Biol Cybern (2005)
DOI 10.1007/s00422-005-0008-x

ORIGINAL PAPER

Biological Cybernetics

Francesco Nori · Ruggero Frezza

A control theory approach to the analysis and synthesis of the experimentally observed motion primitives

Received: 23 November 2004 / Accepted: 1 July 2005

Abstract Recent experiments on frogs and rats, have led to the hypothesis that sensory-motor systems are organized into a finite number of linearly combinable modules; each module generates a motor command that drives the system to a predefined equilibrium. Surprisingly, in spite of the infiniteness of different movements that can be realized, there seems to be only a handful of these modules. The structure can be thought of as a vocabulary of "elementary control actions". Admissible controls, which in principle belong to an infinite dimensional space, are reduced to the linear vector space spanned by these elementary controls. In the present paper we address some theoretical questions that arise naturally once a similar structure is applied to the control of nonlinear kinematic chains. First of all, we show how to choose the modules so that the system does not loose its capability of generating a "complete" set of movements. Secondly, we realize a "complete" vocabulary with a minimal number of elementary control actions. Subsequently, we show how to modify the control scheme so as to compensate for parametric changes in the system to be controlled. Remarkably, we construct a set of modules with the property of being invariant with respect to the parameters that model the growth of an individual. Robustness against uncertainties is also considered showing how to optimally choose the modules equilibria so as to compensate for errors affecting the system. Finally, the motion primitive paradigm is extended to locomotion and a related formalization of internal (proprioceptive) and external (exteroceptive) variables is given.

1 Introduction

Each time that a living creature performs a task, its central nervous system (CNS) has to solve a nontrivial control problem formulated as follows: *control the activation of the muscles so that the whole body accomplishes the given task.* The solution of this problem is a hard challenge, especially because of the system nonlinearities and the need of coordinating a high number of degrees of freedom. Nevertheless, living creatures solve complicated tasks efficiently, displaying robustness against external disturbances and compensating modifications of the system dynamics.

Many studies focus on investigating the way the CNS controls the limbs and achieves the observed robustness and adaptability. Interestingly, there is evidence for the hypothesis that the CNS possesses and updates an *internal model* that approximates the limb dynamics; the model transforms a desired movement into the corresponding motor command.

One striking feature of this internal model is its *adaptability to new contingencies*. As a simple example, consider the changes to which a skeleton of an individual is subject during his growth. Because of these changes, the motor commands that control the body of an adult differ from the commands of a young child. Nevertheless, adults do not need to relearn the movements learnt during childhood. This proves that human sensory-motor system adapts the internal model to compensate for changes in the system dynamics.

Moreover, there's evidence (Shadmehr and Mussa-Ivaldi 1994) supporting the hypothesis that the internal model *compensates for external disturbances and generalizes their effects*. Specifically, in a well known experiment, subjects were asked to perform reaching movements while holding a robotic handle that could exert unexpected forces during the movement. Initially, trajectories were strongly distorted. With practice subjects learned the motor command necessary to compensate the disturbances and were able to reproduce the trajectories followed in absence of the perturbing forces. Interestingly, subjects were consistently able to generalize training experience to movements around the area in which they were trained. When the forces were unexpectedly removed, subjects produced again distorted trajectories, which were mirror images of the trajectories initially produced when the forces were first applied. All findings justify the idea that movements are planned on the basis of an internal model and that, with experience, this model can be adapted to

F. Nori · R. Frezza (✉)
Department of Information Engineering,
Università degli Studi di Padova,
35131 Padova, Italy
E-mail: iron@dei.unipd.it, frezza@dei.unipd.it

reject disturbances and accommodate changes in the system dynamics.

All these observations motivate a number of interesting questions which are, at present, the focus of many research activities. Our interest, in particular, is focused on the following fundamental questions:

(1) Which information is contained in the internal model?
(2) How does the internal model generate different instances of the same action?
(3) How does the CNS adapt the internal model in response to internal and external disturbances?

Question (1) and (2) assume that the CNS exploits an internal model of the system dynamics or, better, of the task dynamics. Here, the task has to be identified with the objective of the action. Therefore, the first two questions are concerned with the representation of the system and task dynamics. Question (3), instead, concerns the way this representation is adapted to disturbances and changes in the system dynamics.

Recent experiments developed by (E. Bizzi, F. A. Mussa-Ivaldi and S. F. Giszter Giszter et al. 1993); (Mussa-Ivaldi and Bizzi 2000); (Mussa-Ivaldi et al. 1994) seem to indicate that the sensory-motor systems of frogs and rats are organized into a finite number of linearly combinable modules. These modules, called spinal fields, are located on the spinal cord and, when activated, generate force fields acting on the limbs. Mathematically, it is as if admissible controls were restricted to the vector space spanned by a handful of elementary control actions. Within this framework, the above questions can be reformulated as follows.

(1) Which information about the system dynamics can be represented in a spinal field and how can it be used to perform a given action?
(2) How does the CNS combine the elementary modules to generate different instances of the same action?
(3) How are the modules activated and modified to reject external disturbances and adapt to changes in the system dynamics?

In this paper we focus on these questions and propose a new perspective on their answers, adopting the fundamental tools of control systems theory. Specifically, question (1) is answered in terms of modules that combine a feedforward plus a feedback control action. Each module is "task oriented" in the sense that it contains the information to perform a specific instance of a given task. Question number (2) will be answered in terms of the linear superposition of the individual modules, with the sum of the linear combinators constrained to be one. Finally, question (3) will be answered in terms of a paradigm which accommodates changes in the system dynamics without modifying the elementary modules; furthermore, random disturbances will be accommodated with an optimal choice of the task instances realized by the modules.

1.1 Modularity and complexity

The idea of decomposing complex systems into elementary modules has been widely applied in the scientific literature. In particular, modularity has always been seen as a way to handle complexity and to simplify learning procedures.

In the field of action recognition, for instance, (D. Del Vecchio, R. Murray and P. Perona Del Vecchio et al. 2003) have interpreted human movements as the result of switching between different causal linear dynamic systems, thought of as elementary building blocks. Similarly, modularity has been applied to the study of planar reaching movements by (T. Sanger Sanger 2000), who has proposed a decomposition of human hand trajectories into a set of elementary movements, learned applying the Karhunen-Loeve decomposition. Interestingly, there seems to be only a handful of components relevant to the description of planar reaching trajectories.

A special attention deserves the application of modularity to learning. It has been pointed out (T.Poggio and S.Smale 2003) that modularity is the core element for learning simplified models of complex systems. These models can be used for generating intelligent behaviors such as data analysis and information extraction. Specifically, in the study of cognition and mental representations, compositionality leads to interesting interpretations in terms of disambiguation, invariance and computation (Bienenstock and Geman 1993).

In this paper our main focus is on the application of modularity to motor control. The results may have important implications on learning motion control, but this issue is not addressed here.

1.2 Outline of the Paper

The paper is organized as follows. Section 2.1 recalls the biological experiment that justifies the idea of decomposing control actions into primitives. Section 2.2 gives a mathematical formulation of the biological experiment; a definition for the fundamental concept of motion primitive is also given. Section 3 defines the problem of synthesizing motion primitives for controlling a given dynamical system to an arbitrary state. Section 4 solves the problem in the case of linear systems. Section 5.1 extends the solution to the case of nonlinear kinematic chains; section 5.2 shows that elementary control actions can be chosen so as to perform a movement in an arbitrary amount of time; section 5.3 demonstrates how to obtain primitives that are invariant with respect to the parameters that we use to model the growth of an individual. Section 6 is inspired by the observation that biological movements display more variability in directions irrelevant to the task than in those relevant. This suggest the problem of synthesizing "task oriented" motion primitives; the problem is solved in section 6.1.2. Section 7 shows how to optimally choose the convergence points realized by each primitive; optimality is intended in terms of disturbance rejection. In section 8 we exemplify our ideas applying our findings to the control of a simple limb model. Finally, section 9 gives a mathematical formulation to the problem of synthesizing motion primitives for locomotion.

The paper follows a standard notation. Given a column vector $\mathbf{v}$ in the n-dimensional Euclidean space $\mathbb{R}^n$, $v_k \in \mathbb{R}$

indicates its k-th component. The superscript, instead, will be used to denote different vectors $\mathbf{v}^1$, $\mathbf{v}^2$, etc. . Matrices are indicated with capital letters. Finally, given a dynamical system and a control action $\mathbf{u}$ defined on the time interval $[0, T]$, the notation $\mathbf{x}_0 \xrightarrow{\mathbf{u}} \mathbf{x}_f$ indicates that $\mathbf{u}$ drives the system state $\mathbf{x}$ from $\mathbf{x}(0) = \mathbf{x}_0$ to $\mathbf{x}(T) = \mathbf{x}_f$.

2 Spinal Fields and their Model

This section is divided into three subsections. We first describe the experiments that justify our control paradigm (section 2.1). Then we give a model of the experiments (section 2.2). Finally we discuss some related previous works (section 2.3).

2.1 Spinal Fields

(E. Bizzi, F. Mussa-Ivaldi and S. Giszter Giszter et al. 1993); (Mussa-Ivaldi and Bizzi 2000; Mussa-Ivaldi et al. 1994) have proposed some experiments suggesting the existence of motion primitives hardwired in the central nervous system of frogs and rats. These primitives act on limbs in terms of muscle synergies, called *spinal fields*. These synergies have been characterized in terms of the isometric force fields elicited at the limb extremity. The main features observed in the experiments are the following:

(a) each spinal field recruits a specific pattern of muscles that *direct the limb towards a given configuration*, regardless of the initial condition;
(b) simultaneous activation of multiple spinal fields leads to the *vectorial summation* of the corresponding force fields.

There is therefore evidence that the complex nonlinearities that characterize the limbs of living creatures are somehow eliminated, since the central nervous system acts linearly, applying the superposition principle.

2.2 Motion Primitives as a Model of the Spinal Fields

In this section we present the concept of motion primitive which we use to model the experimentally observed spinal fields. We propose a model which differs from the one given by (Mussa-Ivaldi and Bizzi in Mussa-Ivaldi and Bizzi 2000).

2.2.1 Limb Dynamics

A limb is modelled here as a fully actuated kinematic chain with m degrees of freedom, corresponding to m revolute joints. The dynamics take the form Murray *et al.* (1994):

$$M(\mathbf{q})\ddot{\mathbf{q}} + C(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + N(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{u}, \tag{1}$$

where $\mathbf{q} \in \mathbb{R}^m$ are the generalized coordinates describing the pose of the kinematic chain, $\mathbf{x} = [\mathbf{q}^\top, \dot{\mathbf{q}}^\top]^\top \in \mathbb{R}^n$ is the state of the dynamic system, $n = 2m$ is the state space dimension and $\mathbf{u} \in \mathbb{R}^m$ is the generalized force vector. To fix the ideas, let $\mathbf{q}$ be the vector of joint angles and $\mathbf{u}$ be the joint torques.

2.2.2 Previous Model of the Spinal Fields

In (Mussa-Ivaldi and Bizzi 2000) the experimentally observed spinal fields are proposed as the evidence of a modular structure at the level of joint torques. Each module is a time-varying nonlinear control action which depends on the current position $\mathbf{q}$ and velocity $\dot{\mathbf{q}}$:

$$\mathbf{u} = \Phi^k(\mathbf{x}(t), t) \qquad \Phi^k : \mathbb{R}^{2m} \times \mathbb{R}^+ \to \mathbb{R}^m. \tag{2}$$

In consideration of the experiments, each Φ^k has a unique convergence point. Moreover, admissible control actions are the linear combination of the elementary modules $\Phi^1, \ldots, \Phi^K$:

$$\mathbf{u} = \sum_{k=1}^{K} \lambda_k \Phi^k(\mathbf{x}(t), t). \tag{3}$$

This model presents some limitations. Specifically, the choice of a time-varying feedback on the *current* state does not present any drawback if the execution time is left free. However, if the action has to be completed in finite time T, the feedback control would necessarily diverge as t reaches T, unless we reduce ourselves to reach approximatively the final position up to a ball of radius ϵ determined by the resolution of the sensors. In this case, the final time T would correspond to the entry time in the ball and it would be finite even in the case of asymptotic convergence.

In this paper we focus instead on actions which concern reaching an exact target in a predetermined execution time.

2.2.3 A New Model of the Spinal Fields

We propose a different model where a spinal field corresponds to a control action that combines a feedforward plus a feedback term. The feedback stabilizes about the nominal open-loop trajectory generated by the application of the feedforward part. Interestingly, there is evidence for this structure. Experiments on deafferented monkeys revealed that during reaching "the muscular activation does not specify a force or a torque [...] nor a final target position. Instead, [...] the activation of muscles produces a gradual shift of the limb's equilibrium from start to end position" (Mussa-Ivaldi and Bizzi 2000), i.e. a nominal or *virtual trajectory* (Hogan 1985b).

We achieve this structure with a control action which depends on the *initial* state $\mathbf{x}(0)$, on the *current* state $\mathbf{x}(t)$ and on the time elapsed from the action beginning. The formalization of the model leads to the following definitions. It is worth noting that the proposed model, as well as the previous one, agrees with the experimental evidence outlined in section 2.1. Details are given in the appendix A.

Definition 1 (motion primitive) *Consider a dynamical system; let its state belong to the set $\mathcal{X}$ and let its initial condition belong to $\mathcal{X}_0$. We say that the function $\Phi : \mathcal{X}_0 \times \mathcal{X} \times [0, T] \to \mathbb{R}^m$ is a motion primitive with convergence point $\mathbf{x}_f$, if the control $\mathbf{u} = \Phi(\mathbf{x}(0), \mathbf{x}(t), t)$ drives the system toward the state $\mathbf{x}(T) = \mathbf{x}_f$, regardless of the initial condition.*

Definition 2 (control with motion primitives) *We say that a given system is controlled with the set of motion primitives*

$\{\Phi^1, \ldots, \Phi^K\}$, *if the input of the system belongs to the space spanned by the motion primitives, i.e.:*

$$\mathbf{u} = \sum_{k=1}^{K} \lambda_k \Phi^k(\mathbf{x}(0), \mathbf{x}(t), t). \tag{4}$$

Note 1 In order to be more general, we initially consider a generic initial time t_0. At the same time, to simplify the notation, motion primitives will be denoted $\Phi^k(\mathbf{x}_0, \mathbf{x}, t - t_0)$ instead of $\Phi^k(\mathbf{x}(t_0), \mathbf{x}(t), t - t_0)$.

2.3 Previous works on Motion Primitives

A fundamental problem concerns the existence of a set of motion primitives which can generate the wide repertoire of control patterns displayed by biological systems.

This problem was first faced by (Mussa-Ivaldi Mussa-Ivaldi 1997), who reformulated it as an approximation of the vector fields of generalized forces necessary to realize a set of desired movements. The approximation is achieved with a finite number of basis fields chosen to be gradients of Gaussian potential functions centered at different positions in the generalized coordinates space; other basis fields are obtained as the result of an antisymmetric transformation of the previous fields. The combinations of these basis fields (Mussa-Ivaldi and Giszter 1992) is locally capable of approximating different vector fields which correspond to a 'wide' repertoire of control patterns. The coefficients λ_k which combine the basis fields in order to follow a given trajectory are computed via a standard least-squares algorithm. Learning and adapting the motion primitives amount, in this setup, to finding the optimal approximating choice of the Gaussian means and variances.

In the present paper, we instead face the problem of finding motion primitives that generate a 'complete' repertoire of motion patterns; completeness will be characterized in terms of the capability of reaching an arbitrary state in an arbitrary amount of time. Notice that we use the adjective 'complete' rather than the previously used adjective 'wide'. The motivation behind this choice resides in that *the formalism of force fields turned out to be insufficient to predict the trajectory that the state will follow* (Mussa-Ivaldi et al. 1994); in fact, movements generally result from the interaction of forces with the inertias and the dynamical properties of the system. Our formulation differ substantially in that modules do not take the form of force vector fields but rather of elementary control actions. Using the tools given by system control theory, this alternative formulation predicts which configurations and trajectories are achievable and replaces the previous approximation approach with a direct synthesis which covers a complete set of tasks. Moreover, the new formalism will lead to primitives that operate over a large region of the state space, i.e. over a broad receptive field; this is a new feature since previous approaches, based on learning, result in narrow receptive fields (Mussa-Ivaldi 1999).

Other important theoretical questions will be faced in this paper. To our knowledge, these questions have never been discussed in literature before. First, we will consider the problem of determining a lower bound for the minimum number of motion primitives needed to perform a given action. Then, we will give some preliminary results on the adaptability of the motion primitives paradigm to parametric changes of the system. During the life of an individual, his skeleton is subject to changes which are naturally compensated by modifying the control strategy so as to restore the original trajectories. In our model we achieve this compensation by changing the time-invariant combinators, while primitives are left unchanged. A similar solution will be given to the problem of performing actions in an arbitrary amount of time. Finally, we will consider the problem of optimally placing the convergence points for achieving robustness against disturbances.

3 Completeness of a Set of Motion Primitives

In this section we formalize the problem of finding a set of primitives that generate a 'complete' repertoire of motion patterns. The formalization is inspired by the concept of *controllability* as defined in control systems theory. With this formalization, we are able to fix a lower bound on the minimum number of primitives necessary to cover a 'complete' set of tasks.

In control systems theory the input of a dynamical system is usually assumed to belong to an infinite dimensional space, e.g. the set of all piecewise continuous functions. Under this assumption, the system (1) can be proven to be controllable, in the sense that for any desired initial and final state $\mathbf{x}_0, \mathbf{x}_f$ there exists an input $\mathbf{u} : [0, T] \rightarrow \mathbb{R}^m$ driving the system state from $\mathbf{x}(0) = \mathbf{x}_0$ to $\mathbf{x}(T) = \mathbf{x}_f$.

In our framework, the structure (4) may corrupt controllability since it reduces the set of admissible inputs from an infinite dimensional space to the linear space spanned by the elementary control actions $\Phi^1, \ldots, \Phi^K$. This reduction to a smaller set of admissible inputs corresponds to a reduction of the movements that the system is capable of performing. Obviously, motion primitives should be chosen so as to preserve the system controllability. This issue is captured in the statement of the following problem. Here the execution time is constant and equals to T.

Problem 1 (Synthesis of Motion Primitives for Reaching.) *Consider a dynamic system and let its state* $\mathbf{x}$ *belong to the open set* $\mathcal{X} \subseteq \mathbb{R}^n$. *Find a set of motion primitives* $\{\Phi^1, \ldots, \Phi^K\}$ *and a continuously differentiable function* $\lambda : \mathcal{X} \rightarrow \mathbb{R}^K$, *such that for every desired final state* $\mathbf{x}_f \in \mathcal{X}$ *the input:*

$$\mathbf{u} = \sum_{k=1}^{K} \lambda_k(\mathbf{x}_f) \Phi^k(\mathbf{x}_0, \mathbf{x}, t - t_0) \tag{5}$$

steers the system state to $\mathbf{x}(T) = \mathbf{x}_f$ *regardless of the initial condition.*

Note 2 The problem above requires that, for all admissible $\mathbf{x}_f$, the input $\mathbf{u}$ defined in (5) has to be a motion primitive with convergence point $\mathbf{x}_f$.

Note 3 From now on, let us assume $t_0 = 0$.

3.1 Lower Bound on K

In this section we prove that for any given solution the number of primitives K is grater than or equal to n. Before giving the main result, we prove the following lemma, claiming the injectivity of the function λ.

Lemma 1 *Let $\{\Phi^1, \ldots, \Phi^K\}$ and $\lambda : \mathcal{X} \to \mathbb{R}^K$ be a solution to problem 1. Then λ is injective.*

Proof Suppose by contradiction that λ is not injective. Equivalently, $\exists \mathbf{x}_f^1, \mathbf{x}_f^2$ such that $\mathbf{x}_f^1 \neq \mathbf{x}_f^2$ but $\lambda(\mathbf{x}_f^1) = \lambda(\mathbf{x}_f^2)$. Define:

$$\mathbf{u}^1 \triangleq \sum_{k=1}^{K} \lambda_k(\mathbf{x}_f^1)\Phi^k(\mathbf{x}_0, \mathbf{x}, t), \tag{6}$$

$$\mathbf{u}^2 \triangleq \sum_{k=1}^{K} \lambda_k(\mathbf{x}_f^2)\Phi^k(\mathbf{x}_0, \mathbf{x}, t). \tag{7}$$

Under the given assumption $\mathbf{u}^1 = \mathbf{u}^2$ but this contradicts the fact that $\mathbf{u}^1$ drives the system to $\mathbf{x}_f^1$ while $\mathbf{u}^2$ drives the system to $\mathbf{x}_f^2 \neq \mathbf{x}_f^1$. By contradiction, this proves that λ is injective. □

Proposition 1 *Let $\{\Phi^1, \ldots, \Phi^K\}$ and $\lambda : \mathcal{X} \to \mathbb{R}^K$ be a solution to problem 1. Then $K \geq n$, i.e. the control of (1) requires at least n motion primitives.*

Proof Using lemma 1, we have that λ is an injective function from an open subset of $\mathbb{R}^n$ to $\mathbb{R}^K$. It can be proven that this implies $K \geq n$ (see Boothby (2002) for details). □

4 Primitives for Linear Systems

This section solves a particularization of problem 1 to the case of linear dynamical systems. The linear case is crucial in our formulation, since the motion primitives for nonlinear systems will be constructed combining a feedback linearization with a set of primitives for linear systems. The section is divided into two subsections. Section 4.1 presents a solution solely based on a feedforward control action. This solution is made more robust in section 4.2, by adding a feedback on the nominal trajectory generated by the application of the feedforward part.

We consider a controllable linear system:

$$\dot{\mathbf{z}} = A\mathbf{z} + B\mathbf{v}, \tag{8}$$

where $\mathbf{z}$ is the system state (belonging to the state space $\mathcal{Z} = \mathbb{R}^n$) and $\mathbf{v} \in \mathbb{R}^m$ is the input. The particularization of problem 1 to the case of linear dynamics consists in finding a set of motion primitives $\{\phi^1, \ldots, \phi^K\}$ and a function $\lambda : \mathcal{Z} \to \mathbb{R}^K$ such that $\forall \mathbf{z}_f$ the input:

$$\mathbf{v} = \sum_{k=1}^{K} \lambda_k(\mathbf{z}_f)\phi^k(\mathbf{z}_0, \mathbf{z}, t), \tag{9}$$

drives the system to $\mathbf{z}(T) = \mathbf{z}_f$ regardless of the initial condition.

Note 4 For linear systems we use the letters $\mathbf{z}$ and $\mathbf{v}$ instead of the letters $\mathbf{x}$ and $\mathbf{u}$ to indicate the state and the input respectively. Moreover, motion primitives are indicated ϕ^k instead of Φ^k. The reason for this notation will be clear later on.

Note 5 The well known superposition principle suggests how primitives for linear systems can be chosen and combined. Since the state space is a linear vector space, it is intuitive to take each primitive to be a control that drives the system towards an element of a state space basis. These ideas lead to an instructive solution. Details are given in appendix B.

4.1 Feedforward Primitives

In this section we introduce the feedforward component of our control scheme. Within this context, motion primitives take the form of *feedforward motion primitives* $\zeta^k(\mathbf{z}(0), t)$ where $\zeta^k : \mathcal{Z}_0 \times [0, T] \to \mathbb{R}^m$ and $\mathcal{Z}_0$ is the set of admissible initial conditions. Again, we assume that each primitive drives the system toward a unique state $\mathbf{z}_f^k$ regardless of the initial condition, that is:

$$\mathbf{z}_0 \xrightarrow{\zeta^k} \mathbf{z}_f^k, \qquad \forall \mathbf{z}_0 \in \mathcal{Z}_0. \tag{10}$$

Using the tools given by linear systems theory, we first construct and characterize these primitives. Then, we show how feedforward primitives have to be combined so that their sum is again a motion primitive. Finally, all these considerations are used to characterize and construct the solutions of problem 1.

Proposition 2 *The control action $\zeta : \mathcal{Z}_0 \times [0, T] \to \mathbb{R}^m$ is a feedforward motion primitive with convergence point $\mathbf{z}_f$ if and only if:*

$$\zeta(\mathbf{z}_0, t) = B^\top e^{A^\top(T-t)} W_T^{-1} \left[\mathbf{z}_f - e^{AT}\mathbf{z}_0\right] + \mathbf{w}(\mathbf{z}_0, t), \tag{11}$$

where $\mathbf{w} : \mathcal{Z}_0 \times [0, T] \to \mathbb{R}^m$ is such that $0 \xrightarrow{\mathbf{w}} 0$, $\forall \mathbf{z}_0$.

Proof The hypothesis of controllability on the linear system guarantees the existence of an input which drives the system between two arbitrary states. More precisely, it can be proven Fornasini (1994) that any $\mathbf{v}$ such that $\mathbf{z}_0 \xrightarrow{\mathbf{v}} \mathbf{z}_f$ can be written as:

$$\mathbf{v}(t) = B^\top e^{A^\top(T-t)} W_T^{-1} \left[\mathbf{z}_f - e^{AT}\mathbf{z}_0\right] + \mathbf{w}(t), \tag{12}$$

where W_T is the system controllability Gramian and where $\mathbf{w} : [0, T] \to \mathbb{R}^m$ is any input such that $0 \xrightarrow{\mathbf{w}} 0$. To obtain (11) we need only to observe that in our case $\mathbf{w}$ can itself be a function of the initial condition. □

The following proposition describes how feedforward motion primitives can be combined. Its proof is given in the appendix.

Proposition 3 *Let $\zeta^1 : \mathcal{Z}_0 \times [0, T] \to \mathbb{R}^m$ and $\zeta^2 : \mathcal{Z}_0 \times [0, T] \to \mathbb{R}^m$ be a couple of feedforward motion primitives for the linear system (8); let $\mathbf{z}_f^1$ and $\mathbf{z}_f^2$ be their convergence* *points. Consider two scalars* λ_1 *and* λ_2. *Then, the sum* $\lambda_1\zeta^1 + \lambda_2\zeta^2$ *is itself a feedforward motion primitive if and only if* $\lambda_1 + \lambda_2 = 1$. *In this case we have:*

$$\mathbf{z}_0 \overset{\lambda_1\zeta^1+\lambda_2\zeta^2}{\longrightarrow} \lambda_1\mathbf{z}_f^1 + \lambda_2\mathbf{z}_f^2. \tag{13}$$

Moreover, the application of the input $\mathbf{u} = \lambda_1\zeta^1+\lambda_2\zeta^2$ *forces the system to follow:*

$$\mathbf{z}(t) = \lambda_1\mathbf{z}^1(t) + \lambda_2\mathbf{z}^2(t) \quad t \in [0, T], \tag{14}$$

where $\mathbf{z}^1$ *and* $\mathbf{z}^2$ *are the state trajectories followed when* ζ^1 *and* ζ^2 *are applied alone.*

Proposition 3 and its trivial generalization (to the case of finite sums) are crucial for characterizing the solutions of problem 1 based on feedforward motion primitives. Specifically, in the following proposition the generalization is used to give necessary and sufficient conditions for a given set of feedforward motion primitives to be a solution of the reaching problem.

Proposition 4 *Consider the controllable linear system (8) with* $\mathcal{Z}_0 = \mathcal{Z} = \mathbb{R}^n$. *Let* $\{\zeta^1(\mathbf{z}_0, t), \ldots, \zeta^K(\mathbf{z}_0, t)\}$ *be a set of feedforward motion primitives with convergence points* $\mathbf{z}_f^1, \ldots, \mathbf{z}_f^K$. *There exists a function* $\lambda : \mathbb{R}^n \to \mathbb{R}^K$ *that, together with the given primitives, solves problem 1 if and only if the linear system:*

$$\sum_{k=1}^{K} \lambda_k \mathbf{z}_f^k = \mathbf{z}_f \tag{15a}$$

$$\sum_{k=1}^{K} \lambda_k = 1. \tag{15b}$$

is solvable in the unknowns $\lambda_1, \ldots, \lambda_K$ *for all* $\mathbf{z}_f \in \mathbb{R}^n$. *Moreover, any solution to problem 1 based on the given primitives has combinators* $\lambda(\mathbf{z}_f)$ *that satisfy (15). Viceversa, if the combinator function* $\lambda(\mathbf{z}_f)$ *satisfies (15), then (together with the given primitives) it solves problem 1.*

Proof ($\Leftarrow$) Let $\{\zeta^1(\mathbf{z}_0, t), \ldots, \zeta^K(\mathbf{z}_0, t)\}$ be a set of motion primitives and assume that (15) is solvable. We want to show that the given primitives solve problem 1 together with $\lambda : \mathbb{R}^n \to \mathbb{R}^K$ obtained as any function which associates to each $\mathbf{z}_f \in \mathbb{R}^n$ one of the solutions of (15). Keeping in mind the generalization of proposition 3 together with (15b), we have that:

$$\mathbf{v} = \sum_{k=1}^{K} \lambda_k(\mathbf{z}_f)\zeta^k(\mathbf{z}_0, t), \tag{16}$$

is itself a motion primitive for all $\mathbf{z}_f \in \mathbb{R}^n$. Then, using (15a) together with a generalization of (13) we conclude that this motion primitive has $\mathbf{z}_f$ as its convergence point. This concludes the first part of the proof.
($\Rightarrow$) Consider a generic solution $\left\{\zeta^1(\mathbf{z}_0, t), [4] \ldots, \zeta^K(\mathbf{z}_0, t)\right\}$ and $\lambda : \mathbb{R}^n \to \mathbb{R}^K$; we have to prove that (15) is solved by $\lambda(\mathbf{z}_f)$. According to note 2, any linear combination with combinators $\lambda(\mathbf{z}_f)$ has to be itself a motion primitive with convergence point $\mathbf{z}_f$. Using the generalization of proposition 3, we conclude that $\lambda_k(\mathbf{z}_f)$ must sum up to one, i.e. (15b) is satisfied; moreover, using a generalization of (13) we get (15a). This proves that (15) is solvable and that $\lambda(\mathbf{z}_f)$ is one of its solutions for all $\mathbf{z}_f$. □

Proposition 4 suggests a way to solve problem 1. The first step consists in using (11) to construct K motion primitives. The second step consists in choosing a combinator function $\lambda(\mathbf{z}_f)$ that satisfies (15). Obviously, the convergence points have to be chosen so that (15) is solvable. The following proposition gives a necessary and sufficient condition on $\mathbf{z}_f^1, \ldots, \mathbf{z}_f^K$ for the solvability of (15).

Proposition 5 *The linear system (15) is solvable for all* $\mathbf{z}_f$ *in* $\mathbb{R}^n$ *if and only if:*

$$\mathrm{Span}(\mathbf{z}_f^1 - \mathbf{z}_f^K, \ldots, \mathbf{z}_f^{K-1} - \mathbf{z}_f^K) = \mathbb{R}^n. \tag{17}$$

Proof Easy substitutions allow to write (15) as follows:

$$\sum_{k=1}^{K-1} \lambda_k(\mathbf{z}_f^k - \mathbf{z}_f^K) = \mathbf{z}_f - \mathbf{z}_f^K, \tag{18}$$

whose solution exists for all $\mathbf{z}_f$ if and only if condition (17) is satisfied. □

Trivially, (17) can be satisfied only if $K \geq n + 1$; therefore, if primitives depend only on the initial condition, the lower bound given by proposition 1 is not achievable. A solution with $K = n + 1$ primitives can instead be easily found requiring $\mathbf{z}_f^1 - \mathbf{z}_f^{n+1}, \ldots, \mathbf{z}_f^n - \mathbf{z}_f^{n+1}$ to be a basis of $\mathbb{R}^n$. From now on, we will consider solutions with a minimum number of motion primitives, i.e. $K = n + 1$. In this case there's a unique solution of (15) given by:

$$\lambda(\mathbf{z}_f) = \Lambda^{-1}\begin{bmatrix}\mathbf{z}_f \\ 1\end{bmatrix}, \qquad \Lambda = \begin{bmatrix}\mathbf{z}_f^1 & \ldots & \mathbf{z}_f^{n+1} \\ 1 & \ldots & 1\end{bmatrix}, \tag{19}$$

where the invertibility of the matrix is guaranteed by (17).

Note 6 In some approaches combinators are required to be positive. Different considerations suggest to impose such a constraint on the combinators. First of all, at the actuator level, muscles cannot push and this constraint can be enforced with a positivity constraint on the combinators. In fact, if each individual primitive (thought of as a muscle synergy) does not push, then the same holds true for the linear combination of primitives if combinators are chosen positive. Secondly, the positivity of combinators is a sufficient (but not necessary) condition for the stability of the superposition of stabilizing force fields (see Slotine 2003 for details).

In our formulation, (15) is solvable by positive combinators if $\mathbf{z}_f$ belongs to the convex hull defined by $\mathbf{z}_f^1, \ldots, \mathbf{z}_f^K$. Therefore, our approach can be easily generalized to positive combinators if the set to be reached, $\mathcal{Z}$ or $\mathcal{X}$, is convex.

4.1.1 Generalization to LQ Optimal Motion Primitives

In this section we show that the feedforward motion primitives $\zeta^k(\mathbf{z}_0, t)$ introduced in section 4.1 can be chosen to be optimal in a linear quadratic (LQ) sense. Other choices are feasible using (11) but LQ optimality presents some good control qualities which we want to take advantage of. Moreover, "the use of optimization to model natural behavior is appealing because of the analogy it bears to the optimization presumed to occur as a result of natural selection" (quoting T. Flash in Flash and Hogan 1985).

Consider the LQ optimal control problem with fixed final state Lewis and Syrmos (1995):

$$\min_{\mathbf{u}} \frac{1}{2}\int_0^T \left[\mathbf{z}^\top(t)Q\mathbf{z}(t) + \mathbf{v}^\top(t)R\mathbf{v}(t)\right] \mathrm{d}t, \tag{20}$$

subject to:

$$\begin{cases} \dot{\mathbf{z}} = A\mathbf{z} + B\mathbf{v} \\ \mathbf{z}(0) = \mathbf{z}_0 \in \mathbb{R}^n, \mathbf{z}(T) = \mathbf{z}_f \in \mathbb{R}^p \end{cases} \tag{21}$$

with (A, B) controllable. The solution $\mathbf{v}^*$ is known to be composed of two time-varying terms, the first depending linearly on $\mathbf{z}_0$, the second depending linearly on the final condition $\mathbf{z}_f$ (see Lewis and Syrmos (1995) for details):

$$\mathbf{v}^*(t; \mathbf{z}_0, \mathbf{z}_f) = H^1(t)\mathbf{z}_0 + H^2(t)\mathbf{z}_f. \tag{22}$$

Coming back to our problem, define:

$$\zeta^k(\mathbf{z}_0, t) = H^1(t)\mathbf{z}_0 + H^2(t)\mathbf{z}_f^k, \tag{23}$$

and choose $\mathbf{z}_f^1, \ldots, \mathbf{z}_f^{n+1}$ so as to satisfy (17). Then, a solution to the reaching problem, is obtained choosing a combinator function $\lambda(\mathbf{z}_f)$ that solves (15). Since $K = n+1$, the solution is unique and is given by (19).

4.2 Feedback on the Nominal Trajectory

The solution proposed in section 4.1 is not robust. The presence of disturbances and errors in the system dynamics may preclude the achievement of the goal. Robustness can be obtained adding a feedback on the error between the nominal trajectory $\mathbf{z}_d$ and the actual trajectory:

$$\mathbf{v} = G[\mathbf{z}(t) - \mathbf{z}_d(\mathbf{z}_0, t)] + \sum_{k=1}^{n+1} \lambda_k(\mathbf{z}_f)\zeta^k(\mathbf{z}_0, t), \tag{24}$$

where $G \in \mathbb{R}^{m\times n}$ is the gain matrix. The trajectory $\mathbf{z}_d$ is the trajectory that the system would follow in absence of errors; therefore, a generalization of (14) leads to:

$$\mathbf{z}_d(\mathbf{z}_0, t) = \sum_{k=1}^{n+1} \lambda_k \mathbf{z}_d^k(\mathbf{z}_0, t) \qquad t \in [0, T], \tag{25}$$

where $\mathbf{z}_d^k$ is the nominal trajectory followed when $\mathbf{v} = \phi_k(\mathbf{z}_0, t)$. Substituting this expression into (24) and using the fact that combinators sum up to one (15b), we finally get:

$$\mathbf{v} = \sum_{k=1}^{n+1} \lambda_k(\mathbf{z}_f) \underbrace{\left\{\zeta^k(\mathbf{z}_0, t) + G\left[\mathbf{z}(t) - \mathbf{z}_d^k(\mathbf{z}_0, t)\right]\right\}}_{=\phi^k(\mathbf{z}_0, \mathbf{z}, t)}. \tag{26}$$

Obviously, each ϕ^k inherits from ζ^k the property of being a motion primitive. Therefore, from the solution $\{\zeta^1(\mathbf{z}_0, t), \ldots, \zeta^{n+1}(\mathbf{z}_0, t)\}$, we have obtained a robust solution $\{\phi^1(\mathbf{z}_0, \mathbf{z}, t), \ldots, \phi^{n+1}(\mathbf{z}_0, \mathbf{z}, t)\}$ with the same combinator function $\lambda(\mathbf{z}_f)$ given by (19).

Note 7 The proposed solution is similar to an impedance control Hogan 1985a. Let η be the error between the nominal and the actual trajectories, i.e. $\eta = \mathbf{z} - \mathbf{z}_d$. If there's no error on the system dynamics, we have:

$$\dot{\eta} = (A + BG)\eta, \tag{27}$$

where the gain matrix $G \in \mathbb{R}^{m\times n}$ is chosen so as to obtain the desired dynamic behavior for η. Usually G is assumed to be time-invariant and chosen so as to make the matrix $A + BG$ stable. However, while performing point to point movements, one is mainly interested in annihilating the error at the end of the movement duration (Todorov and Jordan 2002). In these situations a time-varying gain is preferable and an optimal choice of this gain can be obtained solving a fixed-final-state LQ optimal control problem with a time-varying cost function (see Lewis and Syrmos 1995 for details).

5 Primitives for Nonlinear Kinematic Chains

In the previous section we have proposed a solution to problem 1 in the case of linear dynamics. In this section we extend the above solution to the nonlinear dynamic model of a limb. Having in mind the application to biological motor control, we also study two related problems. Section 5.2 considers the problem of performing an action in an arbitrary execution time. Section 5.3 deals with the problem of compensating the changes to which the skeleton of an individual is subject during his growth.

5.1 Input to State Feedback Linearization

Consider the model (1) with $\mathbf{x} \in \mathcal{X} = \mathbb{R}^n$ and $\mathbf{u} \in \mathbb{R}^m$ with $n = 2m$. It can be easily shown that this model is input to state feedback linearizable (see Slotine and Li 1991 for details), i.e. there exists a feedback that makes the state dynamics linear. In fact, take:

$$\mathbf{u} = M(\mathbf{q})\mathbf{v} + C(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + N(\mathbf{q}, \dot{\mathbf{q}}), \tag{28}$$

and use the invertibility of $M(\mathbf{q})$ to obtain $\ddot{\mathbf{q}} = \mathbf{v}$ whose state space realization is:

$$\dot{\mathbf{z}} = A\mathbf{z} + B\mathbf{v}, \quad A = \begin{bmatrix} 0 & I \\ 0 & 0 \end{bmatrix}, B = \begin{bmatrix} 0 \\ I \end{bmatrix}, \tag{29}$$

where the state of the linear system $\mathbf{z}$ has been chosen equal to the state of the kinematic chain $\mathbf{x} = [\mathbf{q}^\top, \dot{\mathbf{q}}^\top]^\top$. Clearly, results in Section 4.2 can now be directly applied to the linearized dynamics and the new input $\mathbf{v}$ can be written as the linear combination of few motion primitives ϕ^k. These primitives must be modified to obtain the primitives Φ^k for the original input $\mathbf{u}$. Specifically, the input that drives the system to $\mathbf{x}_f$ regardless of the initial condition can be obtained combining (28) with (26):

$$\begin{aligned}\mathbf{u} &= M(\mathbf{q}) \sum_{k=1}^{n+1} \lambda_k(\mathbf{z}_f) \phi^k(\mathbf{z}_0, \mathbf{z}, t) + C(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + N(\mathbf{q}, \dot{\mathbf{q}}) \\ &= \sum_{k=1}^{n+1} \lambda_k(\mathbf{z}_f) \left[M(\mathbf{q}) \phi^k(\mathbf{z}_0, \mathbf{z}, t) + C(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + N(\mathbf{q}, \dot{\mathbf{q}}) \right] \\ &= \sum_{k=1}^{n+1} \lambda_k(\mathbf{x}_f) \Phi^k(\mathbf{x}_0, \mathbf{x}, t),\end{aligned}$$

where we used the fact that the λ_k sum up to one and we have defined:

$$\Phi^k(\mathbf{x}_0, \mathbf{x}, t) = M(\mathbf{q}) \phi^k(\mathbf{x}_0, \mathbf{x}, t) + C(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + N(\mathbf{q}, \dot{\mathbf{q}}).$$

Notice that Φ^k inherits from ϕ^k the property of driving the system to $\mathbf{z}_f^k$ and thus the set $\{\Phi^1, \ldots, \Phi^{n+1}\}$ is effectively a set of motion primitives that solves problem 1 for the nonlinear system (1). Despite the nonlinearity of the dynamics, the number of primitives turns out to be $n+1$ as in the linear case. Interestingly, the fact that the combinators sum up to one (see (15)) has been fundamental to obtain this result.

5.2 Time Scaled Motion Primitives

In the previous section we have shown how to synthesize primitives for a nonlinear kinematic chain; the execution time was assumed to be constant and equal to T. In this section we extend previous results to the case of arbitrary execution times[1].

We consider the case of synthesizing primitives that performs reaching in the time interval $[0, \frac{T}{\alpha}]$. If we repeat the above procedure with $\frac{T}{\alpha}$ in place of T, we obtain different primitives for different values of the parameter α, i.e.:

$$\mathbf{u} = \sum_{k=1}^{n+1} \lambda_k(\mathbf{x}_f) \Phi^{k,\alpha}(\mathbf{x}_0, \mathbf{x}, t).$$

The following proposition shows that a much more desirable decomposition can be obtained with kinematic chains; the fundamental step consists in taking advantage of the double integrator structure (29) that results from the application of the feedback linearization. The proof of the proposition, being irrelevant to the main stream, will be given in the appendix.

Proposition 6 *Consider the nonlinear system (1). There exists a set of motion primitives $\{\Phi^1, \ldots, \Phi^K\}$ and a function $\lambda : \mathbb{R}^n \times \mathbb{R}^+ \to \mathbb{R}^K$ such that the decomposition:*

$$\mathbf{u} = \sum_{k=1}^{K} \lambda_k(\mathbf{x}_f, \alpha) \Phi^k(\mathbf{x}_0, \mathbf{x}, \alpha t), \tag{30}$$

solves the problem 1 with execution time $\frac{T}{\alpha}$.

The above proposition presents an interesting decomposition for accomplishing the reaching task in an arbitrary execution time. The coefficients λ_k depend on the desired task and on the desired execution time. Conversely, motion primitives are almost invariant to these quantities, except for the necessary time scaling.

5.3 Dependence of Motion Primitives on Kinematic Parameters

In this section we want to understand how primitives should change in order to accommodate modifications of the link masses, inertias, centers of mass and positions of the joints; with these parameters we intend to model the growth of an individual. We will not model changes in the direction of the rotational axes that describe the joints of the kinematic chain. Using the Danavit-Hartenberg notation to describe the chain, the assumption implies that the rotation matrices among the chain frames do not change, but the translation do. The motivation of such a choice resides in that these quantities are intuitively invariant in the skeletal structure of a growing up individual.

Consider a kinematic chain, composed of m link joined by m revolute joints; the assumption that the number of links equals the number of joints is nonrestrictive and is made for purposes of simplicity. We number the joints from 1 to m, starting at the base. Let m_i and $\mathcal{I}^i$ be the mass and inertia tensor of the i^{th} link. Joint positions and centers of mass are specified with respect to an inertial reference frame Σ_0 when all the joints are held fixed at the *reference configuration* $q_j = 0, \forall j$ (see Murray *et al.* 1994). Specifically, let $\mathbf{l}^i$ be a point on the joint axis of rotation; let $\mathbf{c}^i$ be the center of mass. Both are specified in the reference configuration. Let's put all the parameters together in a vector $\mathbf{p} \in \mathbb{R}^P$:

$$\mathbf{p} = \left[m_i \; I_1^i \; \ldots \; I_6^i \; \mathbf{l}^{i\top} \; \mathbf{c}^{i\top} \right]^\top, \tag{31}$$

where $I_1^i, \ldots, I_6^i$ represent the entries of the symmetric inertia tensor $\mathcal{I}^i$. The following proposition claims the existence of a set of motion primitives that do not depend on the considered parameters: parametric changes are accommodated by simply modifying the time-invariant combinators with no need of recomputing the motion primitives.

Proposition 7 *Consider the nonlinear system (1). There exists a set of control actions $\Phi^{k,h}(\mathbf{x}, t)$, a function $\lambda : \mathbb{R}^n \to \mathbb{R}^K$ and a function $\mu : \mathbb{R}^P \to \mathbb{R}^H$ such that for any $\mathbf{p}$ and any $\mathbf{x}_f$ the input:*

$$\mathbf{u} = \sum_{k,h} \lambda_k(\mathbf{x}_f) \mu_h(\mathbf{p}) \Phi^{k,h}(\mathbf{x}_0, \mathbf{x}, t), \tag{32}$$

[1] Here the problem of adjusting the execution time is considered from a purely theoretical point of view. In real world situations, the control action should account for limits on how rapidly movements can be executed. Within the proposed framework, taking this into account in the model will only be a matter of imposing bounds on the magnitude of the combinators.

steers the system with parameters $\mathbf{p}$ *to* $\mathbf{x}_f$ *regardless of the initial condition. Moreover, each* $\Phi^{k,h}$ *is a motion primitive in the sense that it drives a system with parameters* $\mathbf{p}^h$ *towards a state* $\mathbf{x}_f^k$ *regardless of the initial condition.*

Note 8 The proposed paradigm compensates parametric changes so as to maintain the resulting system trajectories unchanged. This solution is reasonable since similar behaviors are observed in living organisms: perturbations are compensated trying to modify the control action so as to obtain the same trajectories of the unperturbed system (Mussa-Ivaldi and Bizzi 2000).

6 Task Oriented Motion Primitives

In the statement of problem 1 we implicitly made the assumption that the task of our control is to drive the kinematic chain to a desired configuration. However, in many actions, we are not interested in controlling the kinematic chain configuration but rather a function of its configuration. This distinction is crucial because if the objective is the achievement of a given task, then, accordingly, the *control action should be expressed in the task space* as proposed by (C. Samson in Samson *et al.* 1991).

Interestingly, experiments tell us that even humans express their control actions in the task space while performing certain kind of movements[2]. There is indeed evidence (Todorov and Jordan 2002) that redundant tasks are achieved with much more variability in individual degrees of freedom than in task relevant movement parameters. As an example, humans, when performing hand movements, display much more variability in the joint angles of the arm than in trajectory followed by the hand.

In this section we restate problem 1 in terms of an arbitrary task function. The above considerations lead to a set of motion primitives defined in the task space. The given formulation will reduce the minimum number of primitives.

6.1 Motion Primitives for Output Reaching

Consider the dynamic system (1). Suppose that our task is the control of $\mathbf{z} = [\mathbf{y}^\top, \dot{\mathbf{y}}^\top]^\top$ where $\mathbf{y}$ is a generic function of the system configuration i.e. $\mathbf{y} = h(\mathbf{q}) \in \mathbb{R}^p$, with $h : \mathbb{R}^m \to \mathbb{R}^p$. Reasonably, we can assume that $p \leq m$. The above considerations imply that a task oriented control action should care only for the achievement of the task. Therefore, we have the following definition.

Definition 3 (task oriented motion primitive) *Consider a dynamical system; let its output* $\mathbf{z}$ *belong to the set* $\mathcal{Y}$*. We say that the function* $\Phi : \mathcal{X}_0 \times \mathcal{X} \times [0, T] \to \mathbb{R}^m$ *is a task oriented motion primitive with convergence point* $\mathbf{z}_f$*, if the control* $\mathbf{u} = \Phi(\mathbf{x}_0, \mathbf{x}, t)$ *drives the system output toward* $\mathbf{z}(T) = \mathbf{z}_f$*, regardless of the initial condition.*

[2] Reaching in absence of obstacles is a movement that humans seem to plan in the task space. However, in presence of obstacles, humans are also concerned about avoiding obstacles with the rest of the arm. In this situation a configuration space planning seems to be more suitable than the task space planning.

Problem 2 (Synthesis of Motion Primitives for Output Reaching Tasks.) *Consider the dynamic system (1) and let* $\mathbf{z} \in \mathcal{Y}$*. Find a set of task oriented motion primitives* $\{\Phi^1, \ldots, \Phi^K\}$ *and a continuously differentiable function* $\lambda : \mathcal{Y} \to \mathbb{R}^K$ *such that for every* $\mathbf{z}_f \in \mathcal{Y}$*, the input:*

$$\mathbf{u} = \sum_{k=1}^{K} \lambda_k(\mathbf{z}_f)\Phi^k(\mathbf{x}_0, \mathbf{x}, t) \tag{33}$$

steers the output to $\mathbf{z}_f$ *regardless of the initial condition.*

A trivial solution to problem 2 can be obtained with a reformulation in terms of problem 1. The key step is the solution of an inverse kinematic problem that consists in finding a state $\mathbf{x}_f$ whose corresponding output position and velocity are exactly $\mathbf{y}_f$ and $\dot{\mathbf{y}}_f$. After this preliminary step, the output reaching is reformulated in terms of state reaching and can be solved as proposed in section 5.1. Therefore, the trivial solution is composed of $K = n + 1$ primitives. However these primitives are not task oriented. Moreover, it seems like the trivial solution uses more primitives than necessary. Considering what we have shown in sections 4.1 and 5.1, we might expect to be able to solve problem 2 with $2p + 1$ motion primitives. Section 6.1.2 will indeed show how this can be achieved. The proposed solution has a number of primitives ($K = 2p + 1$) generally smaller then the trivial solution, since $2p \leq 2m = n$.

6.1.1 Lower Bound on K

In this section we extend the result in section 3.1 in order to fix a lower bound on the number of primitives that solve problem 2.

Proposition 8 *Let* $\{\Phi^1, \ldots, \Phi^K\}, \lambda : \mathcal{Y} \to \mathbb{R}^K$ *be a solution to problem 2. Then* $K \geq 2p$*.*

Proof The proof follows the same line of the proof in section 3.1, starting with proving that the function $\lambda : \mathcal{Y} \to \mathbb{R}^K$ is injective. □

6.1.2 Input to Output Feedback Linearization

In this section we synthesize a set of task oriented motion primitives. The fundamental observation is that the system (1) with output $\mathbf{y} = h(\mathbf{q})$ is input to output feedback linearizable (see Isidori 1995 for the definitions and Nori and Frezza 2004b for details). Practically, the linearization constructs a feedback that makes the input to output relation linear. This feedback is defined in every nonsingular configuration of the kinematic chain. The linearizing feedback has the form:

$$\mathbf{u} = \alpha(\mathbf{x})\mathbf{v} + \beta(\mathbf{x}). \tag{34}$$

The corresponding input to output linear relation with its state space realization are given by:

$$\ddot{\mathbf{y}} = \mathbf{v} \quad \longrightarrow \quad \dot{\mathbf{z}} = A\mathbf{z} + B\mathbf{v}. \tag{35}$$

In this case the state of the linear system can be chosen to be exactly $\mathbf{z} = [\mathbf{y}^\top, \dot{\mathbf{y}}^\top]^\top$. Once again, results in Section 4.2 can be directly applied to the linear dynamics and the new input $\mathbf{v}$ can be rewritten as the linear combination of $K = 2p+1$ motion primitives $\phi^k(\mathbf{z}_0, \mathbf{z}, t)$. The input that drives the system to $\mathbf{z}_f$ regardless of the initial condition can be obtained combining (34) with (26):

$$\begin{aligned}\mathbf{u} &= \alpha(\mathbf{x}) \left[\sum_{k=1}^{2p+1} \lambda_k(\mathbf{z}_f) \phi^k(\mathbf{z}_0, \mathbf{z}, t) \right] + \beta(\mathbf{x}) \\ &= \sum_{k=1}^{2p+1} \lambda_k(\mathbf{z}_f) \left[\alpha(\mathbf{x}) \phi^k(\mathbf{z}_0, \mathbf{z}, t) + \beta(\mathbf{x}) \right] \\ &= \sum_{k=1}^{2p+1} \lambda_k(\mathbf{z}_f) \Phi^k(\mathbf{x}_0, \mathbf{x}, t),\end{aligned}$$

where, using the fact that $\mathbf{z}$ can be expressed as a function of $\mathbf{x}$, we have defined:

$$\Phi^k(\mathbf{x}_0, \mathbf{x}, t) = \alpha(\mathbf{x}) \phi^k(\mathbf{z}_0, \mathbf{z}, t) + \beta(\mathbf{x}). \tag{36}$$

Once again the fact that λ_k sum up to one plays a fundamental role. Notice that Φ^k inherits from ϕ^k the property of driving the system to $\mathbf{z}_f^k$ and thus the set $\{\Phi^1, \dots, \Phi^{2p+1}\}$ is effectively a set of task oriented motion primitives that solves problem 2 for the nonlinear system (1).

Note 9 Since the input to output feedback linearization leads to a double integrator, results in section 5.2 can be applied to solve problem 2 with arbitrary execution time.

Note 10 The above procedure is valid for redundant ($p < n$) and non-redundant tasks ($n = p$). In the redundant case, the control of the degrees of freedom irrelevant to the task can be formalized as in (Samson *et al.* 1991) and (Nori and Frezza 2004b). The basic idea consists in augmenting $\mathbf{y} \in \mathbb{R}^p$ with an additional task $\hat{\mathbf{y}} \in \mathbb{R}^{n-p}$ that has to be driven to zero.

7 Optimal Positioning of the Convergence Points

In previous sections we have proposed solutions to problem 1 and 2. The motion primitives and the corresponding convergence points $\mathbf{x}_f^k$ (or more generally $\mathbf{z}_f^k$) can be chosen quite arbitrarily as long as condition (17) is satisfied. In this section we propose a technique for optimally configuring these convergence points. The choice criterion will be disturbance rejection.

Suppose we want to drive the system to $\mathbf{z}_f$. As already observed, since $K = n+1$, combinators are uniquely determined by (19). Before going on, we make and justify a crucial assumption.

Note 11 We will assume $|\det(\Lambda^{-1})| = 1$. This choice is motivated by the fact that we want the combinators λ to have the same variability as $\mathbf{z}_f \in \mathcal{Z}$; mathematically this condition corresponds to requiring the two sets $\mathcal{Z}$ and $\mathcal{L}$ to have the same hypervolume, being $\mathcal{L}$ the set spanned by the combinators λ when $\mathbf{z}_f$ spans $\mathcal{Z}$. This last condition is guaranteed requiring $|\det(\Lambda^{-1})| = 1$.

Let's now suppose that the convergence points realized by the motion primitives are affected by errors in the sense that they differ from the $\mathbf{z}_f^k$'s used to compute the combinators. In particular, let $\tilde{\mathbf{z}}_f^k = \mathbf{z}_f^k + \Delta\mathbf{z}_f^k$ be the state towards which the system is driven by Φ^k. Because of errors, the system is driven to $\tilde{\mathbf{z}}_f = \mathbf{z}_f + \mathbf{e}$ where $\mathbf{e}$ can be computed with the following expression:

$$\mathbf{e} = \sum_{k=1}^{K} \lambda_k(\mathbf{z}_f) \Delta\mathbf{z}_f^k. \tag{37}$$

Obviously, we are interested in keeping this error as small as possible choosing suitable values for $\mathbf{z}_f^1, \dots, \mathbf{z}_f^{n+1}$. The following proposition (whose proof is given in the appendix) formalizes the problem and gives a solution. It can be applied to all the motion primitives proposed in this paper.

Proposition 9 *Suppose that $\mathbf{z}_f$ is a random vector, uniformly distributed on a unitary sphere centered in the origin, i.e. $\mathbf{z}_f \sim \mathcal{U}(\mathcal{S}_0^1)$; moreover, assume $\mathbf{z}_f$ independent of the errors $\Delta\mathbf{z}_f^k$. Finally, suppose that the errors $\Delta\mathbf{z}_f^k$ are independent and identically distributed random vectors with zero mean and variance Σ_Δ. Then:*

$$E[\mathbf{e}] = 0, \qquad E[\mathbf{e}\mathbf{e}^\top] = \Sigma_\Delta E[\lambda(\mathbf{z}_f)^\top \lambda(\mathbf{z}_f)]. \tag{38}$$

Moreover, the error $\mathbf{e}$ with minimum variance is obtained choosing $\mathbf{z}_f^1, \dots, \mathbf{z}_f^{n+1}$ as follows:

$$\mathbf{z}_f^k = \frac{1}{\sqrt[2n]{n+1}} L^{-\top} \left(M\mathbf{v}^k - \mathbf{v} \right) \tag{39}$$

where $\mathbf{v}^1, \dots, \mathbf{v}^n$ is the canonical basis of $\mathbb{R}^n$, $\mathbf{v}^{n+1} = 0$, $\mathbf{v} = [1 \dots 1]^\top$, $M = I + \mathbf{v}\mathbf{v}^\top \in \mathbb{R}^{n\times n}$ and L is any square matrix satisfying the following: $L^\top L = M$.

Note 12 It can be proven that $M = M^\top > 0$. Therefore its square factorization $L^\top L = M$ is not unique. Specifically, it is well known that $L_1^\top L_1 = M = L_2^\top L_2$ if and only if there exists an orthogonal matrix Q such that $L_1 = QL_2$. Accordingly (see the proof for details), if $\mathbf{z}_f^1, \dots, \mathbf{z}_f^{n+1}$ is a minimum variance solution, then so is $Q\mathbf{z}_f^1, \dots, Q\mathbf{z}_f^{n+1}$ for any orthogonal change of basis Q.

Note 13 The sphere where $\mathbf{z}_f$ is distributed may be centered in any position other than the origin. If the center is $\mathbf{z}_c$ the minimal variance solution can be easily proven to be the translation of the solution above, i.e. $\mathbf{z}_c + \mathbf{z}_f^1, \dots, \mathbf{z}_c + \mathbf{z}_f^{n+1}$.

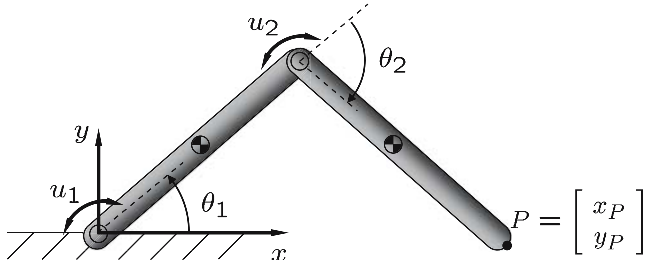

Fig. 1 Two degrees of freedom kinematic chain model of a limb for simulating planar reaching movements

8 Qualitative Comparison with Human Reaching Experiments

In this section we apply the proposed spinal field paradigm to control a 2DOF (two degrees of freedom) planar chain (see Fig. 1); this system has been used as a simplified model of the human arm in (Morasso 1981) and (Mussa-Ivaldi and Bizzi 2000). The dynamics of this model can be expressed in the form (1) with $m = 2$ and $\mathbf{q} = [\theta_1 \quad \theta_2]^\top$; the inputs are the torques applied at the joints, $\mathbf{u} = [u_1 \quad u_2]^\top$, and the task is the control of the cartesian position of the extremity P, i.e. $\mathbf{y} = [x_P \quad y_P]^\top$. Table 1 gives the kinematic and dynamic parameters that we used in the simulations. Table 2 gives the expressions of the matrices M and C (see Mussa-Ivaldi and Bizzi 2000 for details).

Following the ideas in section 6 we synthesized a set of task oriented motion primitives. Specifically, we used results in Section 4.1.1 and Section 4.2 to construct a set of robust and LQ-optimal motion primitives.

In a first simulation, the weighting matrices Q and R of the LQ cost (20) were chosen on the basis of the minimum energy paradigm (Uno *et al.* 1989): thus $Q = 0$ and $R = I$. In a second simulation, we slightly modified the procedure in 4.1.1 in order to obtain a set of motion primitives that realize a minimum jerk control for $\mathbf{y}$ (Flash and Hogan 1985).

Table 1 Parameters of the simulated human arm of Figure 1. This values are the same used in (Mussa-Ivaldi & Bizzi, 2000) and (Shadmehr & Mussa-Ivaldi, 1994)

Parameter	Upper arm	Forearm
mass	1.93 [kg]	1.52 [kg]
center of mass	0.165 [m]	0.19 [m]
inertia	0.0141 [$kg \cdot m^2$]	0.0188 [$kg \cdot m^2$]
length	0.33 [m]	0.34 [m]

Table 2 Explicit expressions for the matrices M and C (2 degrees of freedom planar arm). Parameters values are given in Table 1. The matrix N is in this case identically null since we are not considering external forces; even gravity does not affect the system since we are considering planar movements

Quantity	Expression
$M(\mathbf{q})$	$\begin{bmatrix} .0953\cos(q_2) + .1529 & .0477\cos(q_2) + .0368 \\ .0477\cos(q_2) + .0368 & .0368 \end{bmatrix}$
$C(\mathbf{q}, \dot{\mathbf{q}})$	$\begin{bmatrix} -.0477\sin(q_2)\dot{q_2} & -.0477\sin(q_2)\dot{q_1} - .0477\sin(q_2)\dot{q_2} \\ .0477\sin(q_2)\dot{q_1} & 0 \end{bmatrix}$

In both cases, the number of primitives was $K = 5$. Considering movements with null final velocity, the number of primitives was reduced to $K = 3$ (see below for details).

Derivation of (minimum jerk) motion primitives: Table 3 and Table 4 give a full derivation of a complete set of primitives for performing planar reaching with null initial and final velecity. The first step consists in computing the linearizing feedback, whose expression is given in Table 3 in terms of $\alpha(\mathbf{x})$ and $\beta(\mathbf{x})$. The second step consists in choosing a set of feedforward motion primitives ζ^k for the linearized system, $\ddot{\mathbf{y}} = \mathbf{v}$. The expressions for these primitives are given in Table

Table 3 Explicit expressions for the input to output feedback linearization (see (Nori & Frezza, 2004b) for a complete derivation). The function h gives the output as a function of the pose, i.e. $\mathbf{y} = [x_P \quad y_P]^\top = h(\mathbf{q})$. The matrix J (the Jacobian) gives $\dot{\mathbf{y}}$ as a function of $\dot{\mathbf{q}}$, i.e. $\dot{\mathbf{y}} = J(\mathbf{q})\dot{\mathbf{q}}$. We have, $\mathbf{z} = [h(\mathbf{q}) \quad J(\mathbf{q})\dot{\mathbf{q}}]^\top$ which gives $\mathbf{z}$ as a function of $\mathbf{x} = [\mathbf{q} \quad \dot{\mathbf{q}}]^\top$

Quantity	Expression
$h(\mathbf{q})$	$\begin{bmatrix} 0.33\sin(q_1) + 0.34\sin(q_1 + q_2) \\ 0.33\cos(q_1) + 0.34\cos(q_1 + q_2) \end{bmatrix}$
$J(\mathbf{q})$	$\begin{bmatrix} 0.33\cos(q_1) + 0.34\cos(q_1 + q_2) & 0.34\cos(q_1 + q_2) \\ -0.33\sin(q_1) - 0.34\sin(q_1 + q_2) & -0.34\sin(q_1 + q_2) \end{bmatrix}$
$\alpha(\mathbf{x})$	$M(\mathbf{q})J^{-1}(\mathbf{q})$
$\beta(\mathbf{x})$	$\left[-C(\mathbf{q}, \dot{\mathbf{q}}) + M(\mathbf{q})J(\mathbf{q})\dfrac{dJ}{dt}\right]\dot{\mathbf{q}}$

Table 4 Quantities for the derivation of a complete set of motion primitives (planar arm movements with null initial and final velocity). Primitives are chosen so as to obtain minimum jerk trajectories. The gain matrix G is chosen as in (Shadmehr & Mussa-Ivaldi, 1994)

Quantity	Expression
$\mathbf{z}$	$\begin{bmatrix} x_P & y_P & \dot{x}_P & \dot{y}_P \end{bmatrix}^\top$
$\mathbf{z}_0$	$\begin{bmatrix} x_{P,0} & y_{P,0} & 0 & 0 \end{bmatrix}^\top$
$\mathbf{z}_f^k$ $k = 1, 2, 3$	$\begin{bmatrix} x_{P,f}^k & y_{P,f}^k & 0 & 0 \end{bmatrix}^\top$
$\mathbf{z}_f^1$	$\begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}^\top$
$\mathbf{z}_f^2$	$\begin{bmatrix} 0 & 1 & 0 & 0 \end{bmatrix}^\top$
$\mathbf{z}_f^3$	$\begin{bmatrix} 1 & 1 & 0 & 0 \end{bmatrix}^\top$
$\zeta^k(\mathbf{z}_0, t)$ $k = 1, 2, 3$	$\begin{bmatrix} (x_{P,0} - x_{P,f}^k)\left[180(\frac{t^2}{T^4}) - 120(\frac{t^3}{T^5}) - 60(\frac{t}{T^3})\right] \\ (y_{P,0} - y_{P,f}^k)\left[180(\frac{t^2}{T^4}) - 120(\frac{t^3}{T^5}) - 60(\frac{t}{T^3})\right] \end{bmatrix}$
$\mathbf{z}_d^k(\mathbf{z}_0, t)$ $k = 1, 2, 3$	$\begin{bmatrix} x_{P,0} + (x_{P,0} - x_{P,f}^k)\left[15(\frac{t}{T})^4 - 6(\frac{t}{T})^5 - 10(\frac{t}{T})^3\right] \\ y_{P,0} + (y_{P,0} - y_{P,f}^k)\left[15(\frac{t}{T})^4 - 6(\frac{t}{T})^5 - 10(\frac{t}{T})^3\right] \\ (x_{P,0} - x_{P,f}^k)\left[60(\frac{t^3}{T^4}) - 30(\frac{t^4}{T^5}) - 30(\frac{t^2}{T^3})\right] \\ (y_{P,0} - y_{P,f}^k)\left[60(\frac{t^3}{T^4}) - 30(\frac{t^4}{T^5}) - 30(\frac{t^2}{T^3})\right] \end{bmatrix}$
$\phi^k(\mathbf{z}_0, \mathbf{z}, t)$ $k = 1, 2, 3$	$\zeta^k(\mathbf{z}_0, t) + G\left[\mathbf{z} - \mathbf{z}_d^k(\mathbf{z}_0, t)\right]$
G	$\begin{bmatrix} -15 & -6 & -2.3 & -0.9 \\ -6 & -16 & -0.9 & -2.4 \end{bmatrix}$
$\Phi^k(\mathbf{x}_0, \mathbf{x}, t)$ $k = 1, 2, 3$	$\alpha(\mathbf{x})\phi^k(\mathbf{z}_0, \mathbf{z}, t) + \beta(\mathbf{x})$

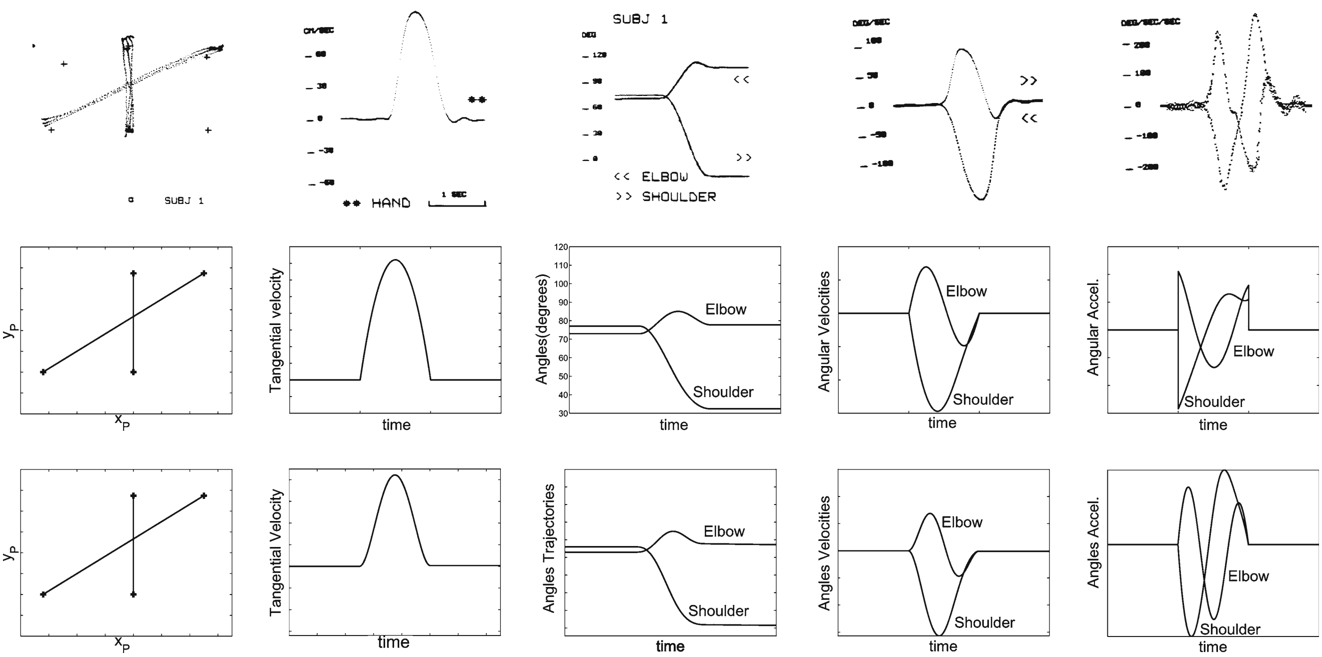

Fig. 2 Row 1: data obtained by Morasso with human subjects (taken from (Morasso, 1981)). Row 2: simulated results obtained using the superposition of motion primitives and the minimum energy paradigm. Row 3: simulated results using the superposition of motion primitives and the minimum jerk paradigm. Col. 1: trajectory; Col. 2: tangential velocity profile; Col. 3: angles; Col. 4: angle velocities; Col. 5: angles accelerations

4. They have been computed as follows: a set of convergence points $\mathbf{z}_f^1, \mathbf{z}_f^2, \mathbf{z}_f^3$ satisfying (17) has been first chosen; then, the feedforward motion primitives $\zeta^k(\mathbf{z}_0, t)$ together with the corresponding nominal trajectory $\mathbf{z}_d^k(\mathbf{z}_0, t)$ have been computed solving the following (minimum jerk) optimal control problems:

$$[\zeta^k, \mathbf{z}_d^k] = \arg\min_{\mathbf{v},\mathbf{z}} \int_0^T \left[\left(\frac{d^3 x_P}{dt^3} \right)^2 + \left(\frac{d^3 y_P}{dt^3} \right)^2 \right] \mathrm{d}t,$$

subject to:

$$\begin{cases} \dot{\mathbf{z}} = \begin{bmatrix} 0 & I \\ 0 & 0 \end{bmatrix} \mathbf{z} + \begin{bmatrix} 0 \\ I \end{bmatrix} \mathbf{v} \\ \mathbf{z}(0) = \mathbf{z}_0, \mathbf{z}(T) = \mathbf{z}_f^k \end{cases}, \tag{40}$$

where $\mathbf{z} = \begin{bmatrix} x_P & y_P & \dot{x}_P & \dot{y}_P \end{bmatrix}^\top$. Interestingly, the solution of this problem can be analytically computed Flash and Hogan (1985). Finally, the expressions for ϕ^k and Φ^k have been computed using (26) and (36), respectively.

Simulation of Reaching Tasks: We simulated some reaching experiments, i.e movements of the output to a desired final position $\mathbf{y}_f$ with null initial and final velocity. Simulations consist in applying the following torques to the planar chain:

$$\mathbf{u} = \sum_{k=1}^{3} \lambda_k(\mathbf{y}_f) \Phi^k(\mathbf{x}_0, \mathbf{x}, t). \tag{41}$$

Choosing $\mathbf{z}_f^1, \mathbf{z}_f^2, \mathbf{z}_f^3$ as in Table 4 we obtain the following generalization of (19):

$$\lambda(\mathbf{y}_f) = \Lambda^{-1} \mathbf{y}_f, \qquad \Lambda = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}. \tag{42}$$

The input (41) has been used to simulate reaching movements to $\mathbf{y}_f = [x_P \quad y_P]^\top$.

We compared our results (row 2 and 3 in Fig. 2) with those obtained by (Morasso Morasso 1981) with human subjects (row 1). Simulated and real data have many features in common: (a) reaching tasks are performed following a linear trajectory from the start to the end positions; (b) the tangential velocity profile is single peaked and displays a strong symmetry. These similarities, achieved with the minimum energy and the minimum jerk paradigm, were already observed in the eighties (see Flash and Hogan 1985 and Uno *et al.* 1989). The innovative contribution of our work consists in showing that these paradigms can be obtained as the result of superimposing a reduced number of task-oriented motion primitives.

Simulation of Spinal Fields: We compared our (minimum jerk) motion primitives Φ^k with the force fields induced by microstimulation of the spinal cord in frogs (Giszter et al. 1993). The comparison is obtained representing each spinal field by the vector field of the equivalent force $\mathbf{F}$ at the limb extremity. Specifically, consider a steady configuration $\mathbf{q}_0 = [\theta_1 \quad \theta_2]^\top$, $\dot{\mathbf{q}}_0 = [0, 0]^\top$, $\mathbf{x}_0 = [\mathbf{q}_0 \quad \dot{\mathbf{q}}_0]^\top$ and the associated output: $[x_P \quad y_P]^\top = h(\mathbf{q}_0)$; then the force field associated to Φ^k is (appendix A):

$$\mathbf{F}(x_P, y_P; t) = J(\mathbf{q}_0)^{-\top} \Phi^k(\mathbf{x}_0, \mathbf{x}_0, t). \tag{43}$$

where the expressions of J and Φ^k are given in Table 3 and Table 4. The force field $\mathbf{F}(x_P, y_P; t)$ can be represented in the $(x_P, y_P)-$plane and qualitatively compared with the force field induced by microstimulation of the spinal chord in frogs (see Figure 3).

Simulated and measured fields share many features: (a) they present a single convergence point (black dot in the picture); (b) the orientation of the field is almost constant

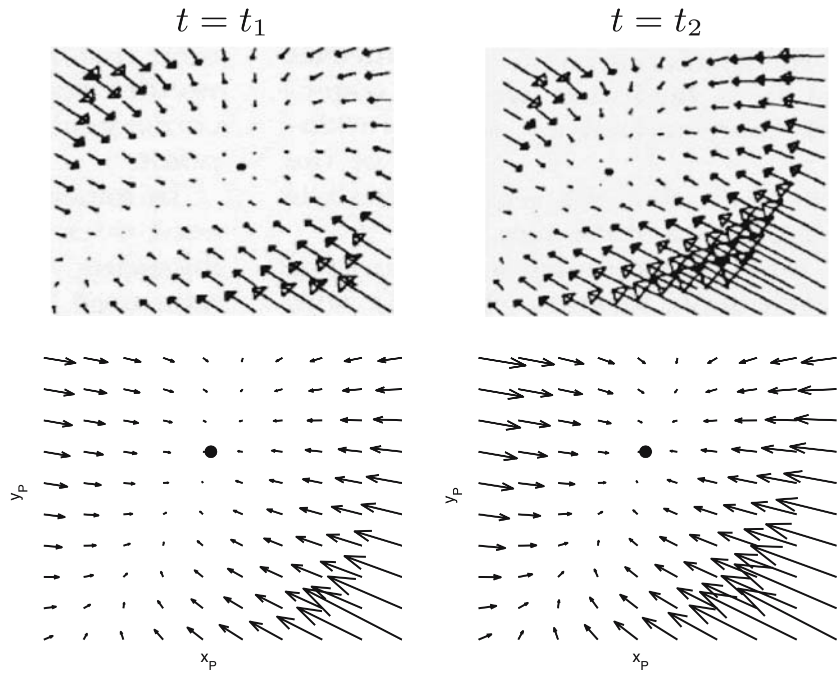

Fig. 3 Temporal evolution of the force field **F** around its convergence point. Top row: field generated by microstimulation of the spinal cord in frogs (from (Giszter et al., 1993)); bottom row: field generated by a synthesized (minimum jerk) motion primitive (see text for details). Left column: fields at $t = t_1$; right column: fields at $t = t_2 > t_1$

(as time goes by) while the magnitude varies: choosing a time-invariant gain G (see section 4.2) the field magnitude increases monotonically; instead, with a time-variant gain (see note 7) the magnitude increases, reaches a peak and then smoothly decreases.

Optimal Positioning of the convergence points: we applied results given in Section 7 to optimally place the convergence points. To simplify the notation, we here assume that the forearm and the upper arm have equal length ($l_1 = l_2 = 0.5$), so that the reachable space is a disk centered in $\mathbf{y} = 0$. Once again, we focus on driving $\mathbf{y}$ to $\mathbf{y}_f$ with zero final velocity. With this simplification, $\mathbf{z}_f^k = [\mathbf{y}_f^k, 0]$ can be represented on a plane. Figure 4 shows an optimal configuration of the convergence points and Table 5 gives their numerical values; any other geometry obtained with an orthogonal transformation would have the same minimum cost. Interestingly, all these solutions are characterized by equally spaced convergence points; a similar feature is observed in microstimulated fields (see Giszter et al. 1993 fig. 9, pag. 477).

9 Motion Primitives for Locomotion

In the previous sections we have shown how the motion primitives can be chosen when the task is defined as a function of the internal state of the system. Specifically, in section 3 the task consists in reaching a given state, while in section 6 a generic function of the state is considered. In these cases, the peculiarity of the decomposition into motion primitives

Table 5 Numerical values for the optimally positioned convergence points

$\mathbf{z}_f^1$	$\frac{1}{\sqrt[4]{3}}\left[0\ -\sqrt{\frac{2}{3}}\ 0\ 0\right]^\top$
$\mathbf{z}_f^2$	$\frac{1}{\sqrt[4]{3}}\left[-\frac{1}{\sqrt{2}}\ \frac{1}{\sqrt{6}}\ 0\ 0\right]^\top$
$\mathbf{z}_f^3$	$\frac{1}{\sqrt[4]{3}}\left[\frac{1}{\sqrt{2}}\ \frac{1}{\sqrt{6}}\ 0\ 0\right]^\top$

is somehow hidden behind the fact that both the tasks and the primitives are a function of the internal state.

A more illustrative situation is presented in this section, where we consider an extension of previous results to the case of locomotion. In this case, the task depends on some *external* variables; in fact, the task is defined with respect to an external reference frame. Primitives, instead, are still a function of the *internal* state of the system as observed by Bizzi. Within this framework, we will formalize the distinction between internal and external variables following the same approach given in (Bloch et al. 1996).

9.1 Internal and External Configuration Variables

Applying the spinal fields paradigm to locomotion requires a formalization of the distinction between internal and external variables. We follow the same approach given in (Bloch et al. 1996) and (Ostrowski 1999), distinguishing the configuration variables of a locomoting robot into two classes. A first set of variables, $\mathbf{g} \in G$, describes the position of the robot in terms

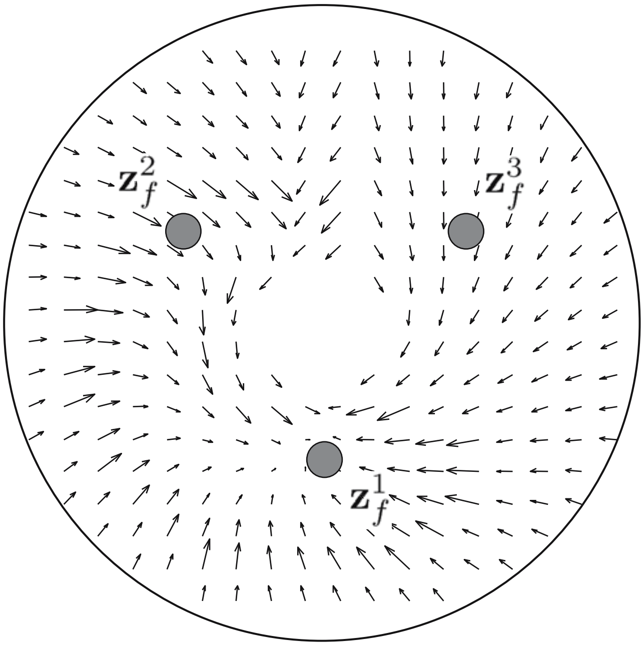

Fig. 4 Optimally positioned convergence points within the output space of a 2DOF manipulator. The points $\mathbf{z}_f^1, \mathbf{z}_f^2, \mathbf{z}_f^3$ are indicated with a dot[3]. The superimposed field corresponds to $\mathbf{z}_f^1$

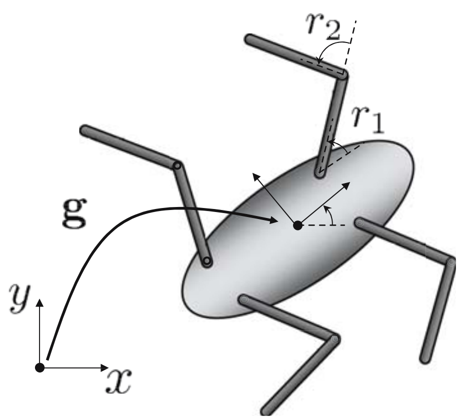

Fig. 5 A locomoting robot

of the displacement between a coordinate frame attached to the robot and an inertial reference frame (Fig. 5). Typically, the set of displacements is chosen to be $SE(m)$ with $m \leq 3$ or one of its subsets. The second class of variables, $\mathbf{r} \in M$, defines the internal configuration, or shape, of the mechanism (Fig. 5); M takes the name of shape space and is required to be a manifold. The total configuration space is therefore $Q = G \times M$.

[3] Notice that some regions of the output space are outside the convex hull formed by $\mathbf{z}_f^1, \mathbf{z}_f^2, \mathbf{z}_f^3$. Points outside the convex hull can be reached only choosing negative combinators. The choice of negative coefficients does not affect stability: positivity of combinators is a sufficient condition (see Note 6) for the stability of the linear combination of stabilizing force fields. However, it is not a necessary condition; therefore, we can insure stability even with negative coefficients.

Let's now try to interpret the experiments by Bizzi within this framework. Here we assume that motion primitives are functions of the internal state only[4]. In the formalization above, the internal state is represented by the shape vector $\mathbf{r}$ and is measured by the so called *proprioceptive* sensors (e.g. muscle spindles). On the other hand, we assume that the combinators are functions of external state only; again, the external state is represented by the vector $\mathbf{g}$ and is measured by the *exteroceptive* sensors (e.g. vision). Consequently, the decomposition into motion primitives corresponds, in our view, to a decomposition at the level of sensors feedback:

$$\mathbf{u} = \sum_{k=1}^{K} \lambda_k(\mathbf{g}) \Phi^k(\mathbf{r}, t). \tag{44}$$

There's no experimental evidence that justifies the choice of combinators that depend only on $\mathbf{g}$, i.e. on exteroceptive sensors. However, in actions such as locomotion, where the task consists in controlling $\mathbf{g}$, this choice is mandatory to achieve robustness. Theoretically, combinators could depend also on other signals but, in our view, this choice is nonintuitive and leads to unnecessary redundancy.

9.2 The Mechanics of Locomotion

The considerations above reveal the importance of separating the dynamics of the internal variables from those of the external ones. Within this framework a fundamental result was proven by (Bloch et al. Bloch et al. 1996) and particularized to robot locomotion in (Ostrowski 1999). Essentially, if the Lagrangian and the nonholonomic constraints of a dynamical system are G-invariant, then the dynamics can be expressed as follows:

$$\mathbf{g}^{-1}\dot{\mathbf{g}} = \xi = -\mathcal{A}(\mathbf{r})\dot{\mathbf{r}} + \tilde{I}^{-1}(\mathbf{r})p, \tag{45a}$$

$$\dot{p} = \frac{1}{2}\dot{\mathbf{r}}^\top \sigma_{\dot{r}\dot{r}}(r)\dot{\mathbf{r}} + p^\top \sigma_{p\dot{r}}(\mathbf{r})\dot{\mathbf{r}} + \frac{1}{2}p^\top \sigma_{pp}(\mathbf{r})p, \tag{45b}$$

$$\tilde{M}(\mathbf{r})\ddot{\mathbf{r}} + \tilde{C}(\mathbf{r}, \dot{\mathbf{r}})\dot{\mathbf{r}} + \tilde{N}(\mathbf{r}, \dot{\mathbf{r}}, p) = \mathbf{u}, \tag{45c}$$

where p is the generalized momentum, ξ is the body representation of the screw velocity, $\mathbf{u}$ is the vector of the forces acting on the shape variables; the definition of the other quantities can be found in (Ostrowski 1999). Interestingly, the above dynamics can be solved separately. In fact, given the temporal evolution of $\mathbf{u}$, the evolutions of $\mathbf{r}$ and p can be obtained integrating (45b) and (45c) from a suitable initial condition. Then, the evolution of the group variable $\mathbf{g}$ can be obtained integrating (45a).

[4] This assumption is compatible with the framework proposed in Mussa-Ivaldi and Bizzi (2000), where motion primitives are modelled as time varying functions of the entire system state $\mathbf{q}$. Specifically, in that framework $\mathbf{q} = \mathbf{r}$ (i.e. the internal state corresponds to the entire system state) and, therefore, primitives are still interpretable as functions of the internal state only.

9.3 Future Works on Locomotion

The spinal field paradigm (44) and the dynamics of locomotion expressed in the form (45) share the common idea of separating internal and external variables; therefore, we believe that the given formulation is ideal for applying the spinal field structure to locomotion; investigating those ideas will be the core of our future works.

Interestingly, the problem of controlling (45) with a modular input structure has already been considered in some recent works (Bullo et al. 1998); (Frazzoli 2004); (Marigo and Bicchi 1998) not directly inspired by the spinal field structure. However, in these works, primitives are concatenated rather than linearly combined. We instead believe that the proposed linear superposition brings to a complexity reduction that we should take advantage of.

10 Conclusions

This paper is inspired by the experimentally observed spinal fields, modelled in terms of motion primitives. The decomposition into motion primitives is appealing from two points of view.

- It simplifies the problem of controlling a complex system, dividing the input $\mathbf{u}$ into two parts: the first, Φ^k, **depending only on the system internal states** (i.e. not depending on the task or on the environment); the second, λ_k, **depending only on the task to be executed**.
- It is a good model for the process through which new actions are learnt. Learning can be thought as the complex problem of choosing an appropriate set of Φ^k for performing a given action. Once the Φ^k have been learned, different instances of the same action are performed choosing different values for the *time-invariant* vector of combinators $\lambda = [\lambda_1, \lambda_2, \ldots, \lambda_K]^\top$.

Taking advantage of these features, we proposed a control paradigm based on motion primitives. A number of theoretical and practical questions have been faced. A first question regards the existence of a set of motion primitives that covers a complete set of tasks; in this sense we have proposed a technique for synthesizing a set of motion primitives that drive the system output to an arbitrary state in an arbitrary amount of time. A second question regards the minimum number of primitives required to perform a given task. We have proven that we need at least p motion primitives if p is the dimension of the task space; moreover, we have shown how to construct a solution with $p+1$ primitives. A third question concerns the problem of adapting a given set of motion primitives to different systems; remarkably, we have found a set of primitives that control a continuum of different kinematic structures which model the growth of an individual. Finally, we have solved the problem of disturbance rejection by optimally placing the convergence points realized by each motion primitive.

A Accordance between the experiment and the model

The two models described in section 2.2.2 and 2.2.3 clearly agree with point (a) in section 2.1. In this appendix we prove that the models agree also with (b). Specifically, we show that the isometric force fields elicited at the limb extremity by the elementary control actions Φ^k, satisfy the superposition principle. Consider a non-redundant kinematic chain and let $\mathbf{F}^k$ be the force field necessary to keep the end-effector (i.e. the limb extremity) in the configuration $\mathbf{q}_0$ when the generalized forces are given by the control action Φ^k. This formalization agrees with the experiments on frogs and rats. Mathematically, the force field $\mathbf{F}^k$ (in absence of gravity and other external forces) is given by (Khatib 1987):

$$\mathbf{F}^k(\mathbf{q}_0, t) = J(\mathbf{q}_0)^{-\top}\Phi^k(\mathbf{x}_0, t), \tag{46}$$

where $\mathbf{x}_0 = [\mathbf{q}_0^\top, 0^\top]^\top$ and $J(\mathbf{q})$ is the Jacobian of the mapping between the configuration space and the end-effector configuration. With this formulation, we observe that the two models (3) and (4) imply the experimentally observed superposition of force fields Giszter et al. (1993). Specifically, the simultaneous activation of the torques $\Phi^1, \ldots, \Phi^K$ leads to a force field $\mathbf{F}$ that is the summation of the fields $\mathbf{F}^k$:

$$\mathbf{F}(\mathbf{q}_0, t) = J(\mathbf{q}_0)^{-\top}\sum_{k=1}^{K}\lambda_k\Phi^k(\mathbf{x}_0, t) = \sum_{k=1}^{K}\lambda_k\mathbf{F}^k(\mathbf{q}_0, t).$$

This is not the case when considering redundant kinematic chains which require different considerations (Mussa-Ivaldi and Hogan 1991), (Gandolfo and Mussa-Ivaldi 1993).

B How to take advantage of the superposition principle

In this appendix we explore the possibility of solving the linear version of problem 1 with the superposition of n controls, each driving the system towards an element of a state space basis. Mathematically, let:

$$\mathbf{v}^k : [0, T] \to \mathbb{R}^m \quad \text{such that} \quad 0 \xrightarrow{\mathbf{v}^k} \mathbf{b}^k, \tag{47}$$

where $k = 1 \ldots n$ and where $\mathbf{b}^1, \ldots, \mathbf{b}^n$ is a basis for the state space, i.e. :

$$\text{Span}(\mathbf{b}^1, \ldots, \mathbf{b}^n) = \mathbb{R}^n.$$

Proposition 10 *With suitable linear combinations of the $K = n$ controls given by (47) the system (8) can be driven from an arbitrary initial state $\mathbf{z}_0$ to an arbitrary final state $\mathbf{z}_f$ during the time interval* $[0, T]$.

Proof We want to show that for suitably chosen combinators $\lambda(\mathbf{z}_0, \mathbf{z}_f)$, the input:

$$\mathbf{v} = \sum_{k=1}^{n}\lambda_k(\mathbf{z}_0, \mathbf{z}_f)\mathbf{v}^k(t) \tag{48}$$

drives the system from $\mathbf{z}_0$ to $\mathbf{z}_f$, i.e. $\mathbf{z}_0 \xrightarrow{\mathbf{v}} \mathbf{z}_f$. It can be proven that under the given assumptions Fornasini (1994):

$$\mathbf{z}_0 \xrightarrow{\mathbf{v}} \mathbf{e}^{AT}\mathbf{z}_0 + \sum_{k=1}^{n}\lambda_k\mathbf{b}^k. \tag{49}$$

Therefore, if we want the system to be driven to $\mathbf{z}_f$, combinators must be chosen so as to satisfy the following:

$$\mathbf{z}_f - \mathbf{e}^{AT}\mathbf{z}_0 = \sum_{k=1}^{n} \lambda_k \mathbf{b}^k. \quad (50)$$

Such combinators always exist since $\mathbf{b}^1, \ldots, \mathbf{b}^n$ is a basis of $\mathbb{R}^n$. □

Using proposition 10, we can preserve the controllability of (8) with an input of the form:

$$\mathbf{v} = \sum_{k=1}^{n} \lambda_k(\mathbf{z}_0, \mathbf{z}_f)\mathbf{v}^k(t). \quad (51)$$

However, the structure (51) does not solve problem 1. In fact, the proposed $\mathbf{v}^k$ cannot be motion primitives according to the given definition. Specifically, both the elementary inputs $\mathbf{v}^k$ and their linear combinations, drive the system towards a state which depends on the initial state[5]. Therefore, the proposed $\mathbf{v}^k$ do not fulfill either definition 1 or what required by the synthesis problem. This fact reflects in that the combinators depend on both $\mathbf{z}_0$ and $\mathbf{z}_f$ while they should not depend on the former (compare (9) and (51)).

C Proof of Proposition 3

Proof The state trajectory $\mathbf{z}(t)$ followed by the linear system (8) in response to the open-loop input $\mathbf{v} : [0, T] \to \mathbb{R}^m$ when $\mathbf{z}(0) = \mathbf{z}_0$ is given by:

$$\mathbf{z}(t) = \mathbf{e}^{At}\mathbf{z}_0 + \int_0^t \mathbf{e}^{A(t-\tau)} B\mathbf{v}(\tau)\mathrm{d}\tau. \quad (52)$$

If the input is a feedforward motion primitive $\zeta^k(\mathbf{z}_0, t)$, we get:

$$\mathbf{z}^k(\mathbf{z}_0, t) = \mathbf{e}^{At}\mathbf{z}_0 + I_k(\mathbf{z}_0, t), \quad (53)$$

where:

$$I_k(\mathbf{z}_0, t) = \int_0^t \mathbf{e}^{A(t-\tau)} B\zeta^k(\mathbf{z}_0, \tau)\mathrm{d}\tau. \quad (54)$$

In particular, if the convergence point of the given primitive is $\mathbf{z}_f^k$ we have that:

$$\mathbf{z}(\mathbf{z}_0, T) = \mathbf{e}^{AT}\mathbf{z}_0 + I_k(\mathbf{z}_0, T) = \mathbf{z}_f^k, \quad \forall \mathbf{z}_0, \quad (55)$$

so that we have:

$$I_k(\mathbf{z}_0, T) = \mathbf{z}_f^k - \mathbf{e}^{AT}\mathbf{z}_0, \quad \forall \mathbf{z}_0. \quad (56)$$

Using the linearity of integration, we get that the final state in response to $\lambda_1\zeta^1 + \lambda_2\zeta^2$ is given by:

$$\mathbf{z}(T) = \mathbf{e}^{AT}\mathbf{z}_0 + \lambda_1 I_1(\mathbf{z}_0, T) + \lambda_2 I_2(\mathbf{z}_0, T) \quad (57)$$
$$= \lambda_1\mathbf{z}_f^1 + \lambda_2\mathbf{z}_f^2 + (1 - \lambda_1 - \lambda_2)\mathbf{e}^{AT}\mathbf{z}_0 \quad (58)$$

Using the invertibility of the exponential matrix, we conclude that the final state in response to $\lambda_1\zeta^1 + \lambda_2\zeta^2$ does not depend on the initial condition if and only if $\lambda_1 + \lambda_2 = 1$. If this is the case, $\mathbf{z}(T) = \lambda_1\mathbf{z}_f^1 + \lambda_2\mathbf{z}_f^2$ and this proves (13). Moreover, the trajectory $\mathbf{z}$ in response to $\lambda_1\zeta^1 + \lambda_2\zeta^2$ is:

$$\mathbf{z}(\mathbf{z}_0, t) = \mathbf{e}^{AT}\mathbf{z}_0 + \lambda_1 I_1(\mathbf{z}_0, t) + \lambda_2 I_2(\mathbf{z}_0, t) \quad (59)$$
$$= \lambda_1(\mathbf{e}^{At}\mathbf{z}_0 + I_1(\mathbf{z}_0, t))$$
$$+\lambda_2(\mathbf{e}^{At}\mathbf{z}_0 + I_2(\mathbf{z}_0, t)) \quad (60)$$
$$= \lambda_1\mathbf{z}^1(\mathbf{z}_0, t) + \lambda_2\mathbf{z}^2(\mathbf{z}_0, t) \quad (61)$$

where $\mathbf{z}^1$ and $\mathbf{z}^2$ are the trajectories in response to ζ^1 and ζ^2 alone. This proves (14) and concludes the proof. □

D Proof of Proposition 6

Proof Given the particular linear system (double integrator) that we have obtained with the feedback linearization, we can easily modify the results of Section 5 to obtain an execution in arbitrary time T for the system (29). The idea is that an execution α-times faster can be obtained with an α-times faster input scaled by α^2. Specifically, if the input $\mathbf{u}(t)$ $t \in [0, T]$ drives the system trough the trajectory $\mathbf{q}(t)$ from the initial condition $\left[\begin{smallmatrix}\mathbf{q}(0)\\ \dot{\mathbf{q}}(0)\end{smallmatrix}\right] = \left[\begin{smallmatrix}\mathbf{q}_i\\ \dot{\mathbf{q}}_i\end{smallmatrix}\right]$, then $\alpha^2\mathbf{u}(\alpha t)$, $t \in [0, \frac{T}{\alpha}]$ drives the system trough the trajectory $\mathbf{q}(\alpha t)$ from the initial condition $\left[\begin{smallmatrix}\mathbf{q}(0)\\ \dot{\mathbf{q}}(0)\end{smallmatrix}\right] = \left[\begin{smallmatrix}\mathbf{q}_i\\ \alpha\dot{\mathbf{q}}_i\end{smallmatrix}\right]$. This simple observation suggests a way for finding $\mathbf{v}$ in (29) so as to drive the system from $\left[\begin{smallmatrix}\mathbf{q}_i\\ \dot{\mathbf{q}}_i\end{smallmatrix}\right]$ to $\left[\begin{smallmatrix}\mathbf{q}_f\\ \dot{\mathbf{q}}_f\end{smallmatrix}\right]$ in the time interval $[0, \frac{T}{\alpha}]$. Let's apply this idea to (22). We can write:

$$\mathbf{v} = H^1(t)\mathbf{z}_0 + H^2(t)\mathbf{z}_f$$
$$= \left[H^1_{\mathbf{q}}(t)\ H^1_{\dot{\mathbf{q}}}(t)\right]\begin{bmatrix}\mathbf{q}_0\\ \dot{\mathbf{q}}_0\end{bmatrix} + \left[H^2_{\mathbf{q}}(t)\ H^2_{\dot{\mathbf{q}}}(t)\right]\begin{bmatrix}\mathbf{q}_f\\ \dot{\mathbf{q}}_f\end{bmatrix},$$

where we used $\mathbf{z} = \mathbf{x} = [\mathbf{q}^\top, \dot{\mathbf{q}}^\top]^\top$, $\mathbf{z}_f = [\mathbf{q}_f^\top, \dot{\mathbf{q}}_f^\top]^\top$ and $\mathbf{z}_0 = [\mathbf{q}_0^\top, \dot{\mathbf{q}}_0^\top]^\top$. Using the observation above, we can easily prove that the input:

$$\mathbf{v} = \left[\alpha^2 H^1_{\mathbf{q}}(\alpha t)\ \alpha H^1_{\dot{\mathbf{q}}}(\alpha t)\right]\begin{bmatrix}\mathbf{q}_0\\ \dot{\mathbf{q}}_0\end{bmatrix}$$
$$+ \left[\alpha^2 H^2_{\mathbf{q}}(\alpha t)\ \alpha H^2_{\dot{\mathbf{q}}}(\alpha t)\right]\begin{bmatrix}\mathbf{q}_f\\ \dot{\mathbf{q}}_f\end{bmatrix}, \quad (62)$$

drive the system (29) from $\left[\begin{smallmatrix}\mathbf{q}_i\\ \dot{\mathbf{q}}_i\end{smallmatrix}\right]$ to $\left[\begin{smallmatrix}\mathbf{q}_f\\ \dot{\mathbf{q}}_f\end{smallmatrix}\right]$ in the time interval $[0, \frac{T}{\alpha}]$. Moreover, it can be proven that such $\mathbf{v}$ can be chosen optimal for a suitably defined fixed final state optimal control problem (see Nori and Frezza 2004a). Now, using (62), we can obtain a set of motion primitives ϕ^k that solves the

[5] This is a direct consequence of having chosen controls $\mathbf{v}^k$ which are *independent of the system state*. Therefore, the application of the same $\mathbf{v}^k$ to different initial conditions leads to different final states. To get a control action capable of driving the system towards the same convergence point (regardless of the initial condition) we need to resort to control actions which *depend on the system state*. In the light of these observations, Section 4.1 considers controls which depend on the initial state (feedforward motion primitives) while Section 4.2 generalizes to feedback on the present state.

reaching task in time $\frac{T}{\alpha}$. Specifically, we can give to the input $\mathbf{v}$ (of the linearized system) the same structure of $\mathbf{u}$ in (30):

$$\mathbf{v} = \sum_{k=1}^{K} \lambda_k(\mathbf{z}_f, \alpha)\phi^k(\mathbf{z}_0, \mathbf{z}, \alpha t). \tag{63}$$

This structure can be obtained defining:

$$\phi^k(\mathbf{z}_0, \mathbf{z}, t) = \left[\alpha_k^2 H^1_{\mathbf{q}}(t)\ \alpha_k H^1_{\dot{\mathbf{q}}}(t)\right]\mathbf{z}_0 + \left[\alpha_k^2 H^2_{\mathbf{q}}(t)\alpha_k H^2_{\dot{\mathbf{q}}}(t)\right]\mathbf{z}_f^k \quad k = 1\dots n+2.$$

Combinators are chosen as follows:

$$\begin{bmatrix} \lambda_1 \\ \vdots \\ \lambda_{n+2} \end{bmatrix} = \begin{bmatrix} \alpha_1^2 & \dots & \alpha_{n+2}^2 \\ \alpha_1 & \dots & \alpha_{n+2} \\ \alpha_1^2\mathbf{q}_f^1 & \dots & \alpha_{n+2}^2\mathbf{q}_f^{n+2} \\ \alpha_1\dot{\mathbf{q}}_f^1 & \dots & \alpha_{n+2}\dot{\mathbf{q}}_f^{n+2} \end{bmatrix}^{-1} \begin{bmatrix} \alpha^2 \\ \alpha \\ \alpha^2\mathbf{q}_f \\ \alpha\dot{\mathbf{q}}_f \end{bmatrix},$$

where $\left[\begin{smallmatrix}\mathbf{q}_f \\ \dot{\mathbf{q}}_f\end{smallmatrix}\right] = \mathbf{z}_f$, $\left[\begin{smallmatrix}\mathbf{q}_f^k \\ \dot{\mathbf{q}}_f^k\end{smallmatrix}\right] = \mathbf{z}_f^k$ and where the given matrix should be invertible. A possible choice is:

$$\alpha_1 = \dots = \alpha_n = 1, \quad \alpha_{n+1} = 1, \quad \alpha_{n+2} = 2,$$
$$\mathbf{q}_f^{n+1} = \dot{\mathbf{q}}^{n+1} = \mathbf{q}_f^{n+2} = \dot{\mathbf{q}}_f^{n+2} = 0,$$

and $\left[\begin{smallmatrix}\mathbf{q}_f^1 \\ \dot{\mathbf{q}}_f^1\end{smallmatrix}\right], \dots, \left[\begin{smallmatrix}\mathbf{q}_f^n \\ \dot{\mathbf{q}}_f^n\end{smallmatrix}\right]$ could be any basis for $\mathbb{R}^n$. Notice that $\phi^k(\alpha_k t)$ drives the system to $\left[\begin{smallmatrix}\mathbf{q}_f^k \\ \dot{\mathbf{q}}_f^k\end{smallmatrix}\right]$ in the time interval $[0, \frac{T}{\alpha_k}]$. Thus ϕ^k can still be considered motion primitives. We are left with modifying the primitives ϕ^k for $\mathbf{v}$ to obtain the primitives Φ^k for $\mathbf{u}$. This final step consists in adding one primitive so as to force the sum of the combinators to be one. The procedure in section 5.1 can then be applied defining:

$$\Phi^k(\mathbf{x}_0, \mathbf{x}, t) = M(\mathbf{q})\phi^k(\mathbf{z}_0, \mathbf{z}, t) + C(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + N(\mathbf{q}, \dot{\mathbf{q}}).$$

□

E Proof of Proposition 7

Proof As a first step, let's try to understand how the matrices M, C and N of the dynamic model (1) change with the vector of parameters $\mathbf{p}$. To simplify the notation, let's define a couple of vectors that depend on $\mathbf{p}$; the first vector ψ contains all elements of the type $m_i q_k$ where q_k is either an element of $\mathbf{l}^k$ or an element of $\mathbf{c}^k$; the second vector Ψ contains elements like $m_i q_k q_j$ or I_j^i. Formally:

$$\psi = [m_1\varphi \dots m_m\varphi]^\top \in \mathbb{R}^{n_\psi}, \quad n_\psi = O(m^2),$$
$$\Psi = \left[\mathbf{r}^{1\top} \dots \mathbf{r}^{m\top}\right]^\top \in \mathbb{R}^{n_\Psi}, \quad n_\Psi = O(m^3),$$

where:

$$\varphi = \left[\mathbf{l}^{1\top} \dots \mathbf{l}^{m\top}\mathbf{c}^{1\top} \dots \mathbf{c}^{m\top}\right]^\top \in \mathbb{R}^{n_\varphi},$$
$$\mathbf{r}^i = \left[\mathbf{s}^{i,1\top} \dots \mathbf{s}^{i,n_\varphi\top} I_{i,1} \dots I_{i,6}\right]^\top,$$
$$\mathbf{s}^{i,j} = \left[m_i\varphi_j^2 m_i\varphi_j\varphi_{j+1} \dots m_i\varphi_j\varphi_{n_\varphi}\right]^\top.$$

In the previous formulas, φ_j indicates the j^{th} element of the vector φ, while $I_1^i, \dots, I_6^i$ are the entries of the inertia tensor:

$$\mathcal{I}_i = \begin{bmatrix} I_1^i & I_2^i & I_3^i \\ I_2^i & I_4^i & I_5^i \\ I_3^i & I_5^i & I_6^i \end{bmatrix}.$$

It can be proven (Sciavicco and Siciliano 2000) that the matrices M and C are linearly parameterized by the elements of the vector Ψ. A similar structure (except for an affine term) holds for the matrix N when assuming that the only external forces are gravity and constant frictions at the joints. Specifically, we have:

$$M(\mathbf{q}) = \sum_{i=1}^{n_\Psi} \Psi_i M^i(\mathbf{q}), \tag{64a}$$

$$C(\mathbf{q}, \dot{\mathbf{q}}) = \sum_{i=1}^{n_\Psi} \Psi_i C^i(\mathbf{q}, \dot{\mathbf{q}}), \tag{64b}$$

$$N(\mathbf{q}, \dot{\mathbf{q}}) = \sum_{j=1}^{n_\psi} \psi_j N^j(\mathbf{q}, \dot{\mathbf{q}}) + N^0(\mathbf{q}, \dot{\mathbf{q}}), \tag{64c}$$

for suitably defined matrices M^i, C^i and N^j that do not depend on $\mathbf{l}_i$, $\mathbf{c}_i$, m_i and $\mathcal{I}_i$. Exploiting this structure, we will show how to obtain primitives Φ^k that solve Problem 1 and do not depend on the parameters, i.e.:

$$\mathbf{u} = \sum_{h,k} \lambda_k(\mathbf{x}_f)\mu_h(\psi, \Psi)\Phi^{k,h}(\mathbf{x}_0, \mathbf{x}, t). \tag{65}$$

Notice that this structure corresponds to (32) since $\mathbf{x} = [\mathbf{q}^\top, \dot{\mathbf{q}}^\top]^\top$ and $\psi(\mathbf{p})$, $\Psi(\mathbf{p})$. This decomposition possesses an interesting property: changes of the parameters Ψ and ψ (and thus of $\mathbf{p}$) are accommodated modifying the combinators λ_k, while the motion primitives $\Phi^{k,h}$ are left unchanged. Practically, such a decomposition can be obtained using the feedback linearizing equation (28) and one of the proposed decompositions for linear systems such as (26). Specifically, we have:

$$\mathbf{u} = M(\mathbf{q})\left[\sum_{k=1}^{n+1} \lambda_k(\mathbf{z}_f)\phi^k(\mathbf{z}_0, \mathbf{z}, t)\right] + C(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + N(\mathbf{q}, \dot{\mathbf{q}}).$$

where as before $\mathbf{z} = \mathbf{x} = [\mathbf{q}^\top, \dot{\mathbf{q}}^\top]^\top$. Then, using (64) we obtain:

$$\mathbf{u} = \sum_{i=1}^{n_\Psi} \Psi_i M^i(\mathbf{q})\left[\sum_{k=1}^{n+1} \lambda_k(\mathbf{z}_f)\phi^k(\mathbf{x}_0, \mathbf{x}, t)\right] + \left[N^0(\mathbf{q}, \dot{\mathbf{q}}) + \sum_{i=1}^{n_\Psi} \Psi_i C^i(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \sum_{j=1}^{n_\psi} \psi_j N^j(\mathbf{q}, \dot{\mathbf{q}})\right].$$

Using the fact that λ_k sum up to one and $\mathbf{z}_f = \mathbf{x}_f$ we get:

$$\mathbf{u} = \sum_{k=1}^{n+1} \lambda_k(\mathbf{x}_f) \left[\sum_{i=1}^{n_\Psi} \Psi_i M^i(\mathbf{q}) \phi^k(\mathbf{x}_0, \mathbf{x}, t) + \sum_{i=1}^{n_\Psi} \Psi_i C^i(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \sum_{j=1}^{n_\psi} \psi_j N^j(\mathbf{q}, \dot{\mathbf{q}}) + N^0(\mathbf{q}, \dot{\mathbf{q}}) \right].$$

With easy considerations, the term within the square parenthesis can be reformulated as a linear sum with combinators depending only on Ψ and ψ. We get:

$$\mathbf{u} = \sum_{k=1}^{n+1} \lambda_k(\mathbf{x}_f) \sum_{h=1}^{H} \mu_h(\Psi, \psi) \Phi^{k,h}(\mathbf{x}_0, \mathbf{x}, t) = \sum_{k,h} \lambda_k(\mathbf{x}_f) \mu_h(\Psi, \psi) \Phi^{k,h}(\mathbf{x}_0, \mathbf{x}, t),$$

for a suitably chosen scalar H and a suitably defined function $\mu_h(\Psi, \psi)$; the function $\Phi^{k,h}$ can be chosen to be a motion primitive for the system with parameters Ψ^h, ψ^h, i.e.:

$$\Phi^{k,h}(\mathbf{x}, t) = \sum_{i=1}^{n_\Psi} \Psi_i^h M^i(\mathbf{q}) \phi^k(\mathbf{x}_0, \mathbf{x}, t) + \sum_{i=1}^{n_\Psi} \Psi_i^h C^i(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \sum_{j=1}^{n_\psi} \psi_j^h N^j(\mathbf{q}, \dot{\mathbf{q}}) + N^0(\mathbf{q}, \dot{\mathbf{q}}).$$

Therefore, $\Phi^{k,h}$ drives the system with parameters Ψ^h, ψ^h to the state $\mathbf{x}_f^k = \mathbf{z}_f^k$. Moreover, it can be proven that Ψ^h, ψ^h can always be chosen in such a way that there exists a $\mathbf{p}^h$ such that $\psi(\mathbf{p}^h) = \psi^h$ and $\Psi(\mathbf{p}^h) = \Psi^h$. This conclude the proof.

Notice that, even if the number of the primitives was noticeably increased (from $O(n)$ to $O(n^4)$), the control paradigm has gained in terms of flexibility. Many different kinematic structures can be controlled to the desired configuration with simple modifications of the function that computes the time invariant combinators. □

F Proof of Proposition 9

Proof Let's first observe that the random vector λ is independent of the errors $\Delta\mathbf{z}_f^k$; this property is in fact inherited from $\mathbf{z}_f$, assumed independent of $\Delta\mathbf{z}_f^k$. Using these independencies together with (37) it can be easily proven that the error $\mathbf{e}$ is zero mean, i.e. $E[\mathbf{e}] = 0$. Considering now the variance of $\mathbf{e}$, we have:

$$\begin{aligned} E[\mathbf{e}\mathbf{e}^\top] &= E\left\{ \left[\sum_k \lambda_k \Delta\mathbf{z}_f^k \right] \left[\sum_j \lambda_j \Delta\mathbf{z}_f^j \right]^\top \right\} \\ &= \sum_{k,j} E\left[\lambda_k \lambda_j \Delta\mathbf{z}_f^k \Delta\mathbf{z}_f^{j\top} \right] \\ &= \sum_{k,j} E\left[\lambda_k \lambda_j \right] E\left[\Delta\mathbf{z}_f^k \Delta\mathbf{z}_f^{j\top} \right] \\ &= \sum_k E\left[\lambda_k^2 \right] E\left[\Delta\mathbf{z}_f^k \Delta\mathbf{z}_f^{k\top} \right] \\ &= \Sigma_\Delta E[\lambda^\top \lambda], \end{aligned}$$

where we used the independence relations between λ_k and $\Delta\mathbf{z}_f^k$.

Now, we are left with choosing $\mathbf{z}_f^k$ so as to minimize the variance of $\mathbf{e}$ under the constraints $|\det(\Lambda)| = 1$; to simplify the proof we assume $\det(\Lambda) = 1$. First of all, we observe that minimizing the variance matrix is equivalent to minimizing $E[\lambda^\top\lambda] = E[\lambda_1^2 + \cdots + \lambda_{n+1}^2]$. We have:

$$E[\lambda^\top\lambda] = c \int_{S_0^1} \left[\lambda_1^2(\mathbf{z}_f) + \cdots + \lambda_{n+1}^2(\mathbf{z}_f) \right] \mathrm{d}\mathbf{z}_f,$$

where c is a suitably defined normalization constant and S_0^1 is the sphere centered in the origin. Rearranging (19) and completing the squares, the above integral can be written:

$$\int_{S_0^1} \left\{ \left[\hat{\lambda}(\mathbf{z}_f) - \lambda_0 \right]^\top M \left[\hat{\lambda}(\mathbf{z}_f) - \lambda_0 \right] + d \right\} \mathrm{d}\mathbf{z}_f,$$

where:

$$\hat{\lambda}(\mathbf{z}_f) = \hat{\Lambda}^{-1} \left(\mathbf{z}_f - \mathbf{z}_f^{n+1} \right), \qquad \lambda_0 = M^{-1}\mathbf{v},$$
$$\hat{\Lambda} = \mathbf{z}_f^1 - \mathbf{z}_f^{n+1} \dots \mathbf{z}_f^n - \mathbf{z}_f^{n+1}, \quad d = 1 - \mathbf{v}^\top M^{-1} \mathbf{v}.$$

Therefore, we can write the integrand explicitly as a function of $\mathbf{z}_f$:

$$\int_{S_0^1} \left[\left(\mathbf{z}_f - \mathbf{z}_c \right)^\top \hat{\Lambda}^{-\top} M \hat{\Lambda}^{-1} \left(\mathbf{z}_f - \mathbf{z}_c \right) + d \right] \mathrm{d}\mathbf{z}_f,$$

where we have defined $\mathbf{z}_c = \mathbf{z}_f^{n+1} + \hat{\Lambda}\lambda_0$.

Going back to the minimization problem, let's introduce a change of variables; define the new optimization variables to be $\mathbf{z}_c$ and D, with $D = \hat{\Lambda}^{-\top} M \hat{\Lambda}^{-1}$; it can be proven that $M = M^\top > 0$ so that $D = D^\top > 0$. Details on how to retrieve the original variables from $\mathbf{z}_c$ and D will be given at the end of the proof. For purpose of simplicity, let's assume D diagonal; this is not restrictive since one can easily prove that a given minimum can be transformed, using a suitable orthogonal matrix, into an equivalent minimum with D diagonal.

Ignoring the constants that do not play a role in the minimization, we are left with the following minimization problem:

$$\min_{D=D^\top>0} \min_{\mathbf{z}_c} \int_{S_0^1} \left[(\mathbf{z} - \mathbf{z}_c)^\top D \, (\mathbf{z} - \mathbf{z}_c) \right] \mathrm{d}\mathbf{z},$$

It can be proven that $\det(\Lambda) = \det(\hat{\Lambda})$ and $\det(M) = 1 + n$ so that the constraint $\det(\Lambda) = 1$ reflects into the constraints $\det(D) = n + 1$. The minimization with respect to $\mathbf{z}_c$ is obtained with $\mathbf{z}_c^* = 0$. This can be proven observing that the above integral equals:

$$\int_{S_0^1} \mathbf{z}^\top D \mathbf{z} \, \mathrm{d}\mathbf{z} + \int_{S_0^1} \mathbf{z}_c^\top D \mathbf{z}_c \, \mathrm{d}\mathbf{z},$$

since:

$$\int_{S_0^1} \mathbf{z}_c^\top D\mathbf{z}\, \mathrm{d}\mathbf{z} = \int_{S_0^1} \mathbf{z}^\top D\mathbf{z}_c\, \mathrm{d}\mathbf{z} = 0 \qquad \forall \mathbf{z}_c.$$

Consequently, it remains to solve the following optimization:

$$\min_{D=D^\top>0} \int_{S_0^1} \mathbf{z}^\top D\mathbf{z}\, \mathrm{d}\mathbf{z} \quad \text{s.t.} \quad \det(D) = n+1.$$

Writing $D = \mathrm{diag}(d_1^2, \ldots, d_n^2)$ we reduce the problem as follows:

$$\min_{d_1,\ldots,d_n} \int_{S_0^1} d_1^2 z_1^2 + \ldots + d_n^2 z_n^2\, \mathrm{d}\mathbf{z} \quad \text{s.t.} \quad d_1^2 \cdot \ldots \cdot d_n^2 = n+1.$$

Observing that:

$$\int_{S_0^1} d_1^2 z_1^2 \mathrm{d}\mathbf{z} = \ldots = \int_{S_0^1} d_n^2 z_n^2\, \mathrm{d}\mathbf{z} = \text{cost}.$$

we finally get:

$$\min_{d_1,\ldots,d_n} \left(d_1^2 + \ldots + d_n^2\right) \quad \text{s.t.} \quad d_1^2 \cdot \ldots \cdot d_n^2 = n+1,$$

whose solution is $d_1^2 = \ldots = d_n^2 = \sqrt[n]{n+1}$. Therefore $D^* = \mathrm{diag}(\sqrt[n]{n+1}, \ldots, \sqrt[n]{n+1})$. To recover the original optimization variables let's write $M = L^\top L$ (L square) so that from $D^* = \hat{\Lambda}^{-\top} M \hat{\Lambda}^{-1}$ we get:

$$D^* = (L\hat{\Lambda}^{-1})^\top L\hat{\Lambda}^{-1}.$$

We are left with a standard problem of retrieving all the square factorizations of a positive definite matrix. The solution is known to be be the product QN, where Q is an arbitrary orthogonal matrix and N is a particular solution (i.e. $D^* = NN^\top$). Therefore:

$$L\hat{\Lambda}^{-1} = QN \quad \Rightarrow \quad \hat{\Lambda} = N^{-1}QL.$$

Choosing $N = \mathrm{diag}(\sqrt[2n]{n+1}, \ldots, \sqrt[2n]{n+1})$ we get:

$$\hat{\Lambda} = \frac{1}{\sqrt[2n]{n+1}} QL.$$

Theoretically we should multiply the matrix $\hat{\Lambda}$ by another orthogonal matrix in order to account for having assumed D diagonal; however, this multiplication can be omitted since the product of orthogonal matrices is again an orthogonal matrix. Let's now construct $\mathbf{z}_f^1, \ldots, \mathbf{z}_f^{n+1}$. From $\mathbf{z}_c^* = \mathbf{z}_f^{n+1} + \hat{\Lambda}\lambda_0 = 0$ we get:

$$\mathbf{z}_f^{n+1} = -\hat{\Lambda}\lambda_0 = -\frac{1}{\sqrt[2n]{n+1}} QL^{-\top}\mathbf{v},$$

where we used the definition $\lambda_0 = M^{-1}\mathbf{v} = L^{-1}L^{-\top}\mathbf{v}$. Rearranging the definition of $\hat{\Lambda}$ we get:

$$\mathbf{z}_f^k = \hat{\Lambda}\mathbf{v}^k + \mathbf{z}_f^{n+1} = \frac{1}{\sqrt[2n]{n+1}} QL^{-\top}(M\mathbf{v}^k - \mathbf{v}),$$

for $k = 1, \ldots, n$. When Q spans the entire set of the orthogonal matrices we get all the minimum variance solutions; a particular solution is then obtained choosing $Q = I$ as in (39). □

References

Bienenstock E, Geman S (1993) Compositionality in neural systems. Arbib MA (ed), The handbook of brain theory and neural networks. MIT Press, Cambridge pp 223–226

Bloch AM, Krishnaprasad PS, Marsden JE, Murray RM (1996) Nonholonomic mechanical systems with symmetry. Arch Ration Mech Anal 136(1):21–99

Boothby William M (2002) An introduction to differentiable manifolds and riemannian geometry. 2nd edn. Academic Press, New York

Bullo F, Leonard NE, Lewis, AD (1998) Controllability and motion algorithms for underactuated lagrangian systems on lie groups. IEEE Tran Automat Control 45(8):1437–1454

Del Vecchio D, Murray, RM, Perona P (2003) Decomposition of human motion into dynamics-based primitives with application to drawing tasks. Automatica 39:2085–2098

Fornasini E, Marchesini G (1994) Appunti di Teoria dei Sistemi. Edizioni Libreria Progetto. In Italian

Flash T, Hogan N (1985) The Coordination of Arm Movements: An Experimentally Confirmed Mathematical Model. J of Neurosci 5:1688–1703

Frazzoli E (2004) Maneuver-based motion planning for nonlinear systems with symmetries. Submitted to IEEE Trans Rob Autom

Gandolfo F, Mussa-Ivaldi, FA (1993) Vector summation of end-point impedance in kinematically redundant manipulators. of: Proceedings of IEEE/RJS International Conference of Intelligent Robots and Systems. IROS93, pp 1627–1634

Giszter SF, Mussa-Ivaldi FA, Bizzi E (1993) Convergent force fields organized in the frog's spinal cord. J Neurosci 13(2):467–491

Hogan N (1985a) Impedance control: An approach to manipulation: part I - theory, part II - implementation, part III - applications. Trans ASME J Dyn Syst Meas Control 107:1–24

Hogan N (1985b) The mechanics of multi-joint posture and movement-control. Biol Cybern 52:315–331

Isidori A (1995) Nonlinear control systems 3rd edn. Springer, Berlin Heidelberg New York

Khatib O (1987) A unified approach for motion and force control of robot manipulators: The Operational Space Formulation. IEEE J Rob Autom RA-3(1):43–53

Lewis FL, Syrmos VL (1995) Optimal control. Wiley, New York

Marigo A, Bicchi A (1998) Steering driftless nonholonomic systems by control quanta. In: Proceedings of 1998 conference on decision and control

Morasso P (1981) Spatial control of arm movements. Exp Brain Res 42(2):223–7

Murray RM, Li Z, Sastry SS (1994) A mathematical introduction to robotic manipulation. CRC, USA

Mussa-Ivaldi FA (1997) Nonlinear force fields: a distributed system of control Promitives for representing and learning movements. In: CIRA '97, Montrey California, USA

Mussa-Ivaldi FA (1999) Modular features of motor control and learning. Curr Opin Neurobiol 9:713–717

Mussa-Ivaldi FA, Bizzi E (2000) Motor learning through the combination of primitives. Philos Trans Royal Soc Biol Sci 355:1755–1769

Mussa-Ivaldi FA, Hogan N (1991) Integrable solutions of kinematic redundancy via impedance control. Int J Rob Res 10(5):481–491

Mussa-Ivaldi FA, Giszter SF, Bizzi E (1994) Linear combinations of primitives in vertebrate motor control. Proc Natl Acad Sci USA 91:7534–7538

Mussa-Ivaldi FA, Giszter SF (1992) Vector field approximation: a computational paradigm for motor control and learning. Biol Cybern 37:491–500

Nori F, Frezza R (2004a) Biologically inspired control of a kinematic chain using the superposition of motion primitives. In: Proceedings of 2004 conference on decision and control

Nori F, Frezza R (2004b) Nonlinear control by a finite set of motion primitives. In: Proceedings of 2004 nonlinear control systems design. IFAC

Ostrowski J (1999) Computing reduced equations for robotic systems with constraints and symmetries. IEEE Trans Rob Autom 15(1): 111–123

Poggio T, Smale S (2003) The mathematics of learning: dealing with data. Not Am Math Soc 50(5):537–544

Samson C, Espiau Bs, Borgne M.Le 1991. Robot control: the task function approach. Oxford University Press, New York

Sanger TD (2000) Human arm movements described by a low-dimensional superposition of principal components. J Neurosci 20(3):1066–1072

Sciavicco L, Siciliano B (2000) Modelling and control of robot manipulators. 2nd edn. Springer, Berlin, Heidelberg New York

Shadmehr R, Mussa-Ivaldi FA (1994) Adaptive representation of dynamics during learning of a motor task. J Neurosci 5(14):3208–3224

Slotine JE, Li W (1991) Applied nonlinear control. Prentice-Hall, Englewood cliffs, USA

Slotine JJ (2003) Modular stability tools for distributed computation and control. Int J Adaptive Control Signal Process 17(6):397–416

Todorov E, Jordan M (2002) Optimal feedback control as a theory of motor coordination. Nat Neurosci 5(11):1226–1235

Uno Y, Kawato M, Suzuki R (1989) Formation and control of optimal trajectory in human multijoint arm movement. Biol Cybern 61(2):89–101