Cognitive Robotics 1 (2021) 92–110

KeAi
CHINESE ROOTS
GLOBAL IMPACT

Contents lists available at ScienceDirect

Cognitive Robotics

journal homepage: http://www.keaipublishing.com/en/journals/cognitive-robotics/

Gesture formation: A crucial building block for cognitive-based Human–Robot Partnership

Pietro Morasso

Italian Institute of Technology, Center for Human Technologies, RBCS (Robotics, Brain, and Cognitive Sciences) Department, Via Enrico Melen 83, Bldg B, 16152 Genova, Italy

ARTICLE INFO

Keywords:
Embodied Cognition
Human-Robot Partnership
Trajectory Formation
Motor Intention

ABSTRACT

The next generation of robotic agents, to employed both in industrial and service robotic applications, will be characterized by a high degree of Human–Robot Partnership that implies, for example, sharing common objectives, bidirectional flow of information, capability to learn from each other, and availability to mutual training. Moreover, there is a widespread feeling in the research community that probably Humans will not accept Robots as trustable Partners if they cannot ascribe some form of awareness and true understanding to them. This means that, in addition to the incremental improvements of *Robotic-Bodyware,* there will be the need for a substantial jump of the *Robotic-Cogniware*, namely a new class of Cognitive Architectures for Robots (CARs) that match the requirements and specific constraints of Human–Robot Partnership. The working hypothesis that underlies this paper is that such class of CARs must be bio-inspired, not in the sense of fine-grain imitation of neurobiology but the large framework of embodied cognition. In our opinion, trajectory/gesture formation should be one of the building blocks of bio-inspired CARs because biological motion is a fundamental channel of inter-human partnership, a true body language that allows mutual understanding of intentions. Moreover, one of the main concepts of embodied cognition, related to the importance of motor imagery, is that real (or *overt*) actions and mental (or *covert*) actions are generated by the same internal model and support the cognitive capabilities of human skilled subjects. The paper reviews the field of human trajectory formation, revealing in a novel manner the fil rouge that runs through motor neuroscience and proposes a computational framework for a robotic formulation that also addresses the Degrees of Freedom Problem and is formulated in terms of the force-field-based Passive Motion Paradigm.

1. Introduction

According to IFR (Internal Federation of Robotics)[1], almost 3 million robots are currently working in factories around the globe, and most of them, for safety reasons, operate in isolation from human contact. We must also consider that there is a recent trend for developing a new generation of collaborative robots (cobots), intended for direct Human–Robot interaction within a shared space; however, this is a limited sector that covers less than 5% of the market and is strongly limited, at the moment, by a *Cogniware* that is still in its infancy, although research in Cognitive Architectures for Robotics (CAR) is more than 40 years old. Technological innovations, concerning perception, interaction, and manipulation, have made cobots at least safe machines with which humans can interact with relative ease. But what about the real confidence that may allow a robust Human–Robot partnership? For example, in

E-mail address: pietro.morasso@iit.it

[1] https://ifr.org/ifr-press-releases/news/record-2.7-million-robots-work-in-factories-around-the-globe

https://doi.org/10.1016/j.cogr.2021.06.004
Received 25 May 2021; Received in revised form 12 June 2021; Accepted 21 June 2021
Available online 25 June 2021

the EIC Work Programme 2021[2], one of the EIC Pathfinder Challenges is "Awareness inside": the point is that having acknowledged that "awareness" is much more than any sensorial sophistication and any Artificial Intelligence existing artifacts, it is stated that "humans will not accept robots as trustable partners if they cannot ascribe some form of awareness and true understanding to them". This is the challenge that CARs for the next generation of Robots are facing, whether in industrial or in service-robotics, considering that the traditional difference between the two markets is going to become narrower and narrower as the issue of Human–Robot interaction, cooperation, and ultimately, partnership tends to gain center stage.

In particular, we suggest that a CAR should provide the robot with several crucial functional capabilities as the following ones:

- Feasibility check of its actions before doing them
- Understanding the consequences of its actions
- Devising an action for hypothesis testing
- Understanding the action of a partner
- Guessing the intention of a partner
- Imitating a partner
- Interpreting a supervising partner feedback
- …..

Since it is quite natural to observe that also inter-human partnership requires the cooperating partners to share a similar set of capabilities, we may postulate that, although Human and Robotic Cognitive Architectures may be different in many technical and implementation aspects, they should also share a common set of basic computational features: trajectory formation, in the general sense of conceiving and generating a large range of goal-oriented and expressive gestures, both overt (real) or covert (imagined), is one of them.

Before focusing on the main topic of this paper, let us briefly outline the alternatives (originated from neuroscience, cognitive science, but mainly from computer science and AI) that have been investigated, with different levels of detail, for characterizing and developing cognitive architectures. A recent review [1] cites a very large number of proposals in the last forty years, most of them conceived in the framework of artificial intelligence. An accepted taxonomy is articulated in three broad classes: symbolic (a.k.a. cognitivist), emergent (a.k.a. connectionist), and hybrid. The majority of current approaches are hybrid, with a small group of emergent architectures, frequently biologically inspired. However, a more specific dichotomy, for addressing the issue of Human–Robot partnership, can be derived by attempting to answer the following question: Cognition is better described in terms of "Agent-Environment Dynamics" or "Computation & Representation"? The former alternative, which can be labeled *Implicit Cognition*, leads to *Reactive Architectures*: for example, Braitenberg's "Vehicles" [2] or Brooks' concept of "Intelligence without representation" [3] brilliantly demonstrate that combining multiple reactive mechanisms can give rise to complex behaviors. Along the same line is also Chemero's emphasis on "Agent-environment dynamics" [4]. For reactive architectures, Cognition & Control are equivalent and both processes occur simultaneously on a common Realtime Axis.

The alternative to Implicit Cognition is characterized by a separation of Cognition and Control, leading to families of Prospective Cognitive Architectures, where control processes occur in real-time whereas cognitive processes develop *behind* as well as *beyond* real-time. Along the same line we should also consider a further dichotomy between approaches, derived from AI and thus based on symbolic representations (for example, Soar [5], ACT-R [6], and CRAM [7]), and formulations derived from Neuroscience, with an emphasis on Embodied Cognition, such as neurorobotics [8], evolutionary robotics [9], morphological computation [10] or robotic formulations [11,12] directly inspired by the Neural Simulation of Overt and Covert Actions [13–16]. In our opinion, such a bio-inspired approach is more appropriate for the kind of robot-human partnership that we are focusing on in this paper.

Generally, the motivation and long-term advantage of a bio-inspired approach to the development of a cognitive architecture for cooperating robots is that, as observed above, humans are not going to accept robots as trustable partners if they cannot ascribe some form of awareness and true understanding to them: awareness is a prerequisite for the real and contextualized understanding of a problem or situation and to adapt one's actions (and their consequences) to the specific circumstances. Ultimately, awareness serves the kind of coherent and purposeful behavior, learning, adaptation, and self-development we expect of intelligent systems, over longer periods, to achieve social interaction and cooperative partnership. Understanding the flexible coordination of the multiple DoFs of the human body, namely the purposive synergy formation that characterizes human gestures and actions, is one of the basic building blocks of a cognitive architecture capable of purposive and efficient actions. This paper describes a computational formulation of this motor cognitive process, starting from a historical background about multi-joint motor control. Thus, the paper is at the same time a focused review on the convergence of motor control and motor cognition concepts in the last fifty years and a detailed algorithmic formulation of the inner mechanisms of synergy formation, with novelty elements as regards the coordination of redundant degrees of freedom and the simultaneous achievements of positional and pose targets.

2. Historical background about multi-joint motor control

The historical background about multi-joint motor control is characterized by the fact that until the '70s motor control was focused on single-joint control paradigms, within a "reductionist" framework that ignored the "holistic view" implied by the Degrees of Freedom Problem [17] or the Motor Equivalence Principle [18]. The first step towards the multi-joint paradigm was the discovery of the Spatio-temporal invariants of 2D and 3D gestures:

[2] https://eic.ec.europa.eu

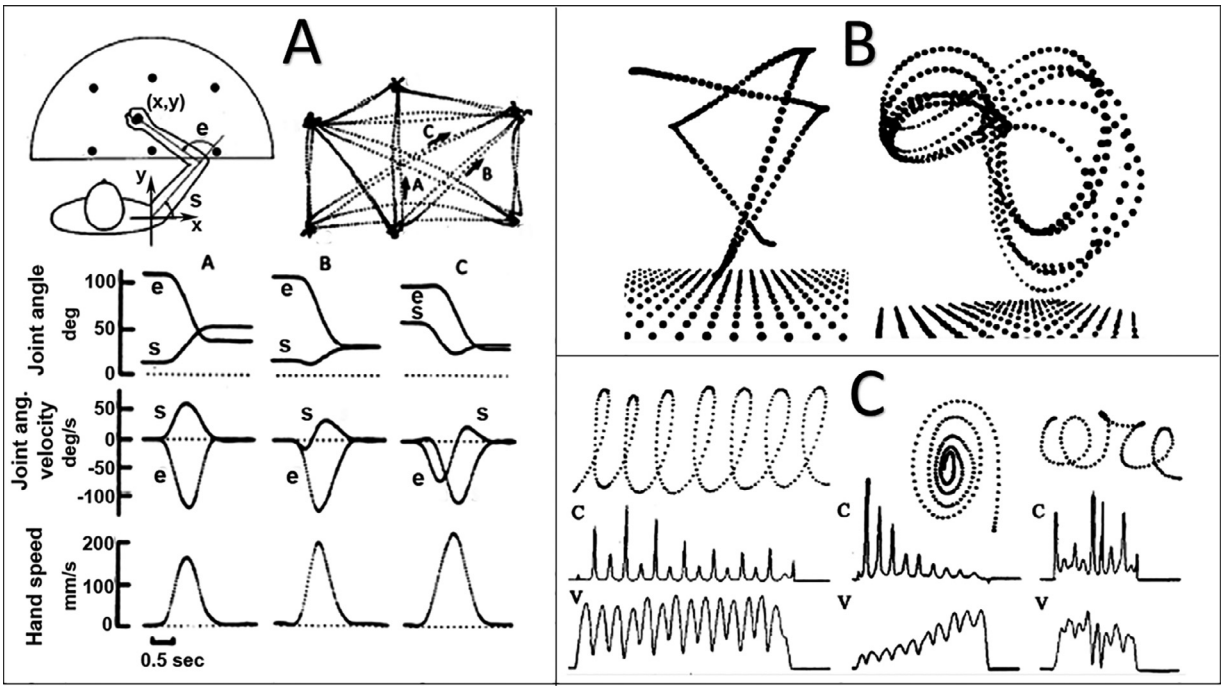

Fig. 1. Spatio-temporal invariants in trajectory formation. Panel **A**: Planar reaching movements between six target points; A, B, and C correspond to three movement examples, characterized by the joint rotation patterns and the corresponding speed profiles of the joints and the hand; note the invariant bell-shaped speed profiles. Panel **B**: Example of hand trajectories in three dimensions that maintain the same invariant features of planar trajectories. Panel **C**: Three examples of continuous hand scribbles displayed as digitized trajectories, including the profiles of the velocity (V) and curvature (C); note the anti-correlation of the two profiles.

- The bell-shaped speed profile & isochrony of reaching movements [19]: Fig. 1A.
- The anti-correlation of speed & curvature profiles of reaching and drawing [20]: Fig. 1C.
- The piece-wise planar organization of 3D arm gestures [21]: Fig. 1B.

This prompted the following questions about such Spatio-temporal invariants: where do they come from and how can we explain them? Among the different explanations that were attempted we may quote the following ones:

- The 2/3 power law [22]: it was found that the velocity profile $v(t)$ and the curvature profile $c(t)$ are approximately related by the following relationship: $v(t) = \gamma\ c(t)^{-1/3}$.
- The minimum jerk model [23]: it was shown that the recorded hand patterns can be characterized by a minimization of the hand jerk (the time derivative of the acceleration profile).
- The VITE model (Vector-Integration-To-Endpoint) [24]: this is a synergy formation process that integrates over time a difference vector, computed as the difference between the target position and the present position of the end-effector.
- The PMP Model (Passive Motion Paradigm) [25]: the model provides a force-field-based simulation model, capable to coordinate implicitly the motion of the end-effector and the corresponding, redundant DoFs.
- Active Inference [26]: this concept rests upon the idea that the brain uses an internal generative model (similar to the PMP) to predict incoming sensory data.

As better explained in the following, it can be shown that the PMP model explains all the others and provides a solid computational framework for both human motor neuroscience and humanoid robot cognition. The basic rationale of the PMP model of trajectory formation is the same as the models of motor control based on a force-field approach, namely the idea that multi-joint motor coordination is the consequence of force fields applied to an internal representation of the body, force fields that express goals, intentions, environmental constraints, etc. This idea can be traced back to the EPH (Equilibrium Point Hypothesis) [27–29] and to the Neural Simulation of both overt and covert actions [13].

The proposal supported by this article is that the PMP model can be the basic kernel of a Prospective Architecture for Human–Robots partnership based on Embodied Cognition, Motor Imagery & Neural Simulation of Overt and Covert Actions, considering the mounting evidence that generation, observation, imagination, and understanding of gestures share similar functional networks in the brain. Moreover, also the main temporal aspects of synergy formation are shared by real and imagined actions, in particular the Fitts's law [30] and the two-thirds power law of real scribbles [31].

3. Trajectory formation of the end-effector using a terminal attractor gradient descent

The trajectory formation mechanism of a reaching movement in 2D, based on a force-field model, can be represented by an attractive, virtual force field F, centered in the target position P_T and applied to the moving end-effector $p(t)$:

$$F(t) = K\ \big(P_T - p(t)\big) \tag{1}$$

If the matrix K is proportional to the unitary matrix ($K = k\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}$), the force field is isotropic with straight force lines; however, we can also instantiate an anisotropic field, characterized by curved force lines, for example by inserting a rotation matrix: $K = k\begin{bmatrix}\cos(\alpha) & \sin(\alpha)\\ -\sin(\alpha) & \cos(\alpha)\end{bmatrix}$, where α is the rotation angle of the field. By applying the force field to the end-effector, modeled as a point object with a viscous coefficient B, the trajectory of the end-effector will be generated by integrating the following equation:

$$\dot{p} = \frac{K\big(P_T - p(t)\big)}{B} \tag{2}$$

The trajectory will reach the target along a straight or a curved path, depending on the isotropic/anisotropic type of force field, with a decreasing exponential speed profile, i.e. in an asymptotic way. In contrast, as shown in Fig. 1, human trajectories are characterized by symmetric, bell-shaped speed profiles. The minimum-jerk model explains this kinematic feature as the result of an optimal control mechanism that aims at minimizing the jerk of the movement while approaching the target. However, a simpler mechanism allows to obtain the same result, without any optimization task: it consists of the non-linear modulation of the attractor force-field of Eq. (2) by a command or gating signal $\Gamma(t; t_0, T)$, thus substituting Eq. (2) with the following one:

$$\dot{p} = \Gamma(t)\ \frac{K\big(P_T - p(t)\big)}{B} \tag{3}$$

Γ has a duration T, a starting time t_0, and a profile that initiates with a null value and then diverges to infinity as the time approaches the final instant or deadline: Appendix A1 provides a possible mathematical expression of Γ. The idea is that to avoid the asymptotic gradient descent of Eq. (2) it is sufficient to progressively boost the field intensity as the time approaches the deadline, thus forcing the system to achieve the final equilibrium in the finite, prescribed time slot: this technique is called *terminal attractor dynamics* [32] and was originally formulated for the quick recall of associative memories. Moreover, the rationale of this method did find support in computational neuroscience: the Γ function can be described as a kind of *neural pace-maker* [33], with a biologically plausible implementation: the loop that links different cortical areas with the basal ganglia and then the thalamus, emphasizing the well-established role of the basal ganglia in the initiation and speed–control of voluntary movements. A similar mechanism is also found in the VITE model (Vector Integration To Endpoint) [24], where it is assumed that the synergy formation process integrates over time a difference vector (DV), computed from the Target Position and the Present Position of the end-effector, multiplied by a GO-signal that is quite similar to the Γ function. The terminal attractor procedure can be applied both to isotropic and anisotropic force fields, as shown by the simulations of the model in Fig. 2 that is carried out by integrating over time Eq. (3). All the simulations presented in this paper were carried out using Matlab® (MathWorks), with the forward Euler method for integrating the differential equations with a time step of 0.1 ms.

It is important to note (as shown by the figure) that the use of the terminal attractor mechanism for the force-field-based trajectory formation model produces automatically a symmetric bell-shaped velocity profile without any need for an optimization process. Moreover, this result depends very weakly on the intensity of the field, i.e. the value of K, because it is dominated by the Γ -command.

3.1. Shape formation for general gestures

The model of trajectory formation described above is not restricted to simple reaching movements but can be easily extended to more general gestures, like hand drawing or handwriting that are concerned with “shape formation”. The force-field-based approach, timed by the non-linearity of the Γ -command, naturally meets the need of compositionality that is essential for the construction and editing of complex gestures, in a similar way to the problem faced by articulatory-prosodic patterns of connected speech [34]. More specifically, it is easy to demonstrate that continuous, arbitrarily shaped trajectories, such as handwritten scribbles or cursive writing can be produced by the simple superposition of sequences of reaching motions (or hand strokes) obtained by starting the Γ-command of the next stroke before the termination of the previous stroke. Fig. 3 shows a simple two-strokes example: the nominal target of the first stroke is not reached due to the overlapping of the two strokes. The procedure can be iterated endlessly, demonstrating how the discontinuous mental representation of a gesture, in terms of a sequence of virtual target points, is seamlessly transformed in a smooth continuous gesture.

3.2. Shape formation and the power-law

The shape formation mechanism, based on the partial overlap of force-field driven hand strokes, provides a simple explanatory framework for addressing the space-time invariance represented by the power law. We may consider that the trajectories of generic human movements are characterized by a robust local relationship between the geometrical and temporal aspects, represented by the curvature and speed profiles, $c(t)$ and $v(t)$, respectively. As previously mentioned, a model that was proposed years ago for describing

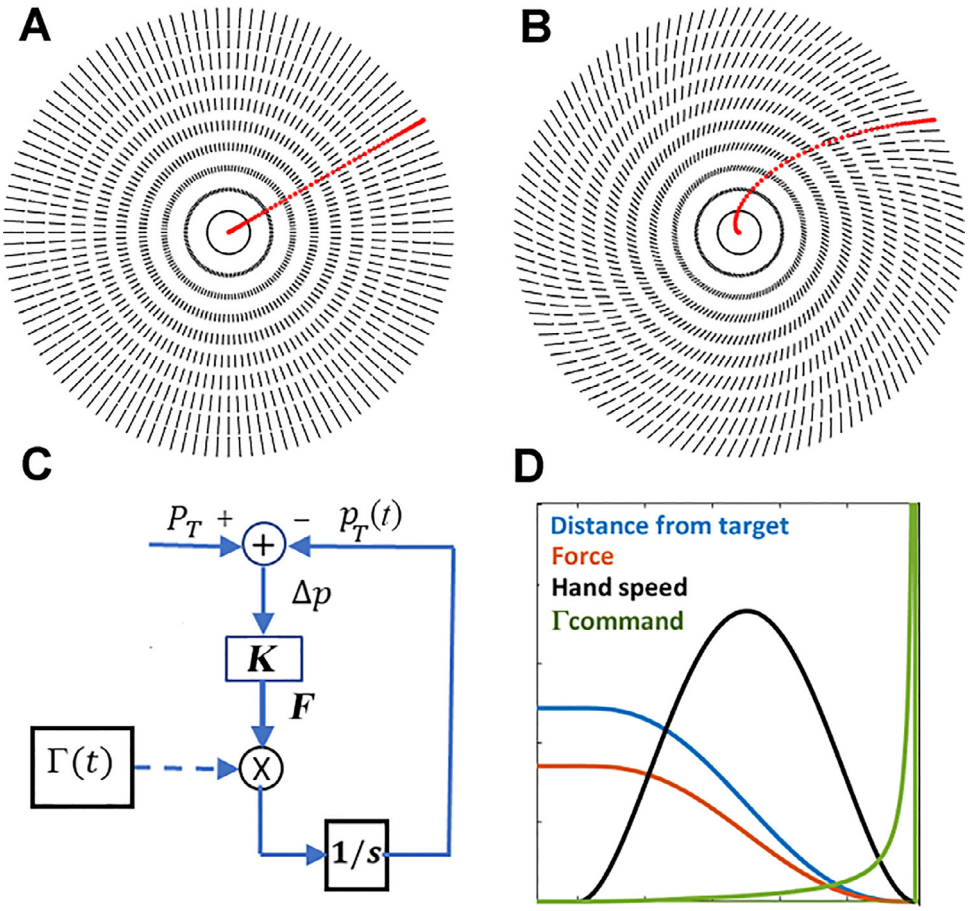

Fig. 2. Trajectory formation in two dimensions as terminal attractor gradient descent in a force field $\boldsymbol{F}$ that can be isotropic (panel **A**) or anisotropic (panel **B**). Panel **C** is the block-diagram which corresponds to equation 3 and is used in the simulations: Γ(t) is the non-linear command that forces the gradient descent to equilibrium in finite time; "**1/s**" stands for integration over time; $K = k[\cos(\alpha) \quad \sin(\alpha) - \sin(\alpha) \quad \cos(\alpha)]$, with $k = 1$, $\alpha = 0$ (simulation of panel A), and $\alpha = -15\ deg$ (simulation of panel B). Note that the bell-shaped speed profile (panel **D**) is an implicit side-effect of the terminal attractor mechanism, represented by the block diagram of panel C.

such relationship is the so-called two-thirds power law: $v(t) = \gamma\, c(t)^{-1/3}$. The underlying rationale of this proposal is that such *law* may be somehow wired in the motor control part of the brain, in some specific network, and thus what should be controlled by the brain, to adapt the control mechanism to different circumstances, would be the activation of the network with appropriate modulation of the gain γ. The power law implies that the speed and curvature of a gesture cannot be controlled independently but the time profile of one variable is inversely correlated to the other: the kinematic analysis of continuous gestures shows indeed that the peaks of the curvature profile are invariably synchronized with the minima of the speed profile and the peaks of speed are synchronized with the curvature minima. The power law can certainly explain such empirical results but they are also consistent with a much simpler mechanism based on a straightforward generalization of the shape-formation model, exemplified in Fig. 3: the superposition of overlapped strokes. Fig. 4 illustrates a movement that consists of the pseudo-repetition of six elliptical figures that are generated by superimposing twelve overlapped curved strokes with target points alternated on the two sides of the elliptical set. If the recorded patterns are plotted in a log-log scale we can verify the adherence to the pawer law; moreover, the bottom panel of Fig. 4 also demonstrates the strong anti-correlation of the curvature and speed profiles. Thus, there is no need to support the idea that the brain holds a special representation of the power-law as an internal control model because the same result is obtained by generalizing the compositionality of the multi-stroke synergy formation based on force fields and terminal attractor dynamics.

4. The Passive Motion Paradigm: how the motor intention "pulls" the motion of the body

In the previous sections, we reviewed the empirical evidence about the Spatio-temporal invariances of human gestures and the possible computational mechanisms that can explain them. However, this is not the end of the story because there is still a big question to answer about all of this evidence: what does it tell us about the underlying multi-joint coordination that makes such invariances possible? Answering this question is also a way to address the vexing question about the degrees of freedom problem: how can

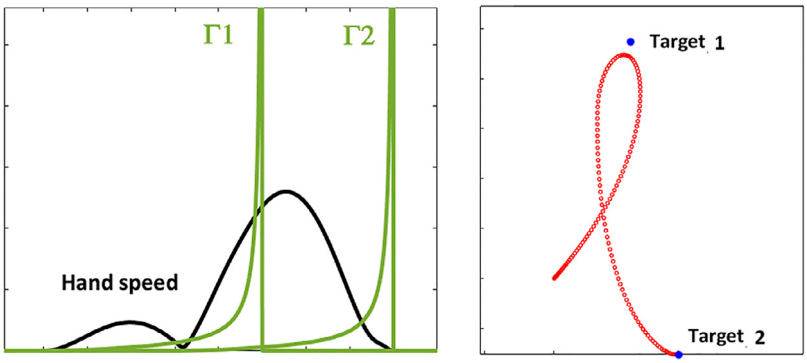

Fig. 3. Generation of continuous scribbles through the time overlap of subsequent curved strokes. In this example, the l-shaped gesture is generated by two curved strokes, attracted by target 1 and target 2, respectively, and synchronized by two Γ-commands (Γ_1 and Γ_2), with a 50% overlap. The figure plots the results of the simulation of the model in fig. 2C with $\alpha = -15\ deg$ for the two strokes.

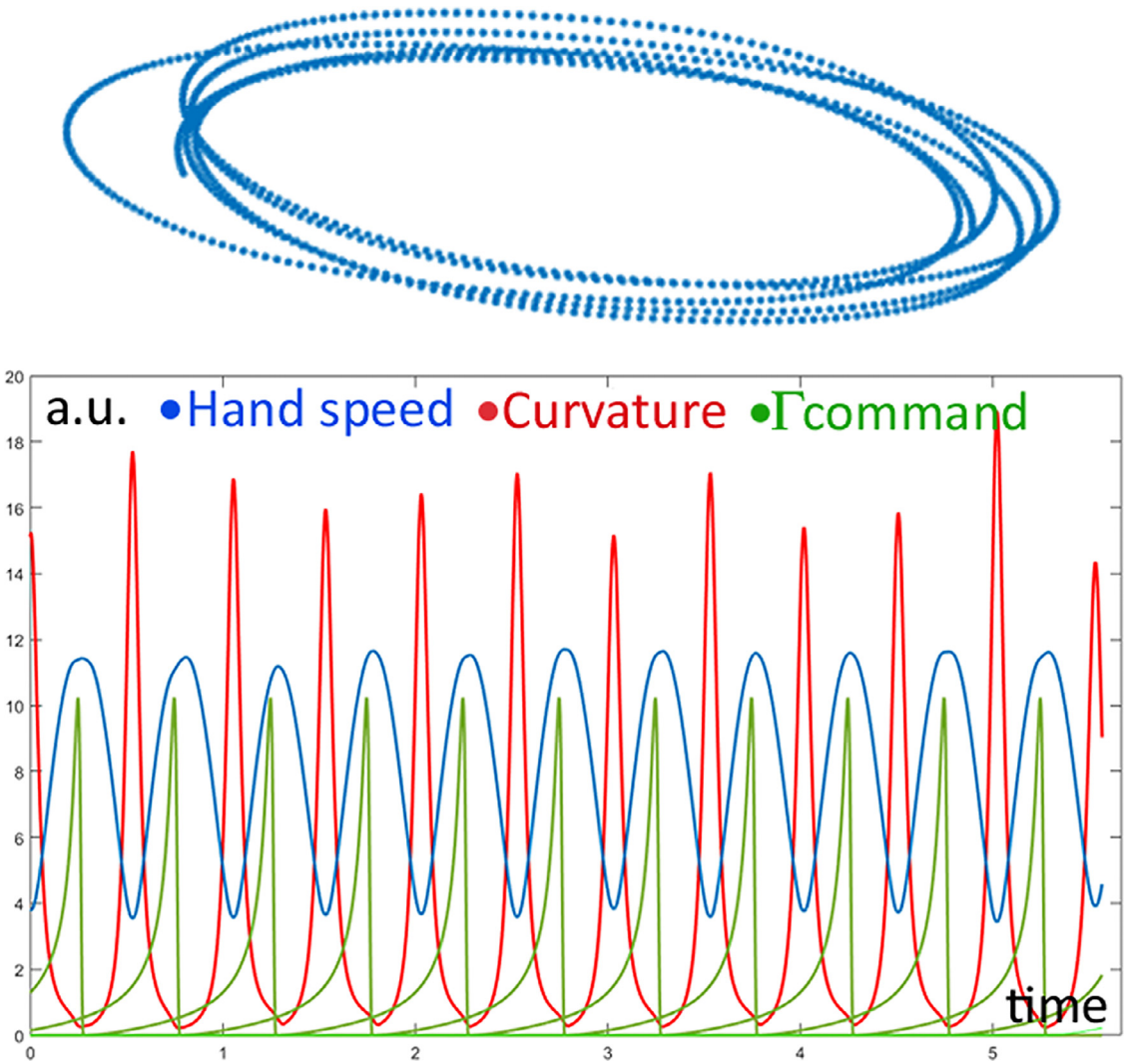

Fig. 4. Iterated elliptical trajectories generated by a sequence of curved hand strokes: the target points are alternated on the left and the right of the scribble, with a slightly randomized jitter. The Γ-commands that drive each stroke have a 50% overlap between one stroke and the next one. The curvature and speed profiles are anticorrelated; moreover, by displaying the simulated profiles with a log-log plot we obtain a regression line with a negative slope close to -1/3, as suggested by the 2/3 power law. The figure plots the results of the simulation of the model in fig. 2C with $\alpha = -15$ deg for all the eleven strokes.

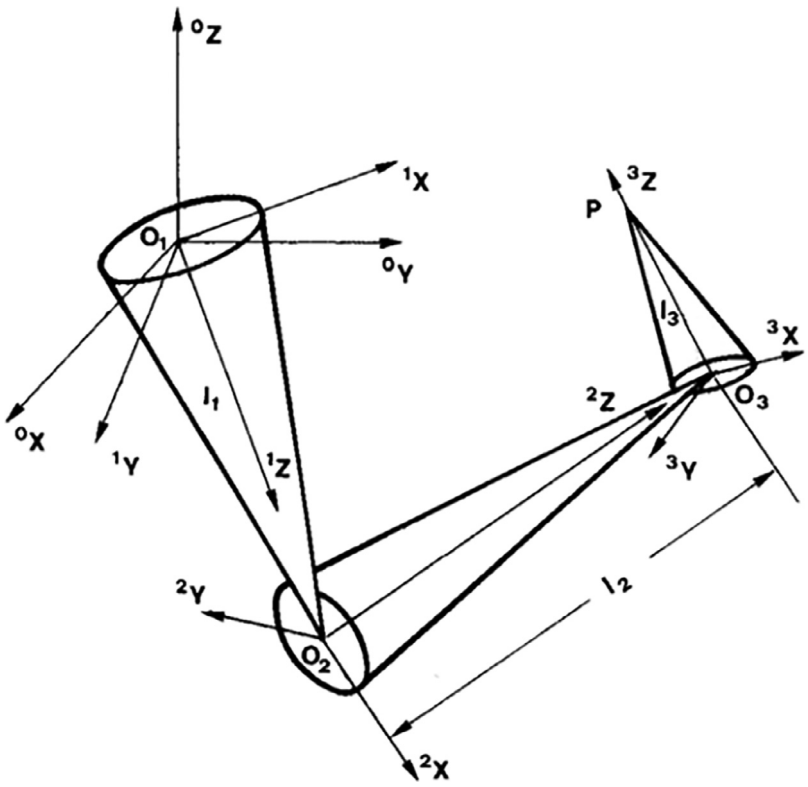

Fig. 5. Anthropomorphic arm with 7 DoFs: 3 at the shoulder, 1 at the elbow, and 3 at the wrist. In the simulations $l_1 = 0.5$ m, $l_2 = 0.5$ m, $l_3 = 0.24$ m.

the brain choose coordinated multi-joint patterns that make possible the Spatio-temporal invariances observed at the end-effector level? The proposed solution, outlined in this paper, starts from the observation that human posture is maintained by the equilibrium between the length-tension properties of opposing antagonistic muscles; in single-joint motion control, this finding led to the idea that movements result from a smooth shift of the equilibrium point caused by a change of the neural input [27,29], a.k.a. Equilibrium Point Hypothesis (EPH).

The extension of this idea from single-joint to multi-joint movements was investigated later on, at the theoretical level [35] as well as an experimental level [36]. However, this line of research did not provide any clue about the neural mechanisms capable to provide this kind of multi-joint coordination. On the other hand, we believe that the ample literature about the central role of motor imagery in motor cognition suggests that the EPH is more general than the direct consequence of the elastic properties of muscles: The Passive Motion Paradigm (PMP) is the computational implementation of this idea [25]. PMP reverses the causality relation between motor control (coordination of multi-joint movements) and achievement of the goal (reaching/touching/gesturing): the matter is not to instruct the motor controller to "push" the end-effector to the target but to allow the goal to "pull" the overall kinematic chain, allowing a "passive" rearrangement of the joint rotation angles, modulated by the virtual compliance of the joints. This idea is further extended in the paper, as a novel development, with the goal to integrate into the same computational framework the treatment of non-linear constraints, such as the joint limits, the bimanual coordination, and the coordination of positional and pose reaching.

In the rest of the paper, it is assumed that the anthropomorphic arm has 7 DoFs (Fig. 5): 3 at the shoulder ($q_1,\ q_2,\ q_3$), one at the elbow (q_4), and three at the wrist ($q_5,\ q_6,\ q_7$). The DoFs are represented according to the yaw/pitch/roll convention, which associates a specific rotation matrix to each joint (see Appendix A2). Although the illustrated simulations adhere to this convention for the 7 DoFs arm, the same computational model can be easily adapted to different conventions and additional DoFs: the main difference would be the calculation of specific Jacobian matrices.

For illustrating the rationale of the PMP model, let us consider, for example, that in the case of reaching movements the goal is to allow the position of the end-effector to achieve a target point P_T (three variables), coordinating the motion of the seven DoFs of the arm. Thus, the degree of redundancy is four and for any reaching task there are ∞^4 possible solutions or none if the target is not reachable because of the arm's length or the limited RoM (Range of Motion) of the joints. In agreement with the moving equilibrium point hypothesis mentioned above, the PMP mechanism does not pull the end-effector directly to the final target P_T, but to the moving target point $p_T(t)$, generated in agreement with Eq. (3) (see also the block diagram of Fig. 2C). Specifically, the block diagram of the PMP model is articulated into two blocks, **A** and **B** as shown in Fig. 6.

Block A: it generates the smoothly moving target $p_T(t)$ through a virtual elastic force field F_T, proportional to the difference between the final target P_T and the current moving target $p_T(t)$. The force field is driven by the 3×3 stiffness matrix K_1 that may or

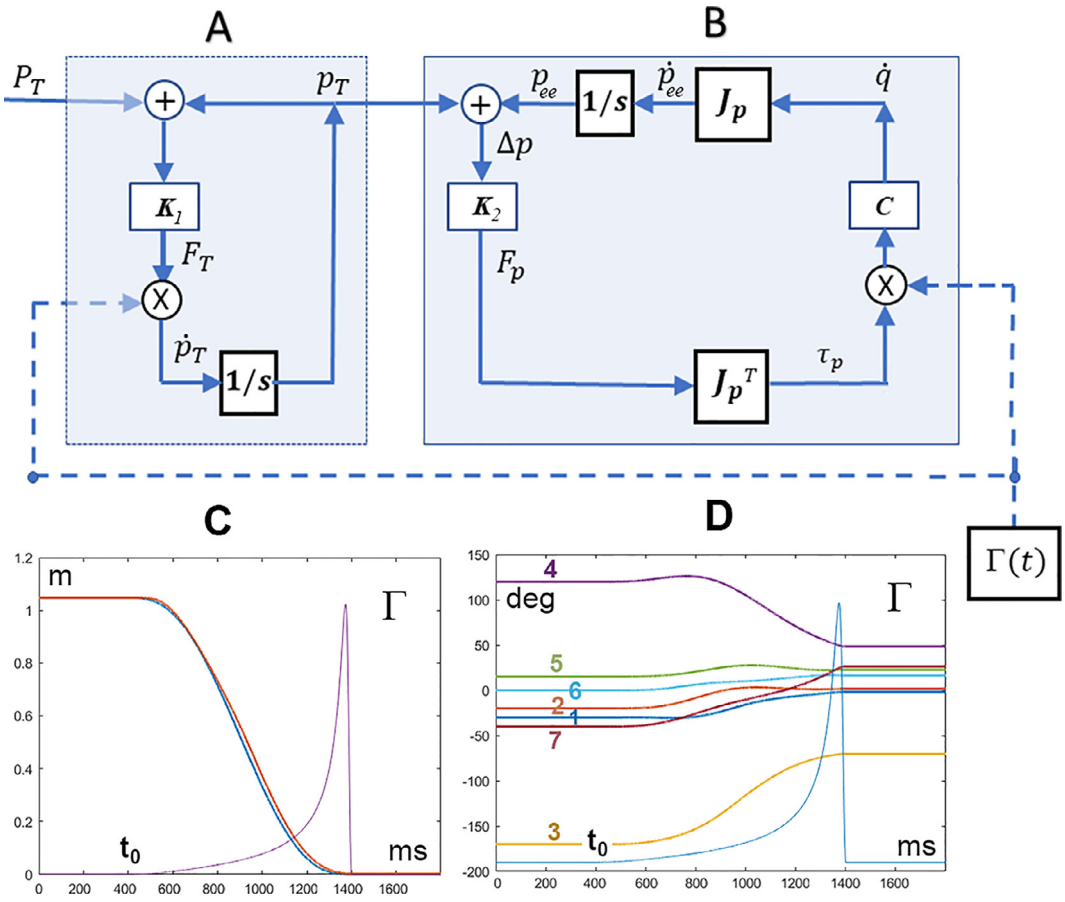

Fig. 6. Synergy formation for a multi-joint arm. The block diagram that implements the PMP model for a reaching movement (P_T is the final target to be reached in 1 second after $t = t_0$) is divided into two parts: module **A** generates the moving target $p = p_T(t)$ integrating the force field F_T that attracts the moving target to the final target, gated by the Γ command (corresponds to Eq. (4)); module **B** is the PMP model that attracts the end-effector to the moving target with the force field F_p; the force field is mapped into the corresponding torque field τ_p through the transpose Jacobian matrix ${J_p}^T$, is gated by the same Γ-command, and distributed to all the joints through the compliance matrix C, yielding the array of joint velocities $\dot{q}$ that are mapped into the corresponding velocity vector of the end-effector $\dot{p}_{ee}$ and then integrated, closing the loop (corresponds to Eq. (7)). Panel **C** shows the time course of the distance from the target of the moving target (blue line) and the position of the end-effector (red line). Panel **D** plots the evolution of the seven DoFs of the anthropomorphic arm. In both plots the Γ-command is saturated (ideally, it can diverge to infinity) and scaled to fit the figure; this command allows the implicit synchronization of the two parts (A and B) of the synergy formation process. Panels C, D plot the results of the simulation of the model in panels A-B with $K_1 = 1$, $K_2 = 50$, and $C = I_7$.

may not include a rotational component; moreover, the field is gated by the Γ-command to induce a terminal-attractor dynamic. In summary, block A of the PMP model is described by the following equation:

$$\begin{cases} F_T(t) = K_1\left(P_T - p_T(t)\right) \\ \dot{p}_T = \frac{\Gamma(t) F_T(t)}{B} \end{cases} \tag{4}$$

Block B: it generates at the same time the trajectory of the end effector $p_{ee}(t)$ and the coordinated joint rotations of the arm $q(t)$. The virtual elastic force field F_p attracts the end effector to the moving target p_T. Again, this field may or may not include a rotational component according to the choice of the stiffness matrix K_2. Thus, in the PMP model, there is a concurrent "double pulling force": from the moving target to the final target (F_T) and from the moving target to the moving hand (F_p). The latter force field is transformed into the corresponding torque field (τ_p) to be applied to all the joints of the kinematic chain by using the transpose Jacobian matrix of the kinematic chain:

$$\tau_p = {J_p}^T \, F_p \tag{5}$$

This equation maps generalized forces from the end-effector space (a.k.a. task space) to the joint space and must be employed together with the following companion equation that maps generalized motion in the opposite direction (from the joint space to the task space):

$$\dot{p}_{ee} = J_p \, \dot{q} \tag{6}$$

The two equations, taken together, express the invariance of the virtual elastic energy stored in the kinematic chain. In summary, the block **B** of the PMP model is described by the following pair of equations:

$$\begin{cases} F_p(t) = K_2\left(p_T(t) - p_{ee}(t)\right) \\ \tau_p(t) = J_p{}^T F_p(t) \end{cases}$$
$$\begin{cases} \dot{q} = C\Gamma(t)\tau_p(t) \\ \dot{p}_{ee} = J_p\,\dot{q} \end{cases} \tag{7}$$

The basic PMP model described above is used by numerical integration of Eqs. (4) and (7). The simulation of Fig. 6 shows that the virtual torque field τ_p distributes incremental rotations to the seven joints of the humanoid kinematic chain through the 7×7 Compliance matrix C. In the simplest case, the matrix is proportional to the unitary matrix and this means that all the joints respond to the virtual torque vector with an equal gain: decreasing the compliance parameter of a joint relative to the others means that such joint yields with smaller speed and such reduction is automatically compensated by an increase of the other joints. Moreover, the diagram shows that the torque field is gated with the same Γ-command that also affects the force field driving the moving target, thus providing a robust and implicit synchronization of the two processes (blocks A and B). In particular, panel C of Fig. 6 shows that the distances from the final target of both the moving target and the end effector smoothly vanish in synchrony with the Γcommand; moreover, panel D shows the coordinated joint rotation patterns generated by the integration of Eq. (7). In the simulations $K_1 = 1$ and $K_2 = 50$.

The Jacobian matrix for the position of the end-effector is defined as follows:

$$J_p = \frac{\partial p}{\partial q} \tag{8}$$

For an anthropomorphic arm with 7 DoFs, this is a 3×7 matrix. The appendix shows how it can be calculated explicitly in real-time or approximated by a multi-layer neural network, trained by utilizing backpropagation. Fig. 6 refers to a reaching example, where the target is located at a distance of 1.04 m from the initial position of the end-effector, and illustrates the smooth inter-joint coordination induced by the Γ-command.

4.1. Bimanual coordination

The crucial synchronization effect induced by the Γ-command is further illustrated if we consider bimanual reaching, i.e. a task in which the two end-effectors are supposed to reach the assigned target points at the same time, independent of the initial positions of the two hands and the positions of the targets. This includes also the special case that the two target points are coincident (see the example of Fig. 7). For bimanual reaching, the PMP model employs two copies of the A and B modules: one copy (A_r, B_r) for the right arm and another copy (A_l, B_l) for the left arm.

It is important to note, as an element of flexibility of the PMP model, that although in the default situation the two pairs of models may correspond to two identical arms, it is quite possible to apply the same computational formulation also to the case in which the two robotic arms are different, either for the geometrical parameters or the number of DoFs. This means, for example, that although the B modules of the two arms are formally identical, the corresponding Jacobian matrices may be different. In all the cases, the same Γ-command can allow the two arms to synchronize their reaching actions, as shown by the graph of the figure that plots the concurrent smoothly decreasing distances of the two arms from the corresponding targets.

4.2. Avoiding the joint limits, i.e. enforcing the RoM

As a further extension of the force-based PMP model, we may consider Fig. 8 that illustrates how the block-diagram implementing the PMP model, which includes modules A and B, can be extended by integrating a mechanism (module C) for avoiding joint rotations beyond the mechanical/physiological joint limits, thus generating coordinated motion patterns that satisfy the RoM of all the joints. This mechanism is force-based as well and exploits the redundancy of the arm. The idea is to represent the joint limits not as geometrical constraints to the kinematic equations of the arm but as generalized repulsive forces to be summed to the attractive forces associated with the selected target points. In this manner, the redundant DoFs of the arm will be coordinated implicitly, in such a way to achieve, at the same time, the reaching part of the task and the RoM avoidance subtask. In particular, let us suppose that the RoM of each joint is characterized as follows:

$$q_{min} < q < q_{max} \tag{9}$$

The following torque applied to each joint will keep the joint angle away from the joint limits:

$$\tau_{RoM} = -K_{RoM}\left[e^{\frac{(q-q_{max})}{\Delta q}} - e^{\frac{-(q-q_{min})}{\Delta q}}\right] \tag{10}$$

where $\Delta q = (q_{max} - q_{min})/\gamma$. K_{RoM} is the gain of the repulsive torque and γ is the normalization factor for the RoM of the joints. This virtual torque is plotted in panel **D** of Fig. 8: the torque amplitude is close to zero in the central part of the RoM and sharply diverges when the joint angle approaches the joint limits on either side. In the simulations $K_{RoM} = 1$ and $\gamma = 50$.

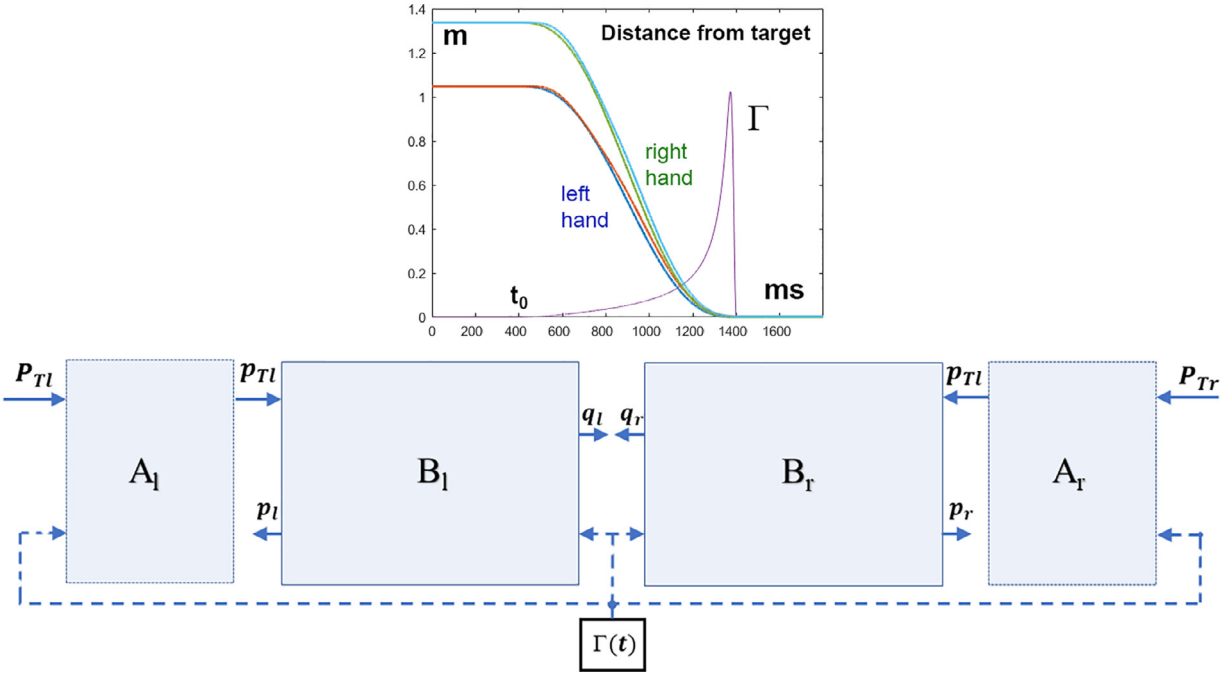

Fig. 7. Bimanual synergy formation obtained with two instances of the PMP model (one for the left arm and the other for the right arm), synchronized by the same Γ-command: this command induces the two end-effectors (p_l and p_r) to reach the corresponding targets (P_{Tl} and P_{Tr}) simultaneously, independent of the initial distance. The top panel plots the distance from the target for the two hands of both the moving target and the corresponding end-effector.

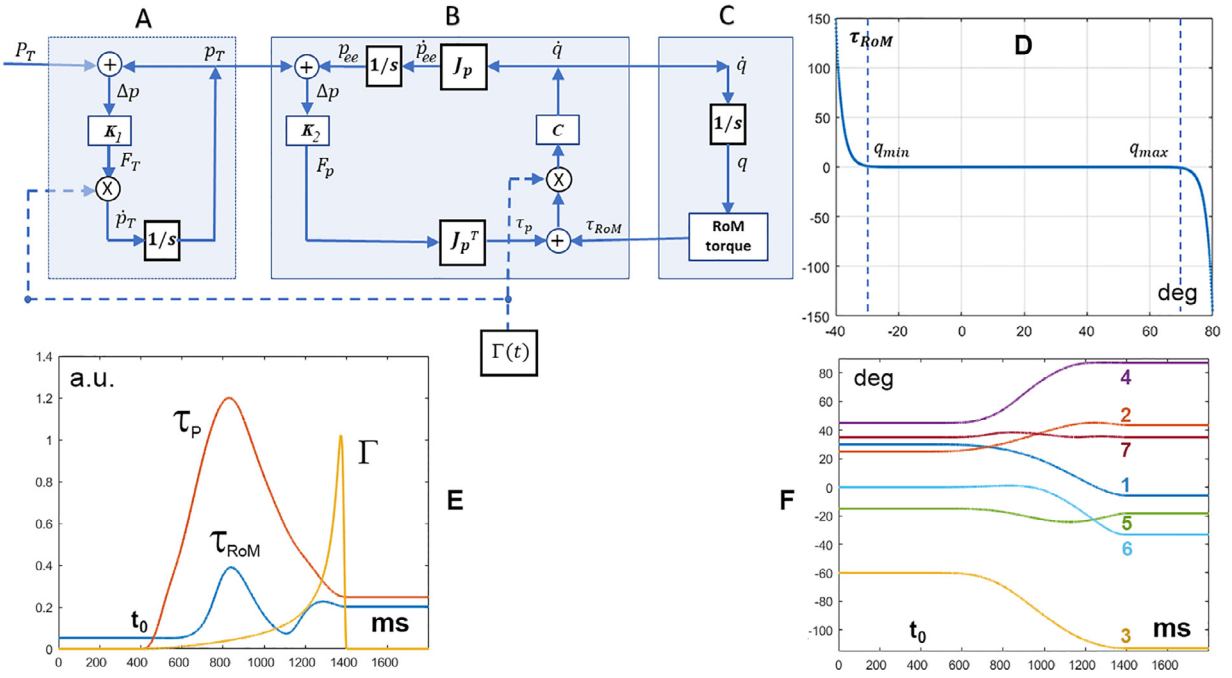

Fig. 8. Extension of the basic PMP model (modules **A** and **B**) to include a module (panel **C**) that generates a virtual torque vector τ_{RoM} that keeps each joint away from the specific join limits; module A corresponds to Eq. (4), module B to Eq. (7), and module C to Eq. (10). Panel **D** plots the variation of the torque in the range of motion of the joint. Panel **E** plots the time course of the amplitudes of both torque vectors. Panel **F** shows the time course of the seven joint angles. In the simulations $K_1 = 1$ (module **A**), $K_2 = 50$ (module **B**), $K_{RoM} = 1$ and $\gamma = 50$ (module **C**), $C = I_7$.

The figure shows a simulation of this extended PMP model. In particular, the block-diagram is quite similar to the one of Fig. 6: block **A** is the same (Eq. (4)), block **B** is extended, with the addition of block **C** which generates on-line τ_{RoM} through Eq. (10), and is mathematically represented by the following pair of equations that extend Eq. (7):

$$\begin{cases} F_p(t) = K_2\big(p_T(t) - p_{ee}(t)\big) \\ \tau_p(t) = J_p{}^T F_p(t) \end{cases}$$
$$\begin{cases} \dot{q} = C\Gamma(t)\big[\tau_p(t) + \tau_{RoM}\big] \\ \dot{p}_{ee} = J_p\, \dot{q} \end{cases} \tag{11}$$

Panel **E** of the Fig. shows the time course of the amplitude of the two torque vectors, coordinated by the common Γ-command; panel **E** plots the evolution of the 7 joint angles, smoothly coordinated in such a way to guide the hand to the designated target while keeping each joint angle inside its RoM. Note that the final positional error (i.e. the magnitude of the Δp vector) is small in any case, but the final value of the magnitude of the τ_{RoM} vector is far from neglectable and, in any case, is larger than the initial value. This means that in the course of the reaching movement the joint synergy is evoking a push-away response, from a subset of joints, inducing automatically an increase of the target-reaching pulling torque τ_p. It is also worth considering that, despite the redundancy of the arm, there is no need to subdivide the whole set of joints in Controlled and Uncontrolled Manifolds because all the joints share the common force field. The uncontrolled manifold concept [37] is one of the different approaches proposed for solving the degrees of freedom problem. It is based on the idea that, for any task, the brain should focus on the subset of DoFs that are more important: the PMP model can implicitly achieve the same effect, without an explicit selection of the crucial set.

5. Extending the Passive Motion Paradigm to the pose of the end-effector

For some tasks it is sufficient to specify the position or the trajectory of the end-effector, say the palm or the fingertip, in three dimensions. This task can be fully described in the Euclidean space. In other tasks, this is not sufficient because it is necessary to specify not only the position of the end-effector but its pose as well. This is a problem that involves rotations in three dimensions and cannot be analyzed in the Euclidean space, mainly because rotations are non-Euclidean in nature. Thus, if we consider gesture formation in full generality, i.e. with a position and a pose target to be reached concurrently, we need to fuse a Euclidean and non-Euclidean sub-task.

In the case of position targets, the error vector that supplies the force field at the basis of the PMP model is the standard vector difference Δp, between the target position and the current position of the end-effector. In contrast, in the case of pose targets there is the problem of parameterizing rotations, which may be handled in different ways, with different metrics: rotation matrices, Euler angles, quaternions, exponential map, Rodrigues parameters, etc. [38,39].

In this section, we describe the extension of the PMP model to the pose of the end effector, with the requirement of synchronizing the two reaching subtasks. In particular, the pose is expressed in terms of both the rotation matrix and Rodrigues parameters as explained in the following. With the 7 DoFs anthropomorphic arm, given the array of joint angles $q = [q_1,\ q_2 \dots\ q_7]$, the pose of the end-effector can be calculated online with the following equation, where the seven 1-DoF matrices are defined in Appendix A2:

$$R_{ee} = [r_{ij}] = R_1(q_1)\ R_2(q_2)\ R_3(q_3) R_4(q_4) R_5(q_5)\ R_6(q_6)\ R_7(q_7) \tag{12}$$

The pose can also be represented by a single, unique rigid rotation of the end-effector that allows reproducing the given pose from the reference frame of the environment. Such rotation is identified by the rotation axis or unit-vector u and a rotation angle θ and is represented by the so-called Rodrigues vector [40][3]:

$$\rho_{ee} = [\rho_1\ \rho_2\ \rho_3]' = tan\frac{\theta}{2}\, u \tag{13}$$

R_{ee} and ρ_{ee} are equivalent and it is possible to transform one into the other and vice versa according to the following pair of equations:

$$R_{ee} = \frac{1}{1+\rho_1^2+\rho_2^2+\rho_3^2}\begin{bmatrix} 1+\rho_1^2-\rho_2^2-\rho_3^2 & 2(\rho_1\rho_2-\rho_3) & 2(\rho_1\rho_3+\rho_2) \\ 2(\rho_1\rho_2+\rho_3) & 1-\rho_1^2+\rho_2^2-\rho_3^2 & 2(\rho_2\rho_3-\rho_1) \\ 2(\rho_1\rho_3-\rho_2) & 2(\rho_2\rho_3+\rho_1) & 1-\rho_1^2-\rho_2^2+\rho_3^2 \end{bmatrix} \tag{14}$$

$$\begin{matrix} \rho_3 = (r_{21}-r_{12})/(1+r_{11}+r_{22}+r_{33}) \\ \{\rho_2 = (r_{13}-r_{31})/(1+r_{11}+r_{22}+r_{33}) \\ \rho_1 = (r_{32}-r_{23})/(1+r_{11}+r_{22}+r_{33}) \end{matrix} \tag{15}$$

This pair of equation is the basis for configuring a block diagram for achieving a pose target similar to the block diagram in Fig. 6 for achieving a positional target, which included two modules: a module **A**, for generating a moving target aimed at the final target, and a module **B**, for pulling the end-effector toward the moving target and distributing the action to the redundant set of joint rotations by means of the Jacobian matrix J_p. These two modules are re-labeled $\mathbf{A_p}$ and $\mathbf{B_p}$ for distinguishing them from the new pair

[3] It is worth considering that, by definition, the magnitude of a Rodrigues vector is the tangent of half the angle of the unique finite rotation that corresponds to a given pose.

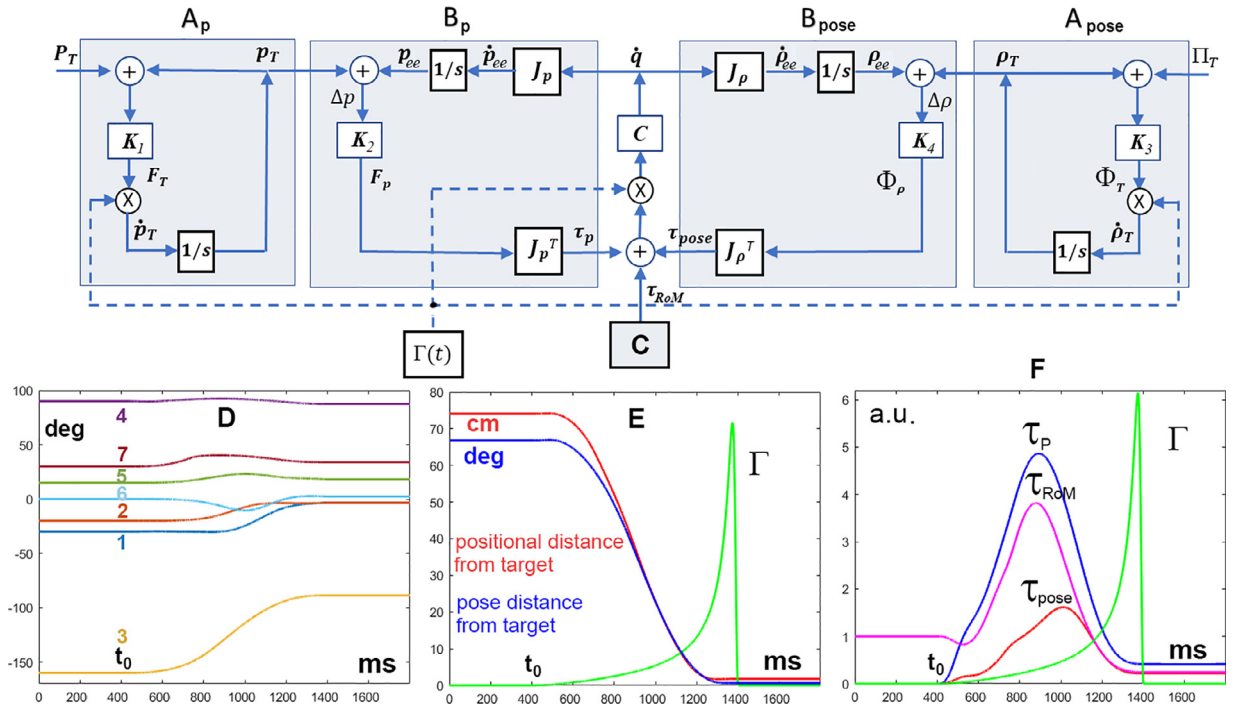

Fig. 9. Extension of the basic PMP model (modules $\mathbf{A_p}$ and $\mathbf{B_p}$ focused on positional reaching) to handle pose reaching (modules $\mathbf{A_{pose}}$ and $\mathbf{B_{pose}}$), in addition to handling the joint limits as well (module **C**). The same Γ command synchronizes the three subtasks (positional reaching, pose reaching, and RoM maintenance): the integration of the three processes is achieved by combining the three torque fields (τ_p, τ_{pose}, and τ_{RoM}), plotted in panel F. Panel E shows that the two targets (positional and Pose) are reached smoothly and simultaneously as a result of the smooth coordination of the 7 DoFs. In the simulations $K_1 = 1$, $K_2 = 50$, $K_3 = 1$, $K_4 = 15$, $K_{RoM} = 1$ and $\gamma = 50$ (module **C**), $C = I_7$.

($\mathbf{A_{pose}}$ and $\mathbf{B_{pose}}$) which has the purpose to control the pose of the end-effector to reach the final pose. The rationale of the new block diagram is the same, namely it is based on force-fields and terminal attractor dynamics, with the following differences: i) the metrics for expressing error vectors and distances is not Euclidean but spherical; ii) the Jacobian matrix for mapping joint rotations into pose modifications is different (J_ρ instead of J_p) and calculated in a different way, as shown in the Appendix. Fig. 9 shows a combined block diagram that links the four modules mentioned above, for the concurrent achievement of positional and pose target, plus the **C** module for the concurrent avoidance of joint limits. The integration of the three subtasks is implemented in a linear way, by adding up the corresponding torque fields in the joint space, whereas the synchronization is assured by the non-linear gating action of the Γ-command.

5.1. Block A_{pose}

The starting point is the definition of the target pose to be reached at the end of the pose reaching sub-task: in Fig. 9 it is expressed as the Rodrigues vector Π_T, which can be derived from the final desired rotation matrix by using Eq. (12) followed by Eq. (15). Also the moving pose target is represented by a Rodrigues vector $\rho_T(t)$: in analogy to the positional part of the $\mathbf{A_p}$ block the moving pose target is generated by integrating over time a virtual force field driven by the "pose error" $\Delta(t) = \Pi_T - \rho_T(t)$ and gated by the Γ-command according to the following equation:

$$\begin{cases} \Phi_T(t) = K_3\left(\Pi_T - \rho_T(t)\right) \\ \dot{\rho}_T = \frac{\Gamma(t)\Phi_T(t)}{B} \end{cases} \tag{16}$$

The initial value of the Rodrigues vector $\rho_T(t_0)$ is derived from the corresponding values of the joint angle vector $q(t_0)$ by applying Eq. (12) and then Eq. (15).

5.2. Block B_{pose}

The second part of the PMP mechanism for pose reaching (module $\mathbf{B_{pose}}$ of Fig. 9) integrates the force field generated by the difference between the moving target-pose and the current pose. This module, in analogy with the module $\mathbf{B_p}$, uses a Jacobian matrix for two purposes: i) mapping the 3D virtual force field Φ_ρ into the corresponding 7D field τ_{pose} and ii) mapping the joint velocity

vector $\dot{q}$ into the corresponding time derivative of the Rodrigues vector $\dot{\rho}_{ee}$. In summary, this block is described by the following equation, which is analogous to Eq. (7) of the module B_p:

$$\begin{cases} \Phi_\rho(t) = K_4\left(\rho_T(t) - \rho_{ee}(t)\right) \\ \tau_{pose}(t) = {J_\rho}^T \Phi_\rho(t) \end{cases} \tag{17}$$

As shown in Fig. 9, there is also the crucial, final step of combining the different torque fields (τ_p from the Module B_p, τ_{pose} from module B_{pose}, and τ_{RoM} from the Module $\boldsymbol{C}$), gating the combined field with the Γ-command and distributing the action to the different joints via the compliance matrix $\boldsymbol{C}$:

$$\begin{cases} \dot{q} = \Gamma C(t)\left[\tau_p(t) + \tau_{pose}(t) + \tau_{RoM}(t)\right] \\ \dot{\rho}_{ee} = J_\rho\ \dot{q} \end{cases} \tag{18}$$

This computational framework allows to further expand or specialize the set of additional constraints that may characterize a given task in a particular environment. In all the cases, a given constraint should be expressed as a torque field, using a corresponding Jacobian operator, and combined linearly in the joint space. Panel D of Fig. 9 shows the joint rotation patterns generated by the combined PMP model, clarifying the fact that inter-joint coordination is rather complex; panel E shows that both targets are reached simultaneously, with a smooth unimodal decrease of the distance (note that the distance from the positional target is expressed in meters whereas the distance from the pose target is expressed in degrees); panel F shows that such smooth coordination is obtained by a complex time course of the three torque fields.

The detailed calculation of both Jacobian matrices is reported in Appendix A2. As already mentioned, all the simulations presented in this paper (Figs. 2–4, 6–9) were carried out using Matlab® (MathWorks), with the forward Euler method for integrating the differential equations with a time step of 0.1 ms. In summary, the following parameter values were used:

- $K_1 = 1$: Gain for the generation of the moving positional target
- $K_2 = 50$: Gain for attracting the end effector to the moving positional target
- $K_3 = 1$: Gain for the generation of the moving pose target
- $K_4 = 15$: Gain for attracting the pose of the end effector to the moving pose target
- $K_{RoM} = 1$: Gain for the generation of the repulsive torque from the joint limits
- $\gamma = 50$: Normalization factor for the repulsive torque from the joint limits
- $C = I_7$: Compliance matrix for the distribution of effort to the DoFs.

Such values were found empirically and the robustness of the model was tested by verifying that the performance of the simulated models was hardly affected by a ±50% variation of the parameters.

6. Discussion and conclusions

In this paper, a bio-inspired model of gesture formation is proposed that bypasses the indeterminacy of the degrees of freedom problem by employing an attractive force field to the target: this internal perturbation is propagated to the redundant set of joints of the body schema, thus distributing a coordinated set of motion patterns, coherent with the goal of the action. We showed that this computational concept can be easily extended in many ways, in agreement with the general view about biological motion as the result of the internal simulation of the body model under the action of motor imagery [41].

In general, this is an approach that allows avoiding the introduction of ill-posed, inverse computational problems like inverse kinematics and inverse dynamics that typically require the use of optimal control formulations [42]. The force-field-based proposed approach, alternative to optimal control, is based on the idea that the internal simulation of covert actions is performed by the same neural mechanism normally involved in overt actions [43]. This approach is also linked to the still-influential tradition underlying the *Ideomotor Theory*, proposed by William James in the 19th century and recently revisited [44], namely the concept that the 'idea' of an action, i.e. the predicted/desired sensory consequences of covert action, applies as well to goal-directed overt actions and is the internal mechanism that ultimately generates it through the simulation of an internal model.

The robustness of this computational mechanism is that the internal simulation of the body model under the action of multiple virtual force fields is always possible, whatever the degree of redundancy of the body schema and the additional constraints that are specifically dependent on task and environment conditions. Even if the task is over-constrained and no solution exists, the internal simulation generates smooth movements and although the final state, achieved at the end of a gesture, maybe short of the planned goal, the error is informative from the cognitive point of view: it can be used for re-planning the course of action, for example by breaking down the overall action in a sequence of sub-actions, driven by appropriate sub-goals.

As regards the additional constraints or sub-tasks that may be integrated into the formulation of an action plan, in the previous section we addressed, in particular, the sub-task of avoiding the joint limits. Another multi-joint coordination synergy was investigated for whole-body reaching while standing [45], namely a double task that requires i) to reach the designated target and ii) to maintain the projection of the center of mass of the body inside the perimeter of the support base. In this paper we showed how to integrate positional reaching with pose reaching, using Euclidean and non-Euclidean metrics at the same time. The power of the approach is that the different sub-tasks are force-field based and the corresponding force fields may be defined in different reference spaces (e.g. joint space, hand positional space, hand pose space, environment space, tool space, etc.) and thus the proposed system of gesture formation is generally *multi-referential.* The integration of the sub-tasks is carried out in the joint space by mapping the task-related

force fields into the common joint space, through the corresponding Jacobian matrices, and then adding the different virtual torque patterns. While the integration of the sub-tasks is carried out by a linear superposition, their synchronization is obtained by the non-linear gating of the Γ-command.

One of the novelties of the proposed model is the performance of soft body gestures that involve both arms, with different positional and pose targets: such smoothness is characteristic of the elegant sequences of actions in Taichi Chuan, in contrast to the jerky movements of many industrial robotic systems that are quite inappropriate to inspire confidence in human partners instructed to cooperate with robots. In comparison with industrial robots, the current models of cobots present in the market have significantly reduced the operational speed, as a crucial requirement to improve safety. However, safety is a very general concept that implies general cognitive issues on both the human and robot side and is likely to take advantage of the present and future advances in artificial intelligence principles and applications [46–49].

However, cognitive robotic architectures for Human–Robot partnership are still in their infancy as is the software technology of present cobots which is mostly focused on ergonomics, in the sense of safety of the Human–Robot interaction and shared-effort collaboration for reducing work-related musculoskeletal disorders [50]. Human–Robot partnership requires the integration of the individual partner's intentions into a shared action plan, which may involve continuous negotiation of intentions based on a reciprocal understanding of actions: this implies not only channels of explicit/implicit communication, but also the capability to access and/or anticipate the partner's internal states and intentions during a cooperative task. In humans, this issue has been the subject of several recent studies also related to the meaning of biological motion [51–55]. In general, we may define an action as a goal-directed sequence of bodily movements initiated, monitored, and fine-tuned by what may be called a motor intention. More specifically, we should also consider the distinction [56] between basic actions, i.e., actions aimed at the achievement of an intermediate/immediate goal, and non-basic actions, whose goal is farther away in space and time. In any case, a module of trajectory/gesture formation that mimics the biological counterparts seems to be crucial for motor intentions, internal simulation of actions, and embodied cognition, including a mechanism of episodic memory [57,58].

Declaration of Competing Interest

The author declares that he has no known competing financial interest or personal relationship that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by Fondazione Istituto Italiano di Tecnologia, RBCS Department, in relation with the iCog Initiative.

Appendices

A1. Gamma function $\Gamma = (t; t_0, T)$

t_0: Initial time instant
T: Duration of the function
t: Time
ξ : Normalized time ($\xi = \frac{t-t_0}{T}$)

$$\Gamma = \frac{\dot{\xi}}{1-\xi} \begin{cases} \xi = 6\,\xi^5 - 15\xi^4 + 10\xi^3 \\ \dot{\xi} = 30\xi^4 - 60\xi^3 + 30\,\xi^2 \end{cases} for 0 < t - t_0 < T$$

$$\Gamma = 0 for\ 0 \geq\ t - t_0 \geq T \tag{A1}$$

A2. Explicit calculation of the Jacobian matrix related to the arm position

Fig. 7 shows the kinematic model of an anthropomorphic arm, with 7 DoFs: 3 at the shoulder (q_1, q_2, q_3), one at the elbow (q_4), and three at the wrist (q_5, q_6, q_7). The DoFs are represented according to the yaw/pitch/roll convention, that associates a specific rotation matrix to each joint:

$$q_1\ (shoulder\ yaw)\ R_1 = \begin{bmatrix} +c_1 & -s_1 & 0 \\ +s_1 & +c_1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{A2}$$

$$q_2\ (shoulder\ pitch)\ R_2 = \begin{bmatrix} +c_2 & 0 & +s_2 \\ 0 & 1 & 0 \\ -s_2 & 0 & +c_2 \end{bmatrix} \tag{A3}$$

$$q_3\ (shoulder\ roll)\ R_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & +c_3 & -s_3 \\ 0 & +s_3 & +c_3 \end{bmatrix} \tag{A4}$$

$$q_4\ (elbow\ roll)\ R_4 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & +c_4 & -s_4 \\ 0 & +s_4 & +c_4 \end{bmatrix} \tag{A5}$$

$$q_5\ (wrist\ yaw)\ R_5 = \begin{bmatrix} +c_5 & -s_5 & 0 \\ +s_5 & +c_5 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{A6}$$

$$q_6\ (wrist\ pitch)\ \ R_6 = \begin{bmatrix} +c_6 & 0 & +s_6 \\ 0 & 1 & 0 \\ -s_6 & 0 & +c_6 \end{bmatrix} \tag{A7}$$

$$q_7\ (wrist\ roll)\ R_7 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & +c_7 & -s_7 \\ 0 & +s_7 & +c_7 \end{bmatrix} \tag{A8}$$

For all the equations above $c_x = cos(q_x),\ s_x = sin(q_x)\ (x : 1 \dots 7)$.

The position of the end-effector P can then be written as follows:

$$\vec{P} = R_1 R_2 R_3 \left\{ \vec{V}^1_{1,2} + R_4 \left[\vec{V}^2_{2,3} + R_5 R_6 R_7 \vec{V}^3_{3,P} \right] \right\};\ \vec{V}^1_{1,2} = \left[0\ 0\ l_1\right]';\ \vec{V}^2_{2,3} = \left[0\ 0\ l_2\right]';\ \vec{V}^3_{3,P} = \left[0\ 0\ l_3\right]' \tag{A9}$$

where $l_1,\ l_2,\ l_3$ represent the length of the humerus, forearm, and hand, respectively. The Jacobian matrix can then be expressed as follows:

$$\boldsymbol{J}_p = \frac{\partial \vec{P}}{\partial \mathrm{q}} = \left[J_1 |\ J_2 | J_3 |\ J_4 | J_5 |\ J_6 |\ J_7\right] \tag{A10}$$

and we can derive the explicit expression of the seven columns of the matrix:

$$J_1 = \frac{\partial \vec{P}}{\partial \mathrm{q}_1} = \frac{\partial R_1}{\partial q_1} R_2 R_3 \left\{ \vec{V}^1_{1,2} + R_4 \left[\vec{V}^2_{2,3} + R_5 R_6 R_7 \vec{V}^3_{3,P} \right] \right\} \text{ with } \frac{\partial R_1}{\partial q_1} = \begin{bmatrix} -s_1 & -c_1 & 0 \\ +c_1 & -s_1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{A11}$$

$$J_2 = \frac{\partial \vec{P}}{\partial \mathrm{q}_2} = R_1 \frac{\partial R_2}{\partial q_2} R_3 \left\{ \vec{V}^1_{1,2} + R_4 \left[\vec{V}^2_{2,3} + R_5 R_6 R_7 \vec{V}^3_{3,P} \right] \right\} \text{ with } \frac{\partial R_2}{\partial q_2} = \begin{bmatrix} -s_2 & 0 & +c_2 \\ 0 & 0 & 0 \\ -c_2 & 0 & -s_2 \end{bmatrix} \tag{A12}$$

$$J_3 = \frac{\partial \vec{P}}{\partial \mathrm{q}_2} = R_1 R_2 \frac{\partial R_3}{\partial q_3} \left\{ \vec{V}^1_{1,2} + R_4 \left[\vec{V}^2_{2,3} + R_5 R_6 R_7 \vec{V}^3_{3,P} \right] \right\} \text{ with } \frac{\partial R_3}{\partial q_3} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & -s_3 & -c_3 \\ 0 & +c_3 & -s_3 \end{bmatrix} \tag{A13}$$

$$J_4 = \frac{\partial \vec{P}}{\partial \mathrm{q}_4} = R_1 R_2 R_3 \left\{ \vec{V}^1_{1,2} + \frac{\partial R_4}{\partial q_4} \left[\vec{V}^2_{2,3} + R_5 R_6 R_7 \vec{V}^3_{3,P} \right] \right\} \text{ with } \frac{\partial R_4}{\partial q_4} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & -s_4 & -c_4 \\ 0 & +c_4 & -s_4 \end{bmatrix} \tag{A14}$$

$$J_5 = \frac{\partial \vec{P}}{\partial \mathrm{q}_5} = R_1 R_2 R_3 \left\{ \vec{V}^1_{1,2} + R_4 \left[\vec{V}^2_{2,3} + \frac{\partial R_5}{\partial q_5} R_6 R_7 \vec{V}^3_{3,P} \right] \right\} \text{ with } \frac{\partial R_5}{\partial q_5} = \begin{bmatrix} -s_5 & -c_5 & 0 \\ +c_5 & -s_5 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{A15}$$

$$J_6 = \frac{\partial \vec{P}}{\partial \mathrm{q}_6} = R_1 R_2 R_3 \left\{ \vec{V}^1_{1,2} + R_4 \left[\vec{V}^2_{2,3} + R_5 \frac{\partial R_6}{\partial q_6} R_7 \vec{V}^3_{3,P} \right] \right\} \text{ with } \frac{\partial R_6}{\partial q_6} = \begin{bmatrix} -s_6 & 0 & +c_6 \\ 0 & 0 & 0 \\ -c_6 & 0 & -s_6 \end{bmatrix} \tag{A16}$$

$$J_7 = \frac{\partial \vec{P}}{\partial \mathrm{q}_7} = R_1 R_2 R_3 \left\{ \vec{V}^1_{1,2} + R_4 \left[\vec{V}^2_{2,3} + R_5 R_6 \frac{\partial R_7}{\partial q_7} \vec{V}^3_{3,P} \right] \right\} \text{ with } \frac{\partial R_7}{\partial q_7} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & -s_7 & -c_7 \\ 0 & +c_7 & -s_7 \end{bmatrix} \tag{A17}$$

A3. Neural network approximation of the Jacobian matrix related to the arm position

Both the Jacobian matrix J_p and its transpose, which are key elements of the block diagram of Fig. 7 for carrying out the internal simulation related to synergy formation, can be implemented with the exact, explicit formula of Eqs. (A10)–(A17), a solution that is rather biologically implausible but directly practical for robotic applications. An alternative is to approximate it with a neural network, trained with a dense training set obtained with a babbling procedure that includes the following steps: i) generation of pseudo-random motion sequences; ii) storage of proprioceptive-exteroceptive measurements, namely the joint rotation angles ($\vec{Q}(r),\ r = 1 \dots N$) and the corresponding positions of the end-effector ($\vec{P}(r),\ r = 1 \dots N$); iii) using the training set for tuning the connection weights of the network. Apart from biological plausibility, the neural approach can make sense also in the robotic case because it can compensate for model and sensory inaccuracies.

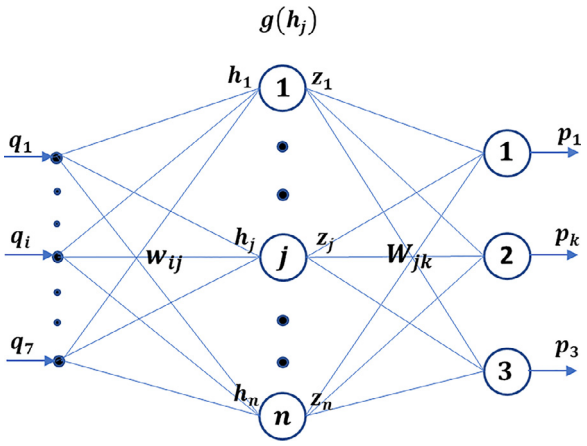

Fig. A1. Feedforward Neural Network, trained for implementing the forward kinematic transformation of the anthropomorphic arm of Fig. 7, mapping the input array of joint angle in the corresponding position of the end-effector in 3D space.

The simplest neural model is a feedforward network, with an input layer (with 7 units, one for each DoF), one or more hidden layers, and an output layer (with 3 units, for the three coordinates of the end-effector). More complex input or output coding schemes can be employed but this an implementation detail beyond the scope of this analysis. For simplicity, let us suppose that there is a single hidden layer and then we can write the equations of the ANN that approximates the direct kinematic equation from the array of joint angles $\vec{Q}$ to the position of the end-effector $\vec{P}$ as follows:

$$\begin{cases} h_j = \sum_{i=1}^{7} w_{ij}\, q_i \\ z_j = g(h_j) \\ p_k = \sum_{j=1}^{n} W_{jk}\, z_j \end{cases} \tag{A18}$$

where (q_i $i = 1 \ldots 7$) is the input vector and (p_k $k = 1, 2, 3$) the corresponding output vector of the ANN; n is the number of hidden neurons and, for each of them, h_j is the net input and z_j the output variable, respectively; w_{ij} & W_{jk} are the connection weights from input to hidden neurons and from the hidden to output neurons, respectively; the hidden neurons are non-linear and $g(.)$ is the input-output transformation, typically a sigmoid function; the output neurons are linear. Eq. (A18) corresponds to the block diagram of Fig. A1.

To clarify the figure, if we apply to the network, trained with the standard back-propagation algorithm, an input vector that corresponds to a given configuration q of the arm, the produced output will be a good approximation of the corresponding position p of the end-effector: in other words, the trained network represents the forward kinematic transformation of the robotic arm. Moreover, it is possible to demonstrate that the same network, with a small modification, can be used as well to represent both the Jacobian and transpose Jacobian matrices (J_p and ${J_p}^T$) that allow mapping generalized velocities ($\dot{p} = J_p\, \dot{q}$) or generalized forces ($\tau = {J_p}^T F$). To demonstrate this statement, let us apply the chain rule to the generic element of the Jacobian matrix as follows:

$$J_p = [J_{ki}] \;\rightarrow\; J_{ki} = \frac{\partial p_k}{\partial q_i} = \sum_{j=1}^{n} \frac{\partial p_k}{\partial z_j}\, \frac{\partial z_j}{\partial h_j}\, \frac{\partial h_j}{\partial q_i} = \sum_{j=1}^{n} W_{jk}\, g'(\hat{h}_j)\, w_{ij} \tag{A19}$$

For each neuron of the hidden layer, $g'(\hat{h}_j)$ is the slope of the sigmoid function, for the net input $\hat{h}_j$, determined by the current array $\hat{q}$ of joint angles: this slope, which represents the 'gain' of the jth hidden neuron, has a maximum value if the argument of the sigmoid (i.e. $\hat{h}_j$) is in the middle of the range and vanishes to zero for values away. The components of the velocity vector of the end-effector can then be calculated by the following equation:

$$\dot{p}_k = \sum_{i=1}^{7} J_{ki}\dot{q}_i = \sum_{i=1}^{7} \left(\sum_{j=1}^{n} W_{jk}\, g'(\hat{h}_j)\, w_{ij} \right) \dot{q}_i = \sum_{j=1}^{n} W_{jk}\, g'(\hat{h}_j) \left(\sum_{i=1}^{7} w_{ij}\, \dot{q}_i \right) \tag{A20}$$

This is equivalent to the network equation of the block diagram of Fig. A2-A, namely a network that uses the same connection weights of the network A1 with the difference that the 'gain' of the hidden neurons is not computed in relation with the input vector to network A2-A but is inherited from the network A1:

$$\begin{aligned} H_j &= \sum_{i=1}^{7} w_{ij}\, \dot{q}_i \\ \{ Z_j &= g'(\hat{h}_j)\, H_j \\ \dot{p}_k &= \sum_{j=1}^{n} W_{jk}\, Z_j \end{aligned} \tag{A21}$$

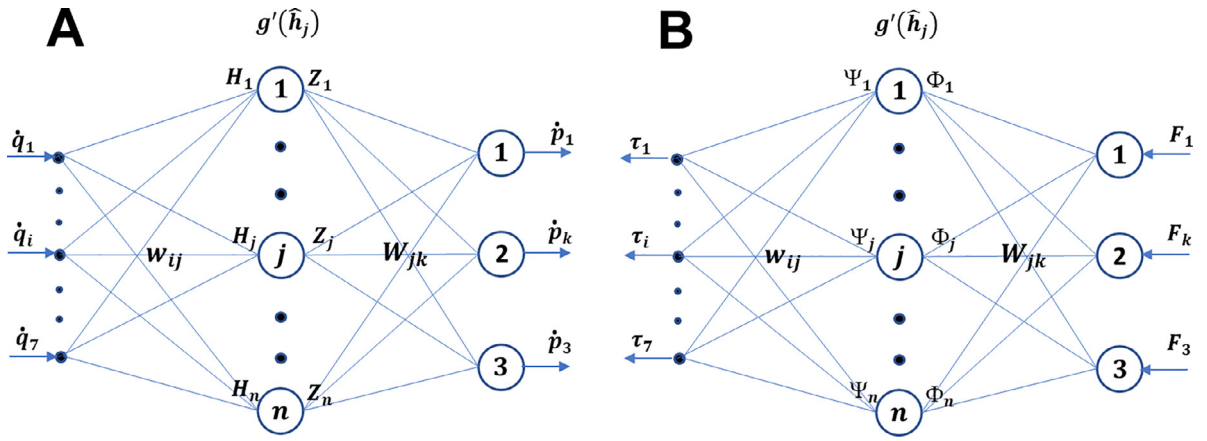

Fig. A2. Panel A: Use of the forward kinematic network of Fig. A1 for computing the Jacobian matrix that allows mapping the speed of the joint vector into the corresponding speed vector of the end-effector. Panel B: Use of the same network in the opposite direction, thus computing the transpose Jacobian matrix that allows mapping the force field applied to the end-effector into the corresponding torque field to be applied to the joints.

A similar result can be obtained as regards the mapping of the generalized forces $\tau = J_p{}^T F$ that reverses input and output vectors to the Jacobian operator:

$$\tau_i = \sum_{k=1}^{3} J_{ik} F_k = \sum_{k=1}^{3} \left(\sum_{j=1}^{n} w_{ji}\, g'(\hat{h}_j)\, W_{kj} \right) F_k = \sum_{j=1}^{n} w_{ji}\, g'(\hat{h}_j) \left(\sum_{k=1}^{3} W_{kj}\, F_k \right) \tag{A22}$$

This is equivalent to the network equation of the block diagram of Fig. A2-B:

$$\begin{cases} \Phi_j = \sum_{k=1}^{3} W_{kj}\, F_k \\ \Psi_j = g'(\hat{h}_j) \Phi_j \\ \tau_i = \sum_{j=1}^{n} w_{ij} \Psi_j \end{cases} \tag{A23}$$

Consider that this an unconventional use of a standard multilayer feedforward neural network, whose connection weights are meant to be strictly unidirectional. In this case, the weights are used bidirectionally: in the standard direction, for approximating the Jacobian matrix, and in the opposite direction for the transpose Jacobian. The equations above, analyzed for simplicity in the case of a single hidden layer, can be easily generalized to networks with multiple layers, similarly applying the chain rule. Such bidirectionality is the crucial trick for solving the degrees of freedom problem of a redundant anthropomorphic arm.

A4 Explicit calculation of the Jacobian matrix related to the arm pose

The pose of the end-effector is implicitly stored in the rotation matrix of the hand R_{ee}, related to the environment reference frame. It is a function of the 7 DoFs of the arm:

$$R_{ee} = [r_{ij}] = R_1(q_1) R_2(q_2) R_3(q_3) R_4(q_4) R_5(q_5) R_6(q_6) R_7(q_7) \tag{A24}$$

This matrix can also be represented in terms of the Rodrigues vector:

$$\rho = \begin{bmatrix} (r_{32} - r_{23})/(1 + r_{11} + r_{22} + r_{33}) \\ (r_{13} - r_{31})/(1 + r_{11} + r_{22} + r_{33}) \\ (r_{21} - r_{12})/(1 + r_{11} + r_{22} + r_{33}) \end{bmatrix} \tag{A25}$$

The pose Jacobian (a 3×7 matrix) is defined as follows:

$$J_\rho = \frac{\partial \rho}{\partial q} = \begin{bmatrix} J_1 \\ J_2 \\ J_3 \end{bmatrix} \tag{A26}$$

From Eq. (A25) it is possible to derive the three rows of the matrix:

$$\begin{cases} J_1 = \frac{\partial \rho_1}{\partial q} = \frac{\frac{\partial r_{32}}{\partial q} - \frac{\partial r_{23}}{\partial q}}{1 + r_{11} + r_{22} + r_{33}} - \frac{r_{32} - r_{23}}{(1 + r_{11} + r_{22} + r_{33})^2} \left(\frac{\partial r_{11}}{\partial q} + \frac{\partial r_{22}}{\partial q} + \frac{\partial r_{33}}{\partial q} \right) \\ J_2 = \frac{\partial \rho_2}{\partial q} = \frac{\frac{\partial r_{13}}{\partial q} - \frac{\partial r_{31}}{\partial q}}{1 + r_{11} + r_{22} + r_{33}} - \frac{r_{13} - r_{31}}{(1 + r_{11} + r_{22} + r_{33})^2} \left(\frac{\partial r_{11}}{\partial q} + \frac{\partial r_{22}}{\partial q} + \frac{\partial r_{33}}{\partial q} \right) \\ J_3 = \frac{\partial \rho_3}{\partial q} = \frac{\frac{\partial r_{21}}{\partial q} - \frac{\partial r_{12}}{\partial q}}{1 + r_{11} + r_{22} + r_{33}} - \frac{r_{21} - r_{12}}{(1 + r_{11} + r_{22} + r_{33})^2} \left(\frac{\partial r_{11}}{\partial q} + \frac{\partial r_{22}}{\partial q} + \frac{\partial r_{33}}{\partial q} \right) \end{cases} \tag{A27}$$

Let us consider that each row is a 7-element array:

$$\begin{cases} J_1 = \left[\frac{\partial \rho_1}{\partial q_1} \; \frac{\partial \rho_1}{\partial q_2} \; \frac{\partial \rho_1}{\partial q_3} \; \frac{\partial \rho_1}{\partial q_4} \; \frac{\partial \rho_1}{\partial q_5} \; \frac{\partial \rho_1}{\partial q_6} \; \frac{\partial \rho_1}{\partial q_7}\right] \\ J_2 = \left[\frac{\partial \rho_2}{\partial q_1} \; \frac{\partial \rho_2}{\partial q_2} \; \frac{\partial \rho_2}{\partial q_3} \; \frac{\partial \rho_2}{\partial q_4} \; \frac{\partial \rho_2}{\partial q_5} \; \frac{\partial \rho_2}{\partial q_6} \; \frac{\partial \rho_2}{\partial q_7}\right] \\ J_3 = \left[\frac{\partial \rho_3}{\partial q_1} \; \frac{\partial \rho_3}{\partial q_2} \; \frac{\partial \rho_3}{\partial q_3} \; \frac{\partial \rho_3}{\partial q_4} \; \frac{\partial \rho_3}{\partial q_5} \; \frac{\partial \rho_3}{\partial q_6} \; \frac{\partial \rho_3}{\partial q_7}\right] \end{cases} \tag{A28}$$

Defining the following variable, which is a function of the current angular configuration

$$D(q) = 1 + r_{11} + r_{22} + r_{33} \tag{A29}$$

we can express the generic elements of Eq. (A28) in the following manner:

$$\begin{cases} \frac{\partial \rho_1}{\partial q_k} = \frac{1}{D}\left[\frac{\partial r_{32}}{\partial q_k} - \frac{\partial r_{23}}{\partial q_k}\right] - \frac{r_{32}-r_{23}}{D^2}\left[\frac{\partial r_{11}}{\partial q_k} + \frac{\partial r_{22}}{\partial q_k} + \frac{\partial r_{33}}{\partial q_k}\right] \\ \frac{\partial \rho_2}{\partial q_k} = \frac{1}{D}\left[\frac{\partial r_{13}}{\partial q_k} - \frac{\partial r_{31}}{\partial q_k}\right] - \frac{r_{13}-r_{31}}{D^2}\left[\frac{\partial r_{11}}{\partial q_k} + \frac{\partial r_{22}}{\partial q_k} + \frac{\partial r_{33}}{\partial q_k}\right] \\ \frac{\partial \rho_3}{\partial q_k} = \frac{1}{D}\left[\frac{\partial r_{21}}{\partial q_k} - \frac{\partial r_{12}}{\partial q_k}\right] - \frac{r_{21}-r_{12}}{D^2}\left[\frac{\partial r_{11}}{\partial q_k} + \frac{\partial r_{22}}{\partial q_k} + \frac{\partial r_{33}}{\partial q_k}\right] \end{cases} \tag{A30}$$

Thus, in order to compute the Jacobian matrix we need to know the nine elements r_{ij} that can be simply extracted from R_{ee} (Eq. (A24)) and the following $3 \times 3 \times 7$=63 elements:

$$\frac{\partial r_{ij}}{\partial q_k} = \left[\frac{\partial R}{\partial q_k}\right]_{ij} \tag{A31}$$

which expresses the ij component of a matrix $S_k = \frac{\partial R}{\partial q_k}$, one for each DoF of the arm. These matrices can be computed as follows:

$$S_1(q) = \frac{\partial R}{\partial q_1} = \frac{\partial R_1}{\partial q_1} R_2 \; R_3 \; R_4 \; R_5 \; R_6 \; R_7 \text{ with } \frac{\partial R_1}{\partial q_1} = \begin{bmatrix} -s_1 & -c_1 & 0 \\ +c_1 & -s_1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{A32}$$

$$S_2(q) = \frac{\partial R}{\partial q_2} = R_1 \frac{\partial R_2}{\partial q_2} \; R_3 \; R_4 \; R_5 \; R_6 \; R_7 \text{ with } \frac{\partial R_2}{\partial q_2} = \begin{bmatrix} -s_2 & 0 & +c_2 \\ 0 & 0 & 0 \\ -c_2 & 0 & -s_2 \end{bmatrix} \tag{A33}$$

$$S_3(q) = \frac{\partial R}{\partial q_3} = R_1 \; R_2 \; \frac{\partial R_3}{\partial q_3} R_4 \; R_5 \; R_6 \; R_7 \text{ with } \frac{\partial R_3}{\partial q_3} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & -s_3 & -c_3 \\ 0 & +c_3 & -s_3 \end{bmatrix} \tag{A34}$$

$$S_4(q) = \frac{\partial R}{\partial q_4} = R_1 \; R_2 \; R_3 \frac{\partial R_4}{\partial q_4} \; R_5 \; R_6 \; R_7 \text{ with } \frac{\partial R_4}{\partial q_4} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & -s_4 & -c_4 \\ 0 & +c_4 & -s_4 \end{bmatrix} \tag{A35}$$

$$S_5(q) = \frac{\partial R}{\partial q_5} = R_1 \; R_2 \; R_3 R_4 \frac{\partial R_5}{\partial q_5} \; R_6 \; R_7 \text{ with } \frac{\partial R_5}{\partial q_5} = \begin{bmatrix} -s_5 & -c_5 & 0 \\ +c_5 & -s_5 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{A36}$$

$$S_6(q) = \frac{\partial R}{\partial q_4} = R_1 \; R_2 \; R_3 \; R_4 \; R_5 \; \frac{\partial R_6}{\partial q_6} R_7 \text{ with } \frac{\partial R_6}{\partial q_6} = \begin{bmatrix} -s_6 & 0 & +c_6 \\ 0 & 0 & 0 \\ -c_6 & 0 & -s_6 \end{bmatrix} \tag{A37}$$

$$S_7(q) = \frac{\partial R}{\partial q_7} = R_1 \; R_2 \; R_3 \; R_4 R_5 \; R_6 \; \frac{\partial R_7}{\partial q_7} \text{ with } \frac{\partial R_7}{\partial q_7} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & -s_7 & -c_7 \\ 0 & +c_7 & -s_7 \end{bmatrix} \tag{A38}$$

This completes the explicit algorithm for the real-time computation of the pose Jacobian and transpose Jacobian matrices, used by the block diagram of Fig. 9. Also in this case it is possible to approximate the exact calculation of the Jacobian matrices by means of a neural network approximation, with a network trained by means of back propagation.

References

[1] J. Kotseruba, J.K. Tsotsos, A Review of 40 Years in Cognitive Architecture Research Core Cognitive Abilities and Practical Applications, Artif. Intell. Rev. 53 (2020) 17–94.
[2] V. Braitenberg, Vehicles: Experiments in Synthetic Psychology, MIT Press, Cambridge MA, 1984.
[3] R. Brooks, Intelligence without representation, Artif. Intell. J. 47 (1991) 139–159.
[4] A.P. Chemero, Radical Embodied Cognitive Science, MIT Press, Cambridge MA, 2009.
[5] J.E. Laird, The Soar Cognitive Architecture, MIT Press, 2012.
[6] F.E. Ritter, F. Tehranchi, J.D. Oury, ACT-R: A cognitive architecture for modeling cognition, Wiley Interdiscip. Rev. Cogn. Sci. 10 (3) (2019) e1488.
[7] M. Beetz, L. Mösenlechner, M. Tenorth, CRAM — A Cognitive Robot Abstract Machine for everyday manipulation in human environments, in: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Taipei, TW, 2010, p. 2010.
[8] J.L. Krichmar, Neurorobotics—a thriving community and a promising pathway toward intelligent cognitive robots, Front. Neurorobot. 12 (2018) 42.
[9] S. Nolfi, D. Floreano, Evolutionary Robotics, The Biology, Intelligence, and Technology of Self-Organizing Machines, The MIT Press, Cambridge, MA, 2000.
[10] R. Pfeifer, J. Bongard, How the Body Shapes the Way We Think: A New View of Intelligence, The MIT Press, Cambridge, MA, 2006.
[11] G. Sandini, V. Mohan, A. Sciutti, P. Morasso, Social cognition for Human–Robot symbiosis - challenges and building blocks, Front. Neurorobotics 12 (2018) 34.
[12] P. Morasso, V. Mohan, The body schema: neural simulation for covert and overt actions of embodied cognitive agents, Curr. Opin. Physiol. 21 (2021) 219–225.
[13] M. Jeannerod, Neural simulation of action: a unifying mechanism for motor cognition, Neuroimage 14 (1) (2001) S103–S109.
[14] R. Grush, The emulation theory of representation: motor control, imagery, and perception, Behav. Brain Sci. 27 (3) (2004) 377–396.
[15] G. Hesslow, The current status of the simulation theory of cognition, Brain Res. 1428 (2012) 71–79.
[16] R. Ptak, A. Schnider, J. Fellrath, The dorsal frontoparietal network: a core system for emulated action, Trends Cogn. Sci. 21 (8) (2017) 589–599.
[17] N. Bernstein, The Co-ordination and Regulation of Movements, Pergamon Press, Oxford, 1967.
[18] K.S. Lashley, Integrative function of the cerebral cortex, Physiol. Rev. 13 (1) (1933) 1–42.
[19] P. Morasso, Spatial control of arm movements, Exp. Brain Res. 42 (1981) 223–227.
[20] P. Morasso, F.A. Mussa Ivaldi, Trajectory formation and handwriting: a computational model, Biol. Cybern. 45 (1982) 131–142.
[21] P. Morasso, Three-dimensional arm trajectories, Biol. Cybern. 48 (1983) 187–19.
[22] F. Lacquaniti, C. Terzuolo, P. Viviani, The law relating the kinematic and figural aspects of drawing movements, Acta Psychol. 54 (1983) 115–130.
[23] T. Flash, N. Hogan, The coordination of arm movements: an experimentally confirmed mathematical model, J. Neurosci. 5 (7) (1985) 1688–1703.
[24] D. Bullock, S. Grossberg, Neural dynamics of planned arm movements: emergent invariants and speed-accuracy properties during trajectory formation, Psychol. Rev. 95 (1) (1988) 49–90.
[25] F.A. Mussa Ivaldi, P. Morasso, R. Zaccaria, Kinematic networks, A distributed model for representing and regularizing motor redundancy, Biol. Cybern. 60 (1988) 1–16.
[26] K. Friston, J. Mattout, J. Kilner, Action understanding and active inference, Biol- Cybern. 104 (2011) 137–160.
[27] A.G. Feldman, AG: Once more on the equilibrium hypothesis (lambda model) for motor control, J. Mot. Behav. 18 (1986) 17–54.
[28] E. Bizzi, N. Accomero, W. Chapple, N. Hogan, Posture control and trajectory formation during arm movement, J. Neurosci. 41 (1984) 2738–2745.
[29] E. Bizzi, N. Hogan, F.A. Mussa-Ivaldi, S. Giszter, Does the nervous system use equilibrium-point control to guide single and multiple joint movements? Behav. Brain Sci. 15 (4) (1992) 603–613.
[30] J. Decety, M. Jeannerod, Mentally simulated movements in virtual reality: does Fitt's law hold in motor imagery? Behav. Brain Res. 72 (1-2) (1995) 127–134.
[31] M. Karklinsky, T. Flash, Timing of continuous motor imagery: the two-thirds power law originates in trajectory planning, J. Neurophysiol. 113 (7) (2015) 2490–2499.
[32] M. Zak, Terminal attractors for addressable memory in neural networks, Phys. Lett. 133 (1988) 218–222.
[33] J. Barhen, S. J., S. Gulati, M. Zak, Neural learning of constrained nonlinear transformations, IEEE Comput. 6 (1989) 67–76.
[34] D. Mücke, M. Grice, T.Chob T, More than a magic moment – Paving the way for dynamics of articulation and prosodic structure, J. Phon. 44 (2014) 1–7.
[35] T. Flash, The control of hand equilibrium trajectories in multi joint arm movements, Biol. Cybern. 57 (1987) 257–274.
[36] R. Shadmeher, F.A. Mussa Ivaldi, E. Bizzi, Postural force fields of the human arm and their role in generating multijoint movements, The J. Neurosci. 13 (1) (1993) 45–82.
[37] J.P. Scholz, G. Schöner, The uncontrolled manifold concept: identifying control variables for a functional task, Exp. Brain Res. 126 (1999) 289–306.
[38] E. Piña, Rotations with Rodrigues' vector, Eur. J. Phys. 32 (2011) 1171–1178.
[39] J.S. Dai, Euler–Rodrigues formula variations, quaternion conjugation and intrinsic connections, Mech. Mach. Theory 92 (2015) 144–152.
[40] O. Rodrigues, O 1840 Des lois géométriques qui regissent les déplacéments d'un systéme solide dans l'espace, et de la variation des coordonées provenant de ses déplacéments considérés indépendamment des causes qui peuvent les produire, J. Math. Pures Appl. 5 (1840) 380–440.
[41] M. Jeannerod, The representing brain: neural correlates of motor intention and imagery, Behav. Brain Sci. 17 (2) (1994) 187–202.
[42] V. Mohan, P. Morasso, Passive Motion Paradigm: an alternative to optimal control, Front. Neurorobotics 5 (4) (2011) 1–28.
[43] V. Mohan, A. Bhat, P. Morasso, Muscleless Motor synergies and actions without movements: from motor neuroscience to cognitive robotics, Phys. Life Rev. 30 (2019) 89–111.
[44] Y.K. Shin, R.W. Proctor, E.J. Capaldi, A review of contemporary ideomotor theory, Psychol. Bull. 136 (6) (2010) 943–974.
[45] P. Morasso, M. Casadio, V. Mohan, J. Zenzeri, A neural mechanism of synergy formation for whole body reaching, Biol. Cybern. 102 (2010) 45–55.
[46] H. Lu, Q. Liu, D. Tian, Y. Li, H. Kim, S. Serikawa, The cognitive internet of vehicles for autonomous driving, IEEE Netw. 33 (3) (2018) 65–73.
[47] H. Lu, Y. Li, S. Mu, D. Wang, H. Kim, S. Serikawa, Motor anomaly detection for unmanned aerial vehicles using reinforcement learning, IEEE Internet of Things J. 5 (4) (2018) 2315–2322.
[48] H. Lu, M. Zhang, X. Xu, Y. Li, H. Tao Shen, Deep fuzzy hashing network for efficient image retrieval, IEEE Trans. Fuzzy Syst. 29 (1) (2020) 166–176.
[49] H. Lu, R. Yang, Z. Deng, Y. Zhang, G. Gao, R. Lan, Chinese image captioning via fuzzy attention-based DenseNet-BiLSTM, ACM Trans. Multimed. Comput. Commun. Appl. 17 (1s) (2021) 1–18.
[50] W. Kim, L. Peternel, M. Lorenzini, J. Babic, A. Ajoudani, A Human–Robot collaboration framework for improving ergonomics during dexterous operation of power tools, Robot. Comput.-Integr. 68 (2021) 102084.
[51] C. Becchio, L. Sartori, M. Bulgheroni, U. Castiello, Both your intention and mine are reflected in the kinematics of my reach-to-grasp movement, Cognition 106 (2) (2008) 894–912.
[52] V. Manera, B. Schouten, C. Becchio, et al., Inferring intentions from biological motion: a stimulus set of point-light communicative interactions, Behavior Res. Methods 42 (1) (2010) 168–178.
[53] A. Sciutti, C. Ansuini, C. Becchio, G. Sandini, Investigating the ability to read others' intentions using humanoid robots, Front. Psychol. 6 (2015) 1362.
[54] A. Vignolo, F. Rea, N. Noceti, et al., Biological movement detector enhances the attentive skills of humanoid robot iCub, in: Proceedings of the (2016) IEEE-RAS 16th International Conference on Humanoid Robots, 2016.
[55] T.S. Grafton, Embodied cognition and the simulation of action to understand others, Ann. N.Y. Acad. Sci. 1156 (2009) 97–117.
[56] E. Pacherie, The content of intentions, Mind Lang. 15 (2000) 400–432.
[57] M.P. Shanahan, A cognitive architecture that combines internal simulation with a global workspace, Conscious Cogn. 15 (2) (2006) 433–449.
[58] D. Vernon, M. Beetz, G.Sandini G., Prospection in cognitive robotics: The case for joint episodic-procedural memory, Front. Robot. A.I 2 (2015) (2015) 19.