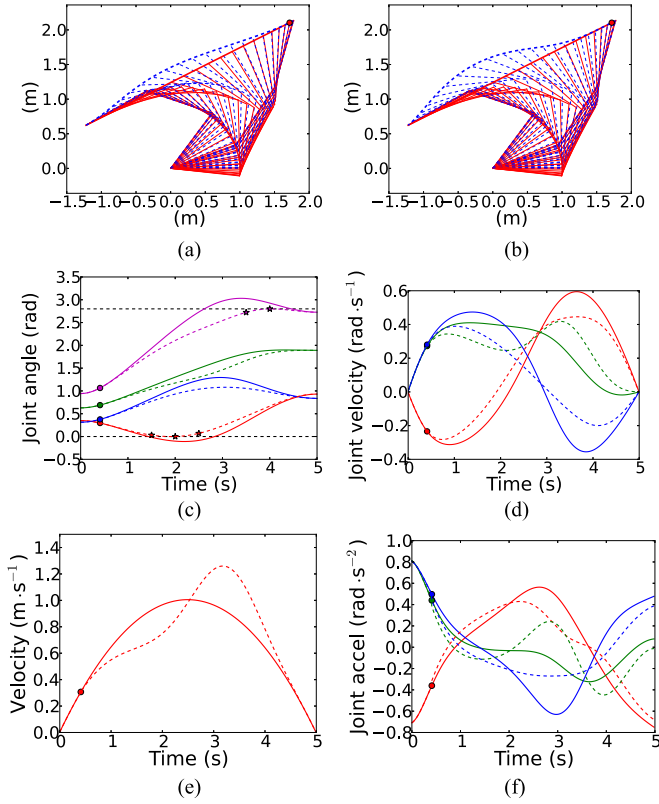
Fig. 1. Trajectory deformation for a planar three-link manipulator under inequality constraints. The deformation operates in joint space. It respects C^1-continuity while keeping the final configuration unchanged ($n = 3$, $p = 1$, $k = 0$). We want to enforce the inequalities $q_1 \geq 0$ and $q_2 + q_3 \leq \alpha$, with $\alpha = 2.9$ rad in (a) and $\alpha = 2.8$ rad in (b). The deformation was made to enforce these constraints at specific time instants while maintaining the deformed trajectory the closest possible to the original one (cf., Section III-A). (a), (b) Thin lines: manipulator configurations in Cartesian space at regular time intervals (plain red: original, dashed blue: deformed); bold lines: end-point trajectories. The red disk marks the position of the deformation. For (c)–(f), $\alpha = 2.8$. (c) Original (plain lines) and deformed (dashed lines) joint angles against time. The joint values for joints 1, 2, and 3 are, respectively, in red, green, and blue. The red disk marks the time instants of the deformation. The stars mark the time instants when the inequality constraints were enforced. (d) Joint velocities, same legend as in (c). Note the continuity of the deformed velocity profiles (C^1-continuity). (e) Velocity of the end-point in Cartesian space (plain: original, dashed: deformed). (f) Joint accelerations, same legend as in (c). Note that the deformed acceleration profiles were not continuous.

where $\| \cdot \|$ is the usual $\mathcal{L}_2$ norm of vectors. In absence of any structure, the only solution would consist in *approximating* $d(\mathbf{q}', \mathbf{q})$, by, e.g., evaluating $\|\mathbf{q}'(t) - \mathbf{q}(t)\|$ at sample points along the trajectory (which would be moreover time-consuming).

Taking advantage of the affine deformation framework, one may observe that

$$\|\mathbf{q}'(t) - \mathbf{q}(t)\| = \|\mathbf{q}(\tau) + \mathbf{M}(\mathbf{q}(t) - \mathbf{q}(\tau)) - \mathbf{q}(t)\|$$

$$= \|(\mathbf{M} - \mathbf{I})(\mathbf{q}(t) - \mathbf{q}(\tau))\| \leq \|\mathbf{M} - \mathbf{I}\| \cdot \|\mathbf{q}(t) - \mathbf{q}(\tau)\|.$$

Thus, one possible way to obtain small values for $d(\mathbf{q}', \mathbf{q})$ consists in minimizing the *induced $\mathcal{L}_2$ norm* $\|\mathbf{M} - \mathbf{I}\|$. However, minimizing $\|\mathbf{M} - \mathbf{I}\|$ is still computationally expensive since it requires iteratively evaluating the largest singular value of $\mathbf{M} - \mathbf{I}$. Another, more tractable method consists in minimizing

the *Frobenius norm*

$$\|\mathbf{M} - \mathbf{I}\|_F \stackrel{\text{def}}{=} \sqrt{\sum_{i,j}(\mathbf{M}_{ij} - \mathbf{I}_{ij})^2} = \|\mathbf{m} - \mathbf{i}\|$$

where $\mathbf{i}$ is the vector of size n^2 obtained by stacking the n^2 coefficients of $\mathbf{I}$. The above expression yields a practical way to obtain small values for $d(\mathbf{q}', \mathbf{q})$, which consists in solving the following minimization problem:

$$\min_{\mathbf{m}} \|\mathbf{m} - \mathbf{i}\|^2 \quad \text{subject to } \mathbf{Sm} = \mathbf{s}, \ \mathbf{Am} \leq \mathbf{b}.$$

Consider next the change of variable $\tilde{\mathbf{m}} = \mathbf{m} - \mathbf{i}$. The above minimization problem becomes

$$\min_{\mathbf{m}} \|\tilde{\mathbf{m}}\|^2 \quad \text{subject to } \mathbf{S}\tilde{\mathbf{m}} = \mathbf{s} - \mathbf{Si}, \ \mathbf{A}\tilde{\mathbf{m}} \leq \mathbf{b} - \mathbf{Ai} \quad (9)$$

which is a classical quadratic program (QP), which can be fully prioritized and efficiently solved using existing software [29], [30].

Case without inequality constraints: In this case, the QP (9) has the following closed-form solution $\tilde{\mathbf{m}}^* = \mathbf{S}^+(\mathbf{s} - \mathbf{Si})$, where $\mathbf{S}^+$ denotes the Moore–Penrose pseudoinverse of $\mathbf{S}$. The optimal value for $\mathbf{m}$ is then given by

$$\mathbf{m}^* = \mathbf{i} + \mathbf{S}^+(\mathbf{s} - \mathbf{Si}).$$

B. Rigidity Maximization

In some applications, it is important to maximize the *rigidity* of the transformation. For instance, the (Euclidean[2]) curvature profile of a wheeled robot trajectory in 2-D or of an underwater vehicle in 3-D should be preserved as much as possible since this curvature is related to the feasibility of the trajectory [31]. In computer graphics, making the deformation *as rigid as possible* preserves the global look of the trajectory or of the image [32].

In our framework, rigidity optimization is naturally achieved by requiring the affine transformation to be as close as possible to an Euclidean transformation. This, in turn, can be achieved by requiring the singular values of $\mathbf{M}$ to be as close as possible to 1. We have thus the following optimization problem:

$$\min_{\mathbf{m}} \sum_{i=1}^{n}(\sigma_i - 1)^2, \qquad (\sigma_1 \ldots \sigma_n) = \text{svd}(\mathbf{M}) \quad (10)$$

subject to equality and inequality constraints (6) and (8). This optimization problem is nonconvex and cannot, therefore, be solved as efficiently as in the previous section: One must use, for instance, a generic gradient-based method. However, it still presents a substantial improvement as compared with, e.g., evaluating the Euclidean curvature at sample points along the deformed trajectory—which other deformation approaches (spline-based and dynamical-system-based) must resort to.

Example: As illustration, consider the task of deforming a trajectory in 3-D Cartesian space, without inequality constraints. The deformed trajectory must respect C^1-continuity and reach a

---

[2]We insist on the term "Euclidean" here because there exists other types of curvatures, such as the affine or equiaffine ones, which are invariant under affine and equiaffine transformations, respectively; see [13].
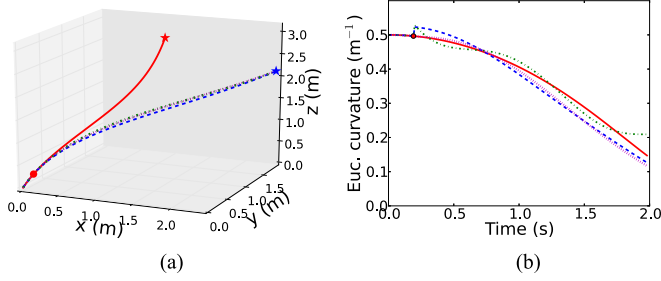
Fig. 2. Maximum rigidity deformation. (a) Original and deformed trajectories in 3-D Cartesian space. The desired final position $\mathbf{q}_d$ (blue star) was shifted by $[+1, +1, -1]$ from the original final position (red star). The original trajectory (solid red) was deformed into the dashed blue trajectory using a maximum-rigidity deformation, into the dashed-dotted magenta one using a polynomial deformation, and into the dotted magenta using a maximum-rigidity deformation with curvature continuity. The red disk marks the time instant of the deformations. The three deformed trajectories were C^1 and reached the new final position ($n = 3$, $p = 1$, $k = 0$). (b) Euclidean curvatures of the original (solid red), maximum-rigidity (dashed blue), polynomial-deformed (dashed-dotted green), and maximum-rigidity with curvature continuity (dotted magenta) trajectories against time. Note that the magenta curvature profile was indeed continuous.

new final position $\mathbf{q}_d$ ($n = 3$, $p = 1$, $k = 0$), while minimizing the change in Euclidean curvature.

We first consider the maximum-rigidity affine deformation just discussed. In case $\mathbf{S}$ has full rank, the "redundant" space has dimension $n(n - p - k - 1) = 3$ and is spanned by the first three eigenvectors of $\mathbf{I} - \mathbf{S}^+\mathbf{S}$. Denote these eigenvectors by $\mathbf{u}_1$, $\mathbf{u}_2$, $\mathbf{u}_3$. Then, the vectors $\mathbf{m}$ satisfying (6) have the form

$$\mathbf{m} = \mathbf{S}^+\mathbf{s} + \lambda_1\mathbf{u}_1 + \lambda_2\mathbf{u}_2 + \lambda_3\mathbf{u}_3, \quad \lambda_1, \lambda_2, \lambda_3 \in \mathbb{R}.$$

For the example depicted in Fig. 2, using a generic gradient-based algorithm, we found the values of λ_1, λ_2, λ_3 that optimize (10) in 0.013 s (Python, GNU/Linux, Intel Core i5 3.2 GHz, 3.8-GB memory). The mean absolute difference between the curvatures of the original and deformed trajectories was 0.023 m^{-1}.

To compare with traditional methods, we next consider a polynomial deformation, which consists in adding a third-degree polynomial to each coordinate of the original trajectory. A third-degree polynomial has four coefficients and there are three constraints (position and velocity at the deformation instant, desired final position). Therefore, one variable is available for optimization. Over the three coordinates, one can thus optimize over a space of dimension 3, which is the same dimension as in the maximum-rigidity affine deformation just discussed. However, there is no clear relationship here between the variables to optimize and the change in Euclidean curvature. One thus have to evaluate the Euclidean curvature at sample points along the deformed trajectory. Experimental results are given in Table I for various numbers of samples N. One can note that the polynomial deformation yielded slightly better results in terms of curvature difference as compared with the maximum-rigidity deformation, but at the expense of significantly increased computation times. More sophisticated spline-based or dynamical-system-based methods would essentially yield similar computation times as the polynomial deformation, since they would also require sampling the Euclidean curvature along the trajectory.

TABLE I
POLYNOMIAL DEFORMATIONS VERSUS MAXIMUM-RIGIDITY AFFINE DEFORMATIONS

Deformation	Mean curv. diff. (m^{-1})	Comp. time (s)
Poly, $N = 180$	0.012	2.26
Poly, $N = 90$	0.012	1.98
Poly, $N = 36$	0.012	2.13
Poly, $N = 18$	3.24	6.79
Poly, $N = 9$	2.03	7.04
Max rigid	0.023	0.013
Max rigid cont curv	0.018	0.028

Finally, we consider maximum-rigidity affine deformation with curvature continuity at the deformation time instant. The latter constraint decreases the dimension of deformation space by 1, but contrary to the 2-D case [26], [27], this constraint is nonlinear in 3-D. To satisfy this constraint, we added to the cost (10) an extra term that penalizes curvature discontinuity, with a large weight. The resulting trajectory was indeed curvature-continuous (see Fig. 2). Computation time was slightly increased, but the curvature difference between the deformed trajectory and the original one was smaller than in the simple maximum-rigidity affine deformation case (see Table I), probably because the extra constraint pulled the gradient descent out of a local minimum.

IV. APPLICATION TO MOTION TRANSFER

As mentioned in Section I, equiaffine velocity is an important invariant in human hand movements both in the plane and in space. The equiaffine velocity of a 3-D trajectory is given by [17]

$$v_{ea}(t) = \left| \frac{d\mathbf{r}}{dt}, \frac{d^2\mathbf{r}}{dt^2}, \frac{d^3\mathbf{r}}{dt^3} \right|^{1/6} \tag{11}$$

where $\mathbf{r}(t)$ is the 3-D coordinate of the hand at time t, and $|u, v, w|$ denotes the scalar triple product of u, v, and w in $\mathbb{R}^3$. It is clear that the so-defined equiaffine velocity is invariant under any equiaffine transformation applied to any part of the trajectory $\mathbf{r}(t)_{t \in [0, T]}$.

To illustrate the concept of motion transfer preserving equiaffine velocity, we first recorded, using the Motion Analysis optical motion capture system, the motion of a human reaching a ball while avoiding obstacles constituted by a plate and a bar (see Fig. 3, top row). We reconstructed the 3-D trajectory of the hand using the wrist markers (see Fig. 3, green trajectory in bottom row). Using inverse kinematics, we found the joint-angle trajectories (shoulder pitch, roll, yaw, elbow flexion) for the HRP4 robot to track this hand trajectories. However, because of the different body structures, the so-obtained robot trajectory would collide with the bar.

Thus, we deformed the 3-D wrist trajectory ($n = 3$) under the following constraints: 1) C^1-smoothness ($p = 1$); 2) final position unchanged ($k = 0$); 3) the transformation is equiaffine ($\det(\mathbf{M}) = 1$); and 4) at a given intermediate time, the Z-coordinate is lower by 7 cm with respect to the original trajectory. As shown in Section II-B, constraints 1 and 2 reduce
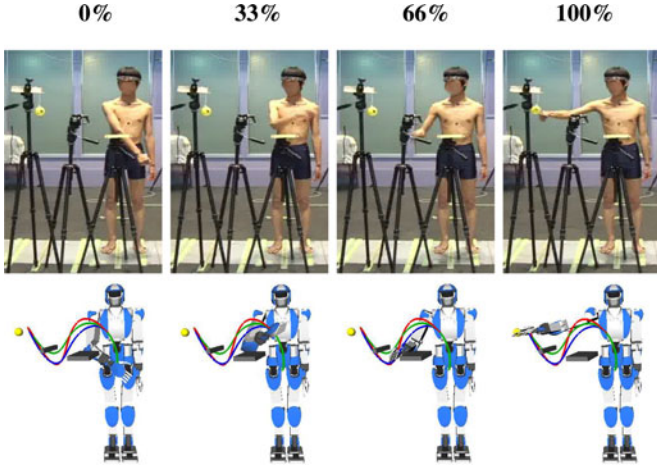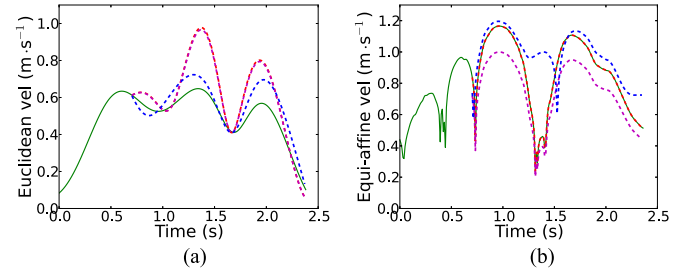
Fig. 4. (a) Euclidean velocity of the hand in space. Green: original trajectory, red: maximum-closeness equiaffine deformation, magenta: maximum-closeness affine deformation, blue: polynomial deformation. (b) Equiaffine velocity of the hand in space. Same legend as in (a). Note that the green and red profiles are identical since the equiaffine velocity is conserved by the deformation. By contrast, the magenta and blue profiles are clearly different from the original profile.

Fig. 3. (Top row) Snapshots of a human spatial reaching movement avoiding obstacles, taken at the beginning (0%), 33%, 66%, and the end of the movement (100%). (Bottom row) Reconstruction of the experimental environment (ball, bar, plate) in OpenRAVE. The original 3-D trajectory of the human hand is in green. Because of the different kinematic structures, the robot motion tracking the green trajectory collides with the bar (not shown). Thus, we deformed the green trajectory into the red trajectory—using an equiaffine transformation—in order to avoid the bar from below. Inverse kinematics (see, e.g., [33], which enables taking into account joint-space constraints) can then be used to find the appropriate joint angles corresponding to the red trajectory. For comparison, a polynomial deformation was used to obtain the blue trajectory.
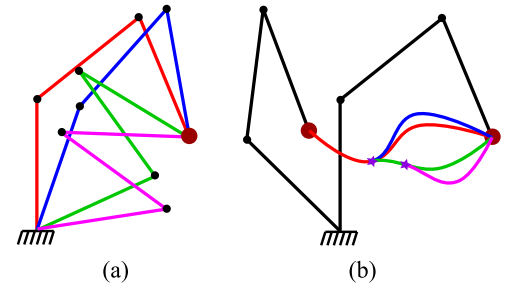

Fig. 5. (a) Configuration redundancy and (b) trajectory redundancy. For simplicity, we have sketched in plot (b) the end-effector trajectory redundancy, but this notion applies more generally to the joint-angle trajectories.

the dimension of the space of admissible deformations to $n(n - p - k - 1) = 3$. Next, constraints 3 and 4 reduce this dimension by 1 each. We then optimize trajectory closeness (cf., Section III-A) over the space of dimension 1 of possible deformations. Note that, since constraint 3 is nonlinear, we had to resort to a gradient method to enforce it as well as to optimize closeness.

Finally, using inverse kinematics, we calculated again the joint angles that enable tracking the deformed trajectory. By this scheme, we have thus obtained a robot motion that has the *exact same* equiaffine velocity profile as in the original human motion [cf., green and red lines in Fig. 4(b)], but avoiding the work space obstacle constituted by the bar.

To compare, we computed trajectories obtained by 1) a maximum-closeness affine deformation (i.e., relaxing the equiaffine constraint) and 2) a polynomial deformation (i.e., adding a third-degree polynomial to the trajectory). The equiaffine velocities of these trajectories were clearly different from that of the original trajectory (see Fig. 4(b): compare green versus red lines on one hand and green versus magenta and blue lines on the other).

Let us note here that the exact preservation of equiaffine velocity comes at a cost: The deformation space is only of dimension 1. Thus, different positions of the obstacles or of the target might lead to a deformed trajectory that departs too far from the original one, and therefore not kinematically trackable (see also discussion in Section II-C regarding severe inequality constraints). This issue can be alleviated, to some extent, by relaxing the equiaffine constraint and considering it instead as an optimization criterion or by composing multiple deformations (see Section V).

V. CHARACTERIZATION OF TRAJECTORY REDUNDANCY BY THE GROUP OF ADMISSIBLE DEFORMATIONS

Building on the affine deformation framework, we now discuss the concept of redundancy from the viewpoint of group theory.

A. Background: Configuration and Velocity Redundancies

A manipulator is said to be kinematically redundant with respect to a task when more degrees of freedom than the minimum number required to execute that task are available (see, e.g., [34], [35]). Consider the system

$$\mathbf{r} = \mathbf{f}(\mathbf{q}) \tag{12}$$

where $\mathbf{r}$ is a vector of dimension m representing the configuration of the end-effector, and $\mathbf{q}$ is a vector of dimension n representing the joint angles. If $n > m$, then there generally exists infinitely many $\mathbf{q}$ that correspond to a given $\mathbf{r}$, which constitutes the notion of *configuration redundancy* [see Fig. 5(a)].

Redundancy can also be studied from a differential viewpoint, which we call *velocity redundancy*. Differentiating (12) indeed yields $\dot{\mathbf{r}} = \mathbf{J_f}(\mathbf{q})\dot{\mathbf{q}}$, where $\mathbf{J_f} = \frac{\partial \mathbf{f}}{\partial \mathbf{q}}$ is the Jacobian matrix of $\mathbf{f}$, of dimension $m \times n$. If $n > m$ and $\mathbf{J_f}(\mathbf{q}(t))$ is nonsingular, then a given desired instantaneous velocity $\mathbf{v_r}^d$ of the end-effector can be achieved by infinitely many different instantaneous velocities $\mathbf{v_q}$ of the joint angles (for notational simplicity, we have dropped the time index t). More precisely, let S represent the null-space

of $\mathbf{J_f}(\mathbf{q})$ and $\mathbf{v_q^*} = \mathbf{J_f}(\mathbf{q})^+\mathbf{v_r^d}$. Then, any joint-angle velocity in the affine subspace $\{\mathbf{v_q^*} + S\}$ will achieve the desired end-effector velocity $\mathbf{v_r^d}$ [34], [35].

From a group-theoretic viewpoint, which will be convenient later on, let T_S denote the space of the translations whose vectors belong to S. This set can actually be viewed as a Lie subgroup of dimension $n - m$ of the general affine group GA(n). The space of all joint-angle velocities $\mathbf{v_q}$ corresponding to a single end-effector velocity $\mathbf{v_r^d}$ described above can then be seen as the orbit of $\mathbf{v_q^*}$ under the action of T_S, and the "degree of velocity redundancy" of the system at $\mathbf{q}$ as the dimension of T_S as a Lie subgroup of GA(n).

B. Trajectory Redundancy

The developments of the previous sections have highlighted another type of redundancy, namely *trajectory redundancy*: Once a particular joint configuration $\mathbf{q}_d$ has been chosen from the many possible joint configurations that achieve a given end-effector configuration, there still exists infinitely many joint-angle *trajectories* that can bring the manipulator from the initial configuration $\mathbf{q}_0$ toward $\mathbf{q}_d$ with a specified velocity, acceleration, etc., while respecting the system kinematic and dynamic constraints [see Fig. 5(b)].

Unlike configuration/velocity redundancies, trajectory redundancy is generally of infinite dimension. Finding a convenient way to *parameterize* a subset of admissible trajectories is then of particular interest. We have seen, from the development of Section II, that given a time instant τ, the space of admissible trajectories that can be obtained by affinely deforming an original trajectory $\mathbf{q}(t)_{t \in [0,T]}$ is the orbit of that trajectory under the action of a subgroup of dimension $n(n - p - k - 1)$ of GA(n) (assuming that $\mathbf{q}_d = \mathbf{q}(T)$ and no inequality constraints).

To make the framework more general, one may *compose* multiple deformations at different time instants. Consider a sequence of L different time instants $\{\tau_1, \ldots, \tau_L\}$, with $0 \leq \tau_1 < \cdots < \tau_L < T$, and a sequence of L affine transformations $\hat{\mathcal{F}} = \{\mathcal{F}_1, \ldots, \mathcal{F}_L\}$. The trajectory $\mathbf{q}'$ obtained from $\mathbf{q}$ by applying successively $\mathcal{F}_1$ at τ_1, then $\mathcal{F}_2$ at τ_2, etc., is given by

$$\forall i \in [1, L], \; \forall t \in (\tau_i, \tau_{i+1}], \; \mathbf{q}'(t) = (\mathcal{F}_1 \circ \ldots \circ \mathcal{F}_i)(\mathbf{q}(t)) \tag{13}$$

with the convention $\tau_{L+1} = T$.

Benefits of the group structure of GA(n): First, each of the $\mathcal{F}_1 \circ \cdots \circ \mathcal{F}_i$ is actually *one* affine transformation by the *composition* property of GA(n). Letting $\mathcal{F}_i^\circ = \mathcal{F}_1 \circ \cdots \circ \mathcal{F}_i$ (note that the cost of computing the $\mathcal{F}_i^\circ$ depends only on the L and not on the length of the trajectory), (13) becomes

$$\forall i \in [1, L], \; \forall t \in (\tau_i, \tau_{i+1}], \; \mathbf{q}'(t) = \mathcal{F}_i^\circ(\mathbf{q}(t)).$$

Thus, even though one has composed L deformations, only *one* affine transformation needs to be applied at each time instant, making computations faster. Second, the *inversion* property may be particularly useful for computer graphics applications: Indeed, a particular requirement for *interactive motion editing* systems to be user-friendly is that every editing operation should be quickly reversible [36]. Finally, the matrix representation of the admissible deformations enables searching efficiently (using

random sampling techniques, gradient-based search, etc.) within the space of trajectory redundancy, as illustrated in Section III.

"Degree of trajectory redundancy": Observe that the space of composed deformations defined as above is, in general, of dimension $Ln(n - p - k - 1)$ and has itself a group structure. For example, the trajectory obtained by composing two composed deformations associated with $\{\mathcal{F}_1, \ldots, \mathcal{F}_L\}$ and $\{\mathcal{G}_1, \ldots, \mathcal{G}_L\}$ is given by

$$\forall i \in [1, L], \; \forall t \in (\tau_i, \tau_{i+1}], \; \mathbf{q}'(t) = (\mathcal{F}_i^\circ \circ \mathcal{G}_i^\circ)(\mathbf{q}(t)). \tag{14}$$

We call this group the *affine deformation group* of $\mathbf{q}(t)_{t \in [0,T]}$ at $\tau_1, \ldots, \tau_L$ and denote it by $A_{\tau_1, \ldots, \tau_L}(\mathbf{q})$.

Recall that we have previously identified the redundancy of configurations/velocities with a certain group of translations T_S of dimension $n - m$. Similarly, one can identify here part of the redundancy of trajectories with the group $A_{\tau_1, \ldots, \tau_L}(\mathbf{q})$, in the sense that the orbit of $\mathbf{q}$ under the action of $A_{\tau_1, \ldots, \tau_L}(\mathbf{q})$ are trajectories that respect the problem constraints (smoothness, final constraints, etc.) One can then associate the "degree of trajectory redundancy" with the *dimension* of $A_{\tau_1, \ldots, \tau_L}(\mathbf{q})$, which provides a novel, *quantitative*, view on the "degree of movement freedom" of a robot: The larger the trajectory deformation group, the more "movement freedom" the robot enjoys and the more easily one can plan and deform its trajectories. In this view, continuity-related constraints (C^1, C^2, ...) or group-related constraints (equiaffine, Euclidean, etc.) are unified in that they both reduce the trajectory deformation group of a robot to one of its subgroups. In wheeled robots, for instance, the structure of car-like robots imposes the supplementary constraint of *curvature-continuity*, thus reducing the deformation groups of omnidirectional robots (of dimension 2 in general, assuming a single deformation) to subgroups of dimension 1 [27]. By contrast, relaxing the constant curvature constraints of bevel needle trajectories [28] would *extend* the Euclidean deformation group of dimension 1 (spanned by the rotations around the needle axis) to one of its supergroups.

VI. CONCLUSION

Motivated by the large body of evidence of affine invariance in human action and perception, we have presented a new approach to deform robot trajectories based on affine transformations. The trajectories obtained by this approach preserve *by construction* affine-invariant properties of the original trajectories. Correlatively, in contrast with spline-based or dynamic-system-based approaches, no artifacts (such as large undulations in spline-based approaches [5], [6], wavelets with too much energy [23], undesirable frequencies, etc.) can be introduced into the deformed trajectory. These distinctive features may prove particularly relevant in character animation or human-to-robot motion transfer applications.

In addition, as the proposed approach enables one to naturally "convert" the constraints and optimization objectives regarding the deformed trajectory into constraints and optimization objectives regarding the matrix of the deformation, it makes constraints satisfaction and optimization substantially easier and faster. For instance, we showed that minimizing the distance

between the deformed and the original trajectories could be achieved by solving a simple QP, while maximizing the rigidity could be achieved without having to evaluate Euclidean quantities at sample points along the trajectory.

The previously discussed advantages come, however, at a cost: Because of the linear and global nature of the deformations, the number of equality constraints *one* deformation may accommodate is limited, precisely by $p + k + 1 \leq n$, where p is the required degree of smoothness of the deformed trajectory, k is the number of derivative constraints at the final time instant ($k = 0$ if only the position is constrained, $k = 1$ if the position and the velocity are constrained, etc.), and n is the dimension of the system. This is an important limitation of our framework, since in typical human-centered applications where affine invariance is concerned, the dimension of the system is $n = 3$ (Cartesian space). This issue can be alleviated, to some extent, by composing multiple deformations (see Section V-B), which would correspond to increasing the number of control points in spline-based approaches. Note, however, that, when $n = 3$, requiring C^2-continuity ($p = 2$) reduces the space of deformations to the identity, irrespective of the number of deformations. In the joint space, the dimensions of typical systems are higher (e.g., $n = 6$ for a humanoid robot arm), thus allowing more flexibility. However, in such cases, the physiological motivations become less relevant (we are not aware of any affine invariance in the joint space)—only the computational motivations remain.

Note also that equality constraints on *continuous* time intervals cannot be addressed using a finite number of deformations. In such cases, a solution may consist in combining affine deformations and downstream kinematic and/or dynamic filters, as suggested in [1]. Finally, the number of inequality constraints such as (7) is not critical, what matters is how *severe* these constraints are.

Based on the presented affine deformation framework, we have also suggested a novel group-based characterization of trajectory redundancy which, besides practical interests (computation speed-up using composition of deformations, reversibility in interactive motion editing using inversion of deformations, reduction of the search space using the matrix-based parameterization of trajectory redundancy, etc.), might also contribute to advance the conceptual understanding of robot motion planning. Our current research focuses on developing this framework for full-scale applications in character animation [4], [6], [33], [37] and humanoid robot control [7], [8].

Nature often finds elegant solutions to address complex problems, as captured by the notion of "simplexity" put forward by A. Berthoz [38]. In this picture, can the simple yet versatile framework of affine deformation be one of the causes of the existence of affine invariance in human action and perception?

ACKNOWLEDGMENT

REFERENCES

[1] Q.-C. Pham and Y. Nakamura, "Affine trajectory deformation for redundant manipulators," in *Proc. Robot.: Sci. Syst.*, 2012, pp. 329–336.

[2] F. Lamiraux, D. Bonnafous, and O. Lefebvre, "Reactive path deformation for nonholonomic mobile robots," *IEEE Trans. Robot.*, vol. 20, no. 6, pp. 967–977, Dec. 2004.

[3] K. Seiler, S. Singh, and H. Durrant-Whyte, "Using Lie group symmetries for fast corrective motion planning," in *Proc. Algorithmic Foundations Robot. IX*, 2010, pp. 37–52.

[4] C. Rose, B. Guenter, B. Bodenheimer, and M. Cohen, "Efficient generation of motion transitions using spacetime constraints," in *Proc. 23rd Annu. Conf. Comput. Graphics Interactive Tech.*, 1996, pp. 147–154.

[5] M. Gleicher, "Retargetting motion to new characters," in *Proc. Annu. Conf. Comput. Graphics Interactive Tech.*, 1998, pp. 33–42.

[6] J. Lee and S. Shin, "A hierarchical approach to interactive motion editing for human-like figures," in *Proc. Annu. Conf. Comput. Graphics Interactive Tech.*, 1999, pp. 39–48.

[7] K. Yamane and Y. Nakamura, "Dynamics filter—Concept and implementation of online motion generator for human figures," *IEEE Trans. Robot. Autom.*, vol. 19, no. 3, pp. 421–432, 2003.

[8] A. Ude, A. Gams, T. Asfour, and J. Morimoto, "Task-specific generalization of discrete and periodic dynamic movement primitives," *IEEE Trans. Robot.*, vol. 26, no. 5, pp. 800–815, Oct. 2010.

[9] F. Lacquaniti, C. Terzuolo, and P. Viviani, "The law relating the kinematic and figural aspects of drawing movements," *Acta Psychol. (Amst)*, vol. 54, nos. 1/3, pp. 115–130, Oct. 1983.

[10] P. Viviani and M. Cenzato, "Segmentation and coupling in complex movements," *J. Exp. Psychol. Hum. Percept. Perform.*, vol. 11, no. 6, pp. 828–845, Dec. 1985.

[11] P. Viviani and R. Schneider, "A developmental study of the relationship between geometry and kinematics in drawing movements," *J. Exp. Psychol. Hum. Percept. Perform.*, vol. 17, no. 1, pp. 198–218, Feb. 1991.

[12] H. Hicheur, S. Vieilledent, M. J. E. Richardson, T. Flash, and A. Berthoz, "Velocity and curvature in human locomotion along complex curved paths: a comparison with hand movements," *Exp. Brain Res.*, vol. 162, no. 2, pp. 145–154, Apr. 2005.

[13] D. Bennequin, R. Fuchs, A. Berthoz, and T. Flash, "Movement timing and invariance arise from several geometries," *PLoS Comput. Biol.*, vol. 5, no. 7, p. e1000426, Jul. 2009.

[14] C. deSperati and P. Viviani, "The relationship between curvature and velocity in two-dimensional smooth pursuit eye movements," *J. Neurosci.*, vol. 17, no. 10, pp. 3932–3945, 1997.

[15] F. E. Pollick and G. Sapiro, "Constant affine velocity predicts the 1/3 power law of planar motion perception and generation," *Vision Res.*, vol. 37, no. 3, pp. 347–353, Feb. 1997.

[16] F. Pollick, U. Maoz, A. Handzel, P. Giblin, G. Sapiro, and T. Flash, "Three-dimensional arm movements at constant equi-affine speed," *Cortex*, vol. 45, no. 3, pp. 325–339, 2009.

[17] U. Maoz, A. Berthoz, and T. Flash, "Complex unconstrained three-dimensional hand movement and constant equi-affine speed," *J. Neurophysiol.*, vol. 101, no. 2, pp. 1002–1015, 2009.

[18] U. Maoz and T. Flash, "Spatial constant equi-affine speed and motion perception," *J. Neurophysiol.*, vol. 111, no. 2, pp. 336–349, 2014.

[19] S. Kandel, J.-P. Orliaguet, and P. Viviani, "Perceptual anticipation in handwriting: The role of implicit motor competence," *Perception Psychophys.*, vol. 62, no. 4, pp. 706–716, 2000.

[20] A. B. Schwartz, "Direct cortical representation of drawing," *Science*, vol. 265, no. 5171, pp. 540–542, 1994.

[21] E. Dayan, A. Casile, N. Levit-Binnun, M. A. Giese, T. Hendler, and T. Flash, "Neural representations of kinematic laws of motion: Evidence for action-perception coupling," *Proc. Nat. Acad. Sci.*, vol. 104, no. 51, pp. 20582–20587, 2007.

[22] A. Casile, E. Dayan, V. Caggiano, T. Hendler, T. Flash, and M. A. Giese, "Neuronal encoding of human kinematic invariants during action observation," *Cerebral Cortex*, vol. 20, no. 7, pp. 1647–1655, 2010.

[23] A. Ude, C. Atkeson, and M. Riley, "Planning of joint trajectories for humanoid robots using B-spline wavelets," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2000, vol. 3, pp. 2223–2228.

[24] A. Ijspeert, J. Nakanishi, and S. Schaal, "Movement imitation with nonlinear dynamical systems in humanoid robots," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2002, pp. 1398–1403.

[25] P. Cheng, E. Frazzoli, and S. LaValle, "Improving the performance of sampling-based motion planning with symmetry-based gap reduction," *IEEE Trans. Robot.*, vol. 24, no. 2, pp. 488–494, Apr. 2008.

[26] Q.-C. Pham, "Fast trajectory correction for nonholonomic mobile robots using affine transformations," in *Proc. Robot.: Sci. Syst.*, 2011, pp. 265–272.

[27] Q.-C. Pham and Y. Nakamura, "Regularity properties and deformation of wheeled robots trajectories," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2012, pp. 3212–3217.

[28] V. Duindam, J. Xu, R. Alterovitz, S. Sastry, and K. Goldberg, "Three-dimensional motion planning algorithms for steerable needles using inverse kinematics," *Int. J. Robot. Res.*, vol. 29, no. 7, pp. 789–800, 2010.

[29] O. Kanoun, F. Lamiraux, and P.-B. Wieber, "Kinematic control of redundant manipulators: Generalizing the task-priority framework to inequality tasks," *IEEE Trans. Robot.*, vol. 27, no. 4, pp. 785–792, Aug. 2011.

[30] A. Escande, N. Mansard, and P.-B. Wieber, "Hierarchical quadratic programming: Fast online humanoid-robot motion generation," *Int. J. Robot. Res.*, vol. 33, pp. 1006–1028, 2014.

[31] J.-P. Laumond, *Robot Motion Planning and Control*. New York, NY, USA: Springer-Verlag, 1998.

[32] K. Shoemake and T. Duff, "Matrix animation and polar decomposition," in *Proc. Conf. Graphics Interface*, 1992, vol. 92, pp. 258–264.

[33] K. Yamane and Y. Nakamura, "Natural motion animation through constraining and deconstraining at will," *IEEE Trans. Vis. Comput. Graphics*, vol. 9, no. 3, pp. 352–360, Jul.–Sep. 2003.

[34] Y. Nakamura, *Advanced Robotics: Redundancy and Optimization*. Reading, MA, USA: Addison-Wesley, 1990.

[35] B. Siciliano and J.-J. E. Slotine, "A general framework for managing multiple tasks in highly redundant robotic systems," in *Proc. Int. Conf. Adv. Robot.*, 1991, pp. 1211–1216.

[36] B. Shneiderman, "Direct manipulation for comprehensible, predictable and controllable user interfaces," in *Proc. 2nd Int. Conf. Intell. User Interfaces*, 1997, pp. 33–39.

[37] K. Yamane, J. Kuffner, and J. Hodgins, "Synthesizing animations of human manipulation tasks," *ACM Trans. Graphics*, vol. 23, pp. 532–539, 2004.

[38] A. Berthoz, *La simplexité*. Paris, France: Odile Jacob, 2009.