Hierarchical Control of Visually-Guided Movements in a 3D-Printed Robot Arm

Adam Matić[1]*, Pavle Valerjev[2] and Alex Gomez-Marin[1]

[1] Behavior of Organisms Laboratory, Instituto de Neurociencias CSIC-UMH, Alicante, Spain, [2] Department of Psychology, University of Zadar, Zadar, Croatia

The control architecture guiding simple movements such as reaching toward a visual target remains an open problem. The nervous system needs to integrate different sensory modalities and coordinate multiple degrees of freedom in the human arm to achieve that goal. The challenge increases due to noise and transport delays in neural signals, non-linear and fatigable muscles as actuators, and unpredictable environmental disturbances. Here we examined the capabilities of hierarchical feedback control models proposed by W. T. Powers, so far only tested *in silico*. We built a robot arm system with four degrees of freedom, including a visual system for locating the planar position of the hand, joint angle proprioception, and pressure sensing in one point of contact. We subjected the robot to various human-inspired reaching and tracking tasks and found features of biological movement, such as isochrony and bell-shaped velocity profiles in straight-line movements, and the speed-curvature power law in curved movements. These behavioral properties emerge without trajectory planning or explicit optimization algorithms. We then applied static structural perturbations to the robot: we blocked the wrist joint, tilted the writing surface, extended the hand with a tool, and rotated the visual system. For all of them, we found that the arm *in machina* adapts its behavior without being reprogrammed. In sum, while limited in speed and precision (by the nature of the do-it-yourself inexpensive components we used to build the robot from scratch), when faced with the noise, delays, non-linearities, and unpredictable disturbances of the real world, the embodied control architecture shown here balances biological realism with design simplicity.

Keywords: robot arm, perceptual control theory, reaching, tracking, human movement

INTRODUCTION

Pointing and reaching toward visual targets are nearly effortless human behaviors. However, an explanation of these processes at levels of detail and abstraction that would allow us to build equally capable artificial systems or to treat common disorders in hand and arm control remains elusive. An understanding of such simple motor behaviors should follow from a broader theory of sensorimotor control, while being consistent with the anatomical structure of the underlying system. Such an understanding would provide insights into the origin of laws, invariances, and principles in the behavior of organisms.

The anatomical structure of the nervous system, together with the behavioral analysis of organisms under different conditions and upon different perturbations, suggests that biological control is hierarchical. One of the earliest hypotheses on the hierarchical nature of the nervous system was proposed by Hughlings Jackson (1884, 1958), discussing the possible evolutionary development of the nervous system in layers (see also Prescott et al., 1999). In turn, neurophysiologist Nikolai Bernstein proposed a hierarchical organization of neural structures underlying movement, where each layer performs a specific function, increasing in abstraction as one ascends the hierarchy (see Profeta and Turvey, 2018). Arguments for the existence of hierarchy of control can also be made from the comparative evolutionary history of the nervous system (Cisek, 2019) and from early development in primates (Plooij and van de Rijt-Plooij, 1990).

However, some findings in spinalized preparations blur the line between the capabilities of different levels in motor control. Cats with a transected spinal cord or cats in a decerebrate preparation can learn to walk on a treadmill (Whelan, 1996); decerebrate ferrets can learn new locomotion trajectories (Lou and Bloedel, 1988, 1992); rats with lesions in the motor cortex can still move in stable, predictable, non-perturbing environments, but not if the environment is rapidly changing (Lopes, 2016). Those experiments show the existence of independent "lower levels" in the spinal cord, capable of relatively complex behaviors on its own, despite normally operating in accord with the higher levels. Therefore, while the consensus seems to be that biological control is hierarchical, it is still unclear what is the function of each particular level, what are their limits and relationship, or even why there is a hierarchy at all.

Hierarchical architectures have been used in robotics, famously by Brooks in the subsumption architecture (Brooks, 1986). More recently, Merel et al. (2019) listed core advantages of hierarchical control appearing in both biological and engineered systems. Hierarchies allow for modularization and simplification of individual controllers and training procedures. Each subsystem can deal with only a part of the incoming sensory information, and, having partial autonomy, can be trained separately with cost functions and performance requirements distinct from the task objective. In contrast, "flat" non-hierarchical controllers receive and process all the sensory information, and directly calculate the behavioral output. Such arrangements make the control algorithm complex, require extensive training, and result in rather incomprehensible information flows. Thus, at least from an engineering perspective, hierarchical architectures can be very beneficial for adaptive behavioral control.

A prominent normative approach to motor behavior is optimal feedback control (Todorov and Jordan, 2002a; Scott, 2004, 2012; Shadmehr and Krakauer, 2008). The theory predicts many of the features of human movement and corrects a long-standing bias against the importance of sensory feedback in online movement (e.g., Flash and Hogan, 1985; Uno et al., 1989). However, this theoretical framework does not necessarily suggest a neural substrate for implementation of the proposed control algorithms. In fact, it is still debated whether some features of the optimal feedback control architecture,

such as internal forward and inverse models are necessary, computationally too complex, and whether they can be found in the brain or not (Loeb, 2012; McNamee and Wolpert, 2019; Hadjiosif et al., 2021). Briefly, forward models estimate the current state from a copy of the motor command and the delayed sensory signals, while the inverse models (also called controllers) provide a motor command that will achieve the desired state given the current state and an inverted model of the plant (Wolpert and Kawato, 1998). The theory does not explicitly address the hierarchical structure of motor control or the role of sensory feedback in subcortical levels and the spinal cord.

Exploring the computational principles that underlie eye-hand coordination and synergistic control in pointing and reaching, William T. Powers designed a series of distinct models of arm control. The first one (Powers, 1999) contained a model of muscles, an arm with three degrees of freedom (DOF) and a binocular vision control layer. The organization of this control system follows roughly the anatomical hierarchical organization of the spinal and some supra-spinal neural structures involved in human motor control and arm coordination. The second model (Powers, 2008) consisted of a 14 DOF arm with more fidelity in arm segment lengths and joint movement limits, but it lacked the muscle model. These models were built by cascading multiple layers of simple proportional and proportional-derivative feedback loops with low-pass filtering. Using hierarchically arranged controllers, and a selection of biologically-inspired controlled variables, Powers managed to avoid computationally expensive calculations of inverse models, unreliable estimates of load properties, and even inverse kinematics.

However, the behavioral capabilities of those models have not been assessed beyond the ideal world of numerical simulations. Our aim here is to test the *in silico* idea *in machina*, namely, to run those arm simulations in a robot arm, thus assessing not only the feasibility of the proposal in controlling a robot arm, but its biological realism in the context of human movement.

Why is this necessary and important? As with many simulations, Powers' arm models contain idealized representations of the nervous system, the body, and the environment. To name a few: there is no friction, no noise, no realistic transport delays (although there are some delays), no contact forces, etc. These idealizations are acceptable for initial testing and demonstration of principles, but as argued by Webb (2001), models of biological structures should be tested in terms of real problems faced by real organisms in the real world. Additionally, as claimed by Brooks (1992), there is a near certainty that programs that work well on simulated robots completely fail in real robots because of the differences between simulated and real-world sensing and actuation. Moreover, designing and building robots that work decently can generate insights about the function of structures in the nervous system that produce analogous behaviors in living organisms (Floreano et al., 2014; Morimoto and Kawato, 2015).
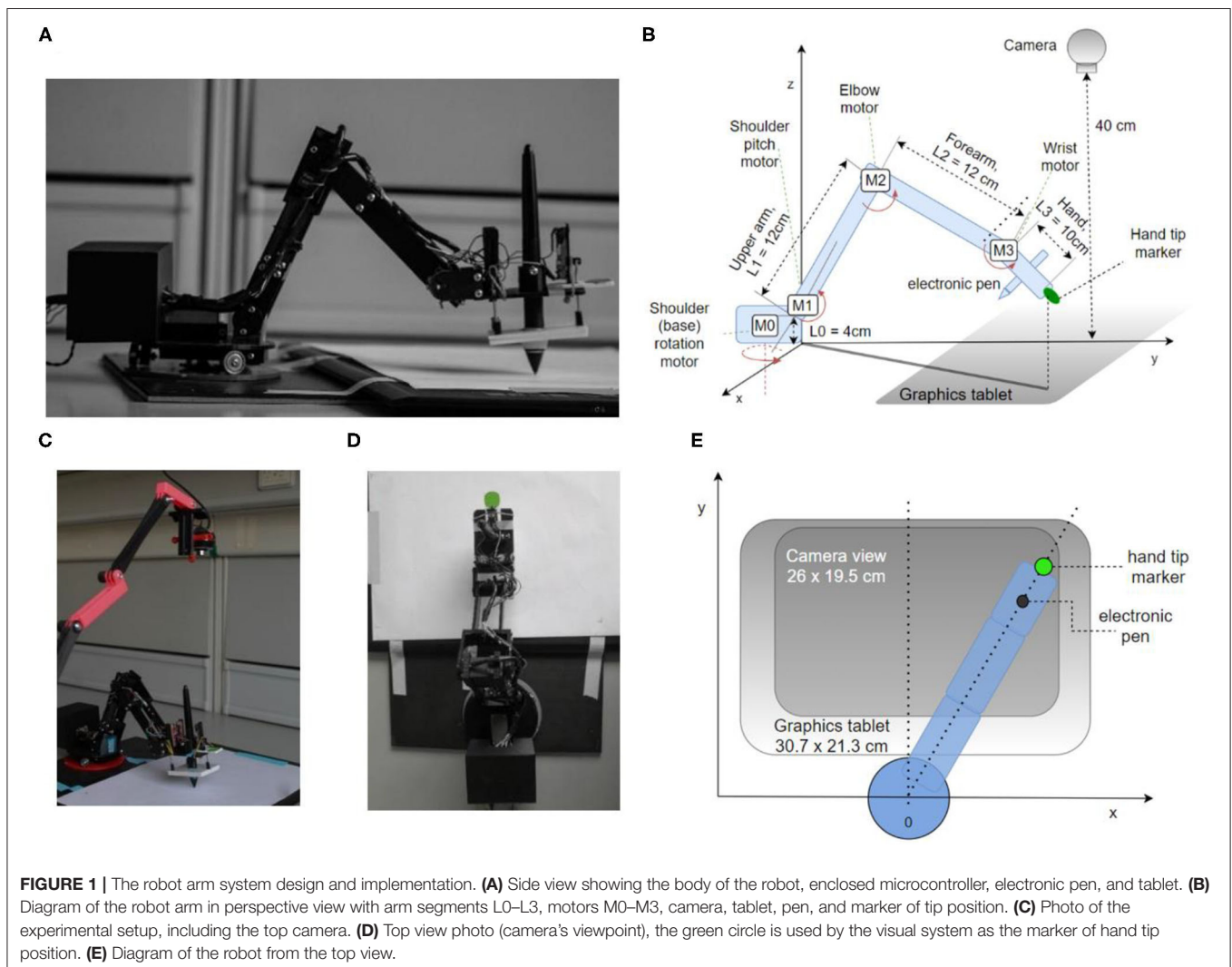
In sum, following this approach, in the present work we adapted and implemented the proposed hierarchical control architecture (Powers, 1999, 2008) to a 4 DOF robot arm in

order to examine its theoretical capabilities in the real world—dealing with noise, delays, non-linearities, and unpredictable environmental disturbances; as well as to generate insights about human control in basic task such as reaching or tracking. In the first part of the manuscript, we show that several fundamental invariant properties found in human hand trajectories—isochrony, bell-shaped velocity profiles and the speed-curvature power law—are also found in the robot arm trajectories without planning or optimization. In the second part, we demonstrate the motor equivalence phenomenon, where the robot arm can still perform reaching and tracking while the wrist joint is blocked, without learning or being reprogrammed. We also show spontaneous behavioral adaptation to the tilt of the writing surface, to the rotation of the visual field with respect to the arm segments, and to the extension of the robot hand with a "tool." We conclude by discussing the limitations of both the robot and its control architecture, specifically in the light of modeling fidelity and potentially higher and lower levels of control.

METHODS

Hardware: The Robot Arm

We designed and 3D-printed robot arm segments and its rotating base in PLA plastics. Several pictures of the arm and its diagram are shown in **Figure 1**. The robot has four degrees of freedom: shoulder rotation angle, shoulder pitch, elbow pitch, and wrist pitch. They are actuated via DC motors M0–M3, respectively (**Figure 1B**). The location of the shoulder pitch joint is 4 cm above the base level. The upper and lower arms are both 12 cm in length, while the hand spans 10 cm from the wrist joint to the hand tip. Each joint has a geared DC motor as the actuator and a potentiometer as a sensor to estimate the joint angle and angular velocity. The motors and gear trains come from RC servos: two HobbyKing 15298 in base and shoulder joints, Futaba S3003 in the elbow, and an N20 DC motor in the wrist. All the electronics and control circuits were removed from the servo motors and replaced by custom control software on the microcontroller. The potentiometers on the gear output shaft of the servos were kept to measure the angular position of the joints.

FIGURE 1 | The robot arm system design and implementation. **(A)** Side view showing the body of the robot, enclosed microcontroller, electronic pen, and tablet. **(B)** Diagram of the robot arm in perspective view with arm segments L0–L3, motors M0–M3, camera, tablet, pen, and marker of tip position. **(C)** Photo of the experimental setup, including the top camera. **(D)** Top view photo (camera's viewpoint), the green circle is used by the visual system as the marker of hand tip position. **(E)** Diagram of the robot from the top view.

The microcontroller used was Teensy 3.1, with a Cortex-M4 processor working at 96 Mhz and 3.3 V logic. It was programmed in C++ in Arduino IDE 1.8.6. It outputs four pulse-width modulated (PWM) signals to two TB6612FNG dual H-bridge 1A motor drivers, each connected to two motors and a 9 V 1A power supply, limited to 5 V in software. The sampling rate for angle potentiometers and control signal calculation rate on the microcontroller was 200 Hz.

To detect and measure the marker position, we used a generic webcam with a resolution of 640 × 480 px and a maximum frame rate of 30 Hz. The camera was placed at a height of ∼40 cm from the writing surface using a 3D printed stand (**Figure 1C**) The camera was pointed down toward the marker on the tip of the hand, covering the area of 26 × 19.5 cm, slightly smaller than the graphics tablet active area size (**Figures 1D,E**). The image from the camera was used to construct two controlled variables, the x and y position of the marker, and formed the basis of visual controlled loops.

To measure pen angle and pressure, we used a graphics tablet Wacom Intuos Pro Paper PTH-860, with an active surface of 30.7 × 21.3 cm, at a spatial resolution of 0.08 mm. The sampling rate of the tablet was 120 samples per second. However, we used 30 samples per second for pen pressure and pen angle control in order to be synchronized with the visual control loops that were limited by the temporal resolution of the camera to 30 Hz. The position of the pen as measured by the tablet itself was not used in arm control.

The PC we used for recording and visual processing had an Intel i5 processor, 8 GB of RAM, and runs on Windows 10 OS.

We initially designed and placed pressure sensors on the hand of the robot, using three linear sliding potentiometers measuring the stretch of an elastic rubber band when the hand is pressing on a surface (visible on the hand in **Figure 1A**). One sensor was placed at the tip of the hand, and two on the base. The sum of travel of all three potentiometers was therefore directly related to the pressure of the palm on a surface, and the difference between the front potentiometer and two back potentiometers was related to the pressure difference and tilt of the hand. However, the Wacom graphics tablet also reports the pressure of the pen on the tablet and the angle of pen tilt, and these readings proved to be more reliable than our custom sensors and were used in control loops.

Software and the Control Architecture
Hierarchical Control

In the arm models, Powers (1999, 2008) provides several simulations of hierarchical control architectures. However, those control systems are not proposals for the exact architecture of the human arm control systems, but rather conceptual models and demonstrations of principles. There are no rigorous rules or recipes given for construction of hierarchical systems. In his 1973 book, Powers proposes that the first, lowest-level control systems are in direct contact with the environment via receptors and effectors, forming fast negative feedback loops, with short transport delays. The reference values for the first level are supplied by the second level in the hierarchy, as outputs of a single control system or functions of outputs of multiple second-level control systems. The controlled variables of the second level in the hierarchy are constructed as functions of first-level controlled variables, have a different *level of abstraction,* and work with a longer signal transport delay. The next level up then continues the pattern of creating controlled variables as functions of lower-level controlled variables and provides the reference signals for the lower level. An example of the lowest level would be spinal loops controlling tendon tensions and muscle lengths, while supra-spinal or cortical loops would implement control of slower-changing, more abstract variables such as arm configurations of sequences of positions.
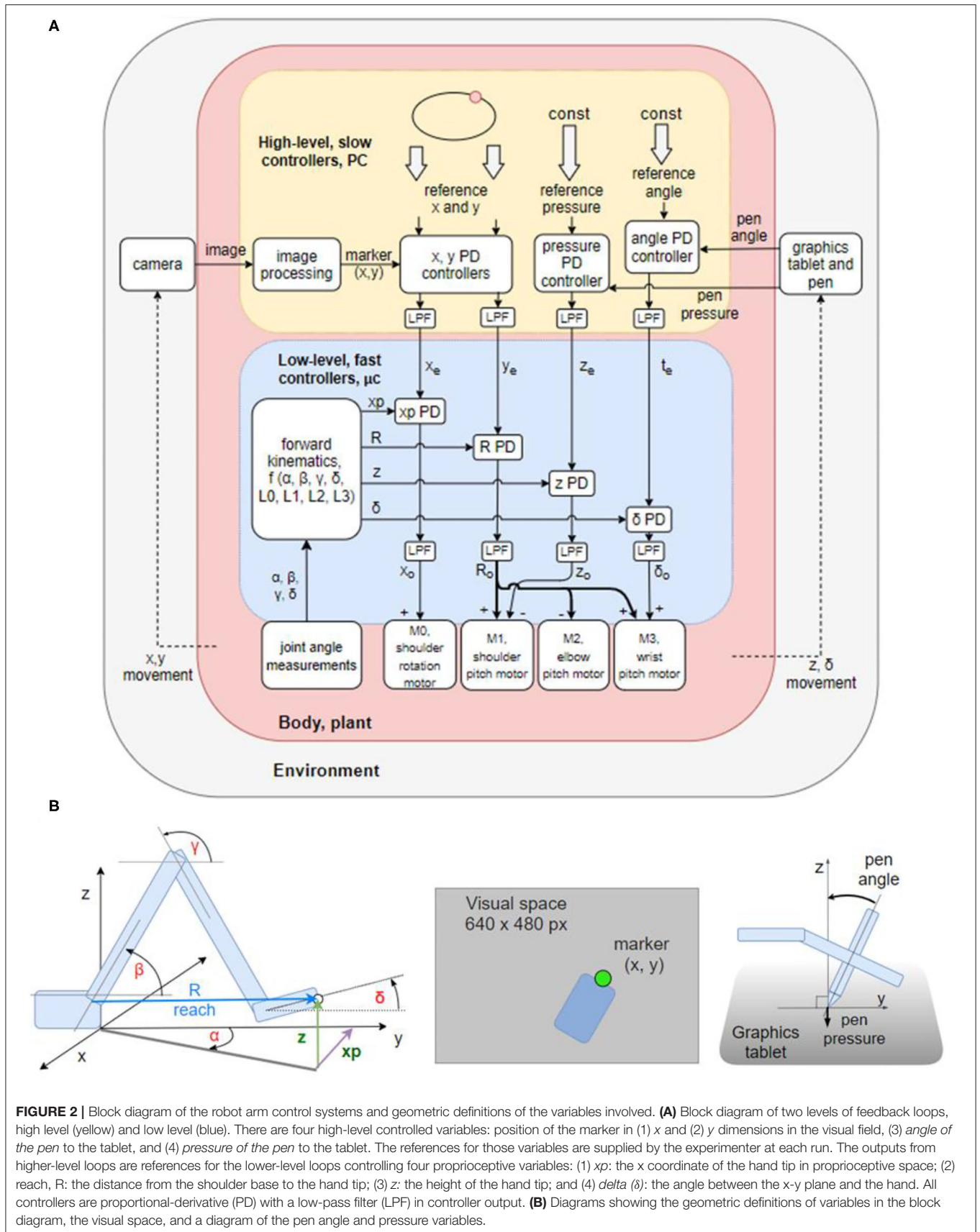
The control architecture of the present robot, as shown in the block diagram in **Figure 2A**, is not a verbatim copy of simulated systems. It is adapted to a 4 DOF arm using the guidelines and rules of thumb described in the literature. In addition to proprioceptive and visual variables, it also uses control of pressure, not found in the simulations. The architecture is composed of two levels with four control systems at each level. Starting from the bottom (body, plant), each degree of freedom of the robot contains a DC motor and a potentiometer for joint angle measurements. The voltages activating the motors are proportional to the net signal arriving from the outputs of lower-level controllers; i.e., *the same motors are used by multiple controllers, creating joint torque-coupling at the controller level.* The gears, potentiometers and housing of the motors come from RC servos, but the servo circuitry has been stripped away (see section Hardware: the Robot Arm for details).

The lower level of control (**Figure 2A**, blue box) is implemented on the microcontroller, it is faster (200 Hz cycle) and has a short signal transport delay (5–10 ms). It is controlling proprioceptive variables, with a relatively low gain. The proprioceptive variables $R, xp, z,$ and δ are constructed from low precision and noisy potentiometer readings and stored arm segment lengths. Their geometric meaning is shown in the **Figure 2B**. Note that joint angles are measured and used in forward kinematics equations, but they are not controlled.

The higher level (**Figure 2A**, yellow box) is implemented on the PC, it is slower (30 Hz cycle) and has a longer transport delay (180–190 ms). The variables controlled at the higher level are visual and tactile, constructed from camera images and graphics tablet and pen recordings. Their geometric meaning is shown on **Figure 2B**. The controllers at this level are slower, but have a higher gain than controllers on the lower level, i.e., they are more sensitive to errors. The errors x_e and y_e between the reference for the visual position (top level signal provided by the experimenter) and the sensed visual (x, y) position of robot arm sets the reference for the proprioceptively sensed endpoint positions xp and R. The tactile variable *pen pressure error* z_e sets the height (z) reference; and the *pen angle error* t_e sets the reference for the hand angle δ.

Proportional-Derivative Controller With a Low-Pass Filter

The basic unit of the control architecture (**Figure 2A**) is the proportional-derivative (PD) controller with a low-pass filtering element (LPF) on the controller output. All of the controllers of

FIGURE 2 | Block diagram of the robot arm control systems and geometric definitions of the variables involved. **(A)** Block diagram of two levels of feedback loops, high level (yellow) and low level (blue). There are four high-level controlled variables: position of the marker in (1) x and (2) y dimensions in the visual field, (3) *angle of the pen* to the tablet, and (4) *pressure of the pen* to the tablet. The references for those variables are supplied by the experimenter at each run. The outputs from higher-level loops are references for the lower-level loops controlling four proprioceptive variables: (1) xp: the x coordinate of the hand tip in proprioceptive space; (2) reach, R: the distance from the shoulder base to the hand tip; (3) z: the height of the hand tip; and (4) *delta (δ)*: the angle between the x-y plane and the hand. All controllers are proportional-derivative (PD) with a low-pass filter (LPF) in controller output. **(B)** Diagrams showing the geometric definitions of variables in the block diagram, the visual space, and a diagram of the pen angle and pressure variables.

the hierarchy can be described by the following equations in the time domain:

$$o = K_p \left(i_{ref} - i \right) - K_v \frac{di}{dt} \qquad (1)$$

$$\frac{dy}{dt} = \frac{o - y}{tc} \qquad (2)$$

where (1) is the PD controller equation with controlled variable i, reference input i_{ref}, output o, the derivative of the controlled variable di/dt, proportional gain Kp, and derivative gain Kv. Equation (2) models the low-pass filter with input o and output y, and tc is the open-loop time constant of the low pass filter.

The purpose of the low pass filter is to modify the bandwidth of the system. In the presence of delays and noise, high-frequency disturbances might cause positive feedback and instability. The low-pass filter attenuates the amplitudes of high-frequency components or completely removes them from the controller output, preserving stability of the loop. At the same time, the proportional and derivative gain can be kept high to maintain error sensitivity and good performance at low frequencies.

Construction of Controlled Variables

Robot arms are usually built with a servo motor in each degree of freedom. Consequently, at the lowest level, controlled variables are joint angles, its derivatives—joint velocities and accelerations—or joint torque. Building the robot arm from scratch enabled the experimentation with alternative control architectures that create coupling between multiple joint motors and skip joint angle control, which is arguably more biologically realistic. Instead of joint angle control, we constructed the controlled variables at the first level using standard forward kinematics equations, stored arm segment lengths and joint angle measurements obtained from potentiometers placed at each joint. Due to imprecise measurements, noise, gear backlash, and flex in the plastics of the arm segments, the gain of control systems on the first level was relatively low.

There are four controlled variables at the first level: (1) xp, the x coordinate of the hand tip in proprioceptive space; (2) R or reach, the distance from the base to the hand tip; (3) z, the height of the hand tip or z coordinate in proprioceptive space; and (4) δ (delta), the angle of the hand with respect to the x-y plane. The position in x dimension of the hand tip in kinesthetic space (xp) is controlled by activating the shoulder rotation motor M0. This configuration limits the work area of the arm to <180° in the upper half-plane (y>0). Reach R is controlled by simultaneously activating shoulder (M1), elbow (M2), and wrist (M3) motors, with the elbow being activated in the opposite angular direction from the other two. Height z of the hand tip is controlled by moving the shoulder motor M1. Angle delta between the hand and the x-y plane is controlled by moving the wrist motor M3. All the variables are controlled simultaneously. For instance, if correcting the height variable creates an error in δ angle, it will be treated as a disturbance to the δ angle control system and will be corrected simultaneously to the height error, creating a coupling between control systems. Joint angles are not calculated before starting the movement as in traditional inverse kinematics. Joint

motors move until all the errors are reduced. These calculations are performed on the microcontroller.

At the second level, there are also four controlled variables: (1) x and (2) y positions of the marker in visual space, (3) pen pressure and (4) pen angle (**Figure 2B**). These control systems are implemented on the PC, in a script written in Python. The image processing algorithm uses the OpenCV library to find the location of a green marker placed on the tip of the hand of the robot (**Figure 2B**). The image from the camera is first converted from BGR color space to HSV; then an inRange filter is applied to extract the green-colored areas. The filtered image is eroded and dilated to remove noise. Finally, the location of the marker is taken to be the center of the largest contour found on the image. The location of the marker is reported in pixels. Each variable is sampled or calculated at ∼30 Hz, determined by the sampling rate of the camera. Signal transport delays in visual loops are ∼180–190 ms. Pen pressure and pen angle are read-out from the Wacom tablet using the PyQt5 tabletEvent api, with pressure being measured in percentages and angle in degrees.

Tuning the Control Systems

We first tuned the lower-level, proprioceptive loops. The tuning procedure started with setting proportional and derivative gains to zero, and the time-constant of the low pass filter to a low value. Next, we increased the proportional gain until oscillations appeared after a step reference, and then we increased the derivative gain until the oscillations would stop. If the precision were not high enough, then we would increase the time constant of the LPF and retune the proportional and derivative gains to a higher value, trading bandwidth for precision and stability. The time constant tc of the low-pass filters was 80 ms for all the controllers at the lower level. For tuning the higher levels, we applied step references and aimed for a critically damped response using the same trial and error procedure as described. The PD controllers at the higher level are identical to lower-level controllers expressed in Equations (1) and (2). They differ only in parameters. Loop gains and open-loop time constants are much larger in higher-level loops in order to achieve stability and precision in conditions of large loop delays and noise from the marker location finding algorithm.

Data Analysis

Robot hand trajectories were extracted from the camera-recorded positions and estimated hand marker locations. We did not use the position of the pen on the tablet since the position of the hand-tip marker was not identical to the position of the pen. The experimental signals were smoothed with low-pass second-order Butterworth filter, with the cutoff frequency specified for each analysis, in order to tame the relatively high levels of noise, aiming for the preservation of position and velocity profiles, and taking into account the speed of arm movement. Trajectories of the computational model were not smoothed.

RESULTS

Beyond computer simulations and blackboard mathematics, we studied the robot arm as an "embodied control architecture"

in the real world to see how it can deal with tasks commonly performed by humans and other primates, while adaptively managing noise, delays, non-linearities, unpredictable disturbances, and perturbations.
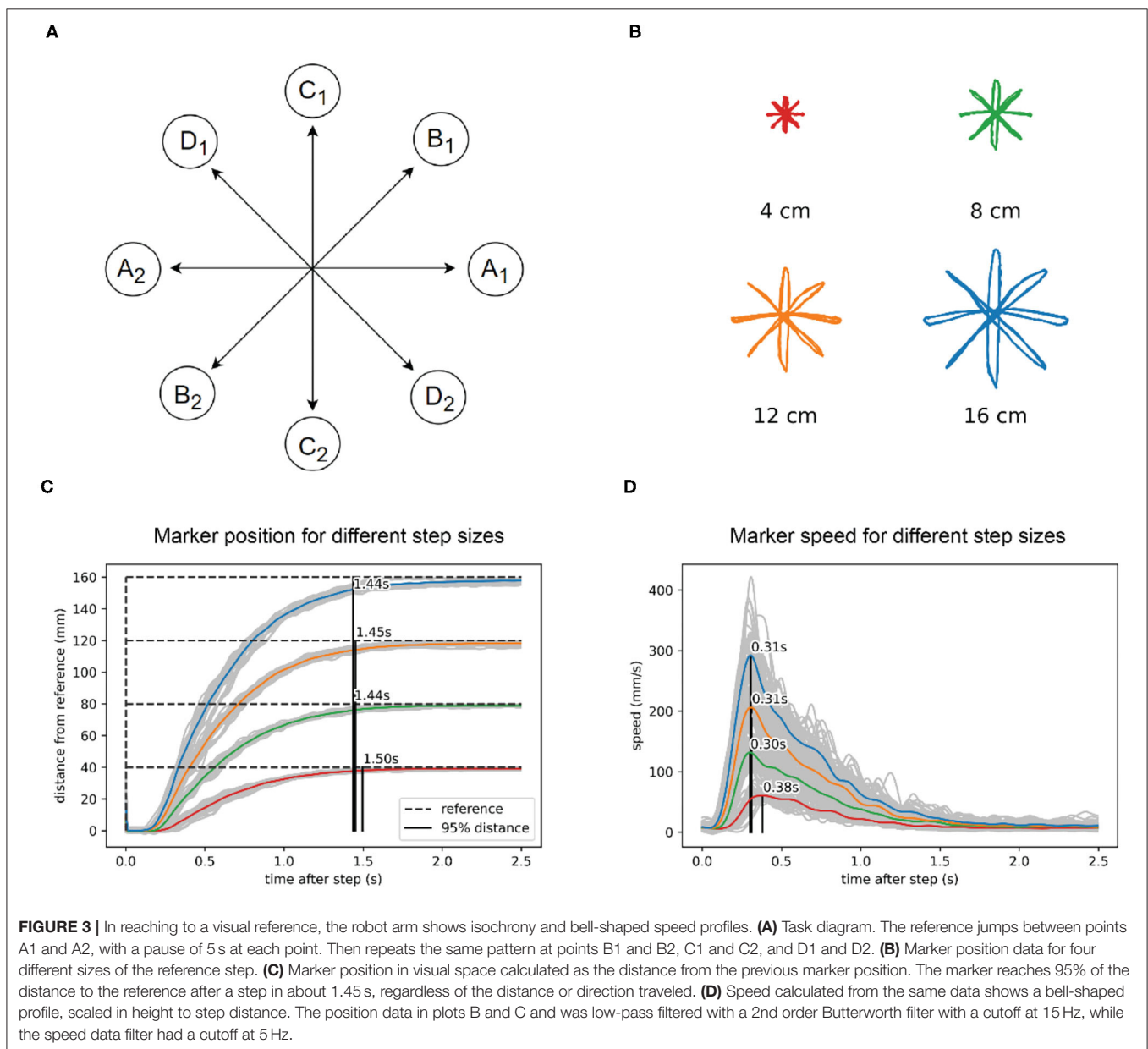
Having built the robot arm hardware from scratch and having implemented the hierarchical control algorithms as described above, our main goal was 2-fold: first, to examine the behavior of the system in its own right, and second, to compare the behavioral features of the robot arm to known properties and invariances of human arm movement.

Task I: Straight Movements in a Reaching Task

The first test is a reaching paradigm similar to the center out reaching task (**Figure 3A**) often used in primate and human

movement research (e.g., Cisek and Kalaska, 2002; Inoue et al., 2018). We applied the step reference signal simultaneously to x and y visual tracking loops. There was no central stopping point: for one size of the task, there were 10 movements in each direction, done in sequence, for a total of 40 movements, with a 5-s pause on the endpoints. The task was repeated 4 times with different lengths of movement at 4, 8, 12, and 16 cm (**Figure 3B**). We did not randomize the movement directions, since the robot did not have any learning capabilities that might have influenced the reaction time or movement trajectories.

We found that the robot performs straight movements across different lengths and different directions in approximately the same time: 1.45 s (**Figure 3C**). For the shortest movements (4 cm) there was a deviation of 50 ms from the average duration of longer length movements. The speed profile (**Figure 3D**) was

FIGURE 3 | In reaching to a visual reference, the robot arm shows isochrony and bell-shaped speed profiles. **(A)** Task diagram. The reference jumps between points A1 and A2, with a pause of 5 s at each point. Then repeats the same pattern at points B1 and B2, C1 and C2, and D1 and D2. **(B)** Marker position data for four different sizes of the reference step. **(C)** Marker position in visual space calculated as the distance from the previous marker position. The marker reaches 95% of the distance to the reference after a step in about 1.45 s, regardless of the distance or direction traveled. **(D)** Speed calculated from the same data shows a bell-shaped profile, scaled in height to step distance. The position data in plots B and C and was low-pass filtered with a 2nd order Butterworth filter with a cutoff at 15 Hz, while the speed data filter had a cutoff at 5 Hz.

roughly bell-shaped with a shorter rise and longer fall segment and scaled with the length of movement. In all movements, the maximum speed was achieved at approximately the same time after the reference step (peak at 0.33 s, **Figure 3D**), except in the shortest movement of 4 cm, where the peak of maximum speed was 70 ms later than the average of the other movements. Robot position data were low-pass filtered using a second-order Butterworth filter with a cutoff at 15 Hz, while the speed data filter had a cutoff at 5 Hz.

Task II: Curved Movements in a Tracking Task

Producing elliptic traces or drawings in humans in a fast and fluid manner results in a speed-curvature relationship known as the 2/3 speed-curvature power law (or 1/3 power law, depending on the variables used), first described by Lacquaniti et al. (1983). In the second task, we tested the production of curved movements. We used a continuously moving reference point, and we report on the situation where the reference moved at a constant speed along an elliptic path.

An elliptic trajectory with a constant tangential speed is a non-power law trajectory ($\beta \approx 0$, $r^2 \approx 0$); the x and y components are not pure sinusoids, but also contain higher frequency components. In the low frequencies, the speed of the robot was close to the reference speed, but at higher frequencies, the speed of the robot was not constant and had a sinusoidal profile and a lower average than the reference speed. At the highest frequency of input ($f = 0.826$ Hz, speed ≈ 406 mm/s), the output trajectory followed a speed-curvature power law with an exponent of $\beta \approx -1/3$ and $r^2 = 0.78$ (**Figure 4C**).

This seems to support the hypothesis that the power law is a consequence of the physical limitations of the human arm, and not a planned invariance. However, the size of the drawn shape was smaller than the reference shape because both x and y components of the reference are attenuated in the output. Additionally, the position data was smoothed with a low-pass filter with a cutoff at 3 Hz and differentiated, which increased the coefficient of determination (r^2) of the power law. To further probe the question of the origin of reaching and tracking invariances, and to minimize the effect of noise, we created and fitted a model of visual loop behavior described in the next section.

Modeling and Characterization of the *in Machina* System

The system controlling the visual y variable (**Figure 5A**) is non-linear because increasing reach affects the y position differently depending on the angle of rotation of the base (α). The three motors involved in changing reach (**Figure 5A**) are different in power and mechanical linkage; they have a different effect on changes in reach depending on the position they are in. Despite non-linear elements, the behavior of high-level visual loops can be described fairly well by a set of linear second-order equations, modeled in the block diagram (**Figure 5B**), the free-body diagram (**Figure 5C**), and a system of equations:

$$e_t = y^{ref}_{t-d} - y_{t-d} \tag{3}$$

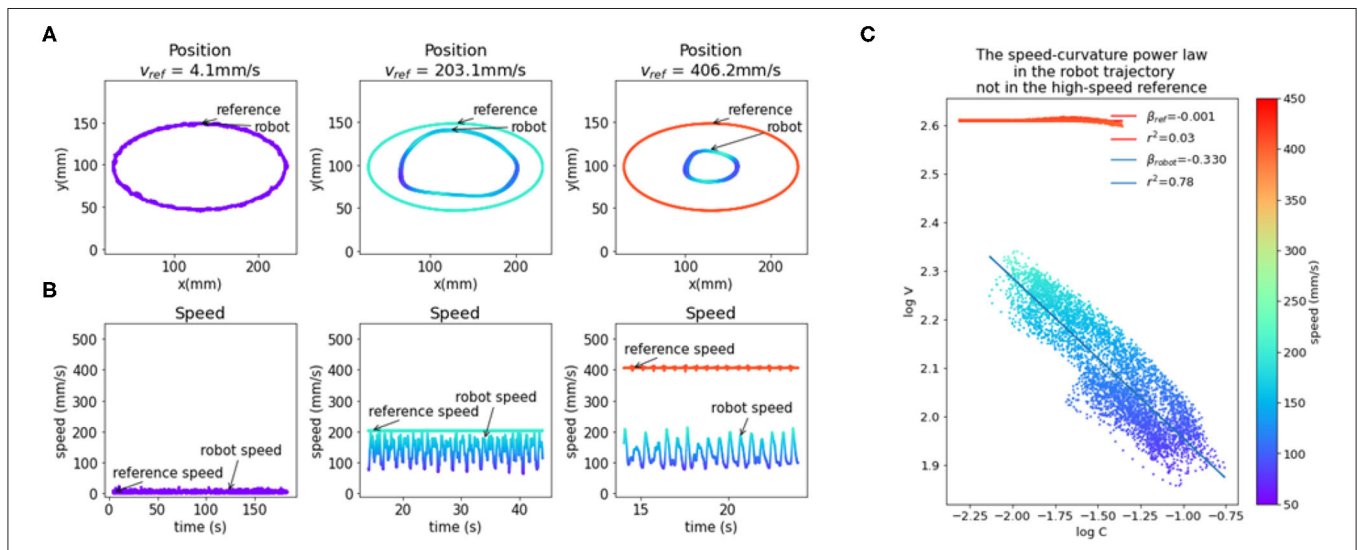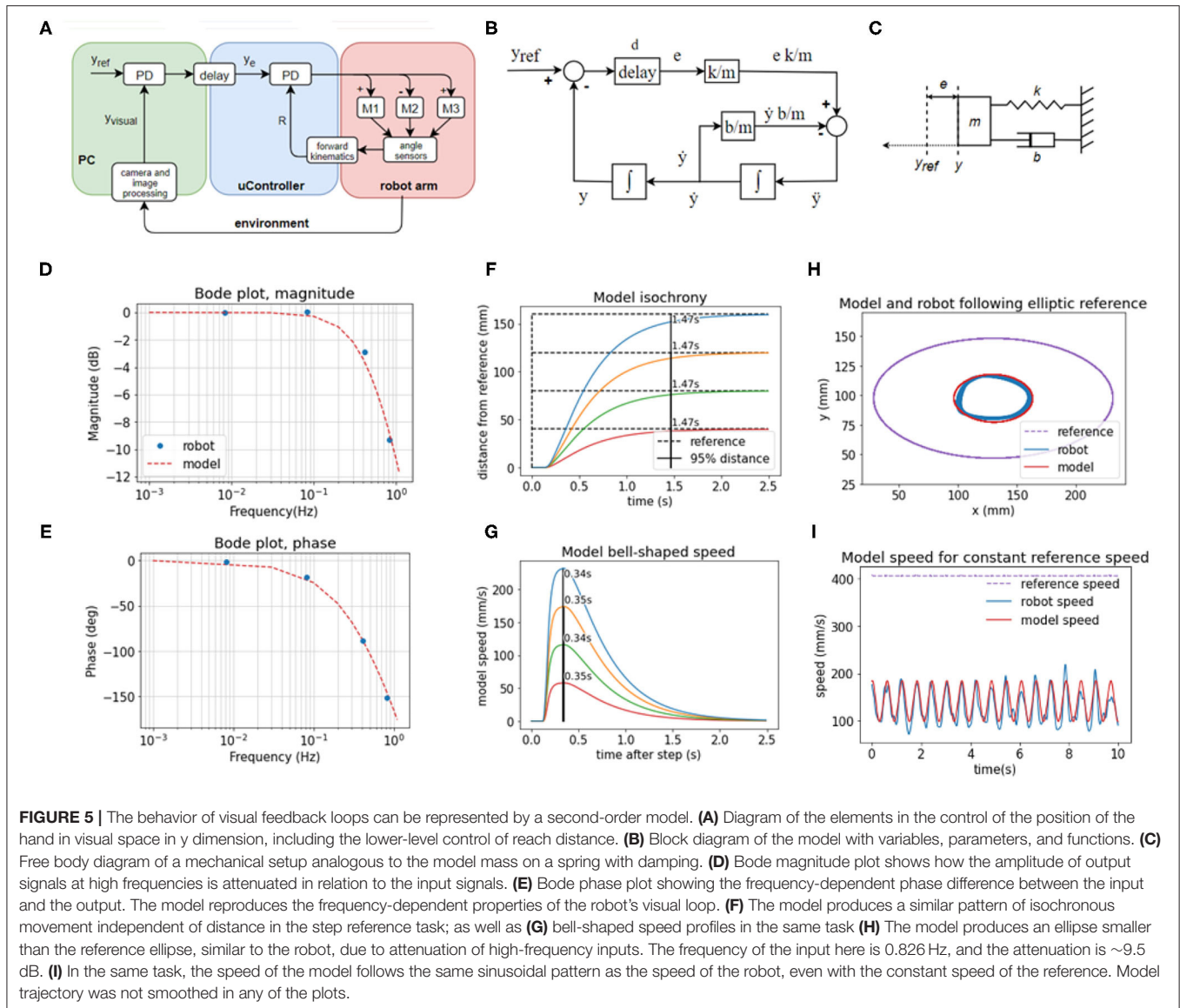$$\ddot{y}_t = \frac{k}{m}e_t - \frac{b}{m}\dot{y}_t \tag{4}$$

FIGURE 4 | The emergence of the speed-curvature power law at high speed. **(A)** Reference and robot positions. When the reference speed is low, the position of the robot hand tip marker in visual space matches the speed of the reference. At higher speeds, there is a magnitude attenuation—the ellipse shape is smaller. **(B)** Plot of reference and robot speeds. At low speed, the robot speed matches the reference. When the reference speed is high, the speed of the robot is lower and oscillates, even though the input speed is constant. **(C)** Oscillations in robot speed for the highest speed input are regular and correlated with local curvature of movement, following the speed-curvature power law with the exponent of $\beta \approx -1/3$ and $r^2 = 0.78$. The reference speed is constant (400 mm/s), and there is no power law ($\beta \approx 0$, $r^2 \approx 0$). In all the plots colors indicate speed, as shown in the color bar on the right. Position data were smoothed with a 2nd order Butterworth filter with a 3 Hz cutoff and differentiated to generate speed data.

FIGURE 5 | The behavior of visual feedback loops can be represented by a second-order model. (A) Diagram of the elements in the control of the position of the hand in visual space in y dimension, including the lower-level control of reach distance. (B) Block diagram of the model with variables, parameters, and functions. (C) Free body diagram of a mechanical setup analogous to the model mass on a spring with damping. (D) Bode magnitude plot shows how the amplitude of output signals at high frequencies is attenuated in relation to the input signals. (E) Bode phase plot showing the frequency-dependent phase difference between the input and the output. The model reproduces the frequency-dependent properties of the robot's visual loop. (F) The model produces a similar pattern of isochronous movement independent of distance in the step reference task; as well as (G) bell-shaped speed profiles in the same task (H) The model produces an ellipse smaller than the reference ellipse, similar to the robot, due to attenuation of high-frequency inputs. The frequency of the input here is 0.826 Hz, and the attenuation is ~9.5 dB. (I) In the same task, the speed of the model follows the same sinusoidal pattern as the speed of the robot, even with the constant speed of the reference. Model trajectory was not smoothed in any of the plots.

where t is time, e is position error, y_{ref} the position reference, y position, d is loop transport delay. The values of the coefficients used are k/m = 40, b/m = 27.5, and $d = 0.185$ s, which were found by fitting the behavior of the model to the behavior of the robot in the step reference task with a 12 cm step distance. The best fit values indicate a slightly overdamped second-order system. We modeled the x and y control loops as independent systems with equal parameters.

The model in Equations (3) and (4) is analogous to a mass-spring-damper system (**Figure 5C**) with a movable equilibrium position and a pure delay element. The approach trajectory of the marker on the hand of the robot to the visual reference in the step-reference task is similar or analogous to the approach trajectory of an object of mass m on a spring with stiffness k and damping b toward its equilibrium position y_{ref}, where the displacement of the equilibrium position happens after a delay

of d seconds. The spring constant is an analog of the visual gain or sensitivity to error; the damping coefficient b is an analog of the combined effect of visual velocity gain (damping term in the visual PD controller), gains at the proprioceptive level, and friction between the pen and the tablet; the mass in the mass-spring-damper system is an analog of the combined contribution of the mass of the robot arm, time constants in the visual loops and inertia of electromotors; and the delay is the total duration of the travel of the signal around the visual loop, combining camera latency, frame rate, and transmission delays of the serial protocol between the PC and the microcontroller.

We examined the behavior of the robot in response to sinusoid inputs across a range of frequencies (0.008, 0.083, 0.413, 0.826 Hz). We applied input as the visual reference signals and measured the output as the position of the marker in the visual field over time. We calculated the relation of output

amplitude A_{out} to input amplitude as $20 \log_{10} (A_{out}/A_{in})$ for the Bode magnitude plot, and we calculated the phase difference between the input and output sinusoids for the phase plot. We then interpolated the plot using the second-order mass-spring-damper model (**Figures 5D,E**, model interpolation in red, experimental values in blue).

Looking at the Bode plot (**Figures 5D,E**), we can see that the system is stable for all input frequencies. At the gain crossover frequency of ~0.1 Hz, the system has a large phase margin of about 160°, and at the phase crossover frequency at ~1 Hz the magnitude attenuation is 11 dB, which satisfies the stability criterion of having both the phase and gain margins positive. The bandwidth is limited by the large transport and processing delays in the visual loops, approximated at 185 ms. The delays cause a phase shift that takes an increasingly longer part of the sine period with the increase in frequency and are compensated by low pass filtering the controller output (see Equations 1 and 2 in Methods section).

We repeated the step-reference task with the model (**Figures 5F,G**). The duration of movement is isochronous across different distances: it takes the same amount of time, 1.47 s, to cross 95% of the distance to the reference. The speeds are bell-shaped and scaled with distance, but they all reach a peak after 0.37 s, replicating very nearly the behavior of the robot arm.

Finally, in the ellipse tracking task (**Figures 5H,I**) at the highest frequency of 0.826 Hz of input, the model replicated the size of the robot trajectory, and also the properties of the speed profile. Even when the reference speed was constant, at this frequency, the speed profile of the model was sinusoidal. Model trajectory followed a speed-curvature power law with the exponent $\beta = -0.40$, $r^2 = 0.98$.

Model trajectories in panels (**Figures 5D–I**) were produced by a model with the same, constant coefficients, simulated with a time step of 5 ms and Euler integration. Model trajectories were not low-pass filtered.

Task III: Robustness to Blocking the Robot Arm's Wrist Joint

The hallmark of biological motor control is robustness to perturbations. In further testing of the robot arm, we applied different perturbations to the controlled variables, keeping them constant for the duration of the task and not changing any of the parameters in the controllers or other parts of the software of the robot. We blocked the wrist, tilted the writing tablet, added a tool that extended the arm, and rotated the visual field.

In the first trial, we blocked the wrist joint and compared the performance of the robot in visual tracking tasks to the performance in normal operation where the wrist was moving freely. Without any changes to the code or parameters of the control systems, the robot arm performed the tasks even with the wrist blocked. In normal operation, the variable *reach* (**Figures 6A,B**) is affected by three motors—in the shoulder (M1), elbow (M2), and wrist (M3). The time-plot of the variables in normal operation in the step reference task (**Figure 6D**) shows joint angles that illustrate how all three motors contribute to the movement. When the wrist is blocked, reach is maintained at

the same desired value as in the normal situation. However, here reach is not affected by three motors, but only by two: the elbow and shoulder motor automatically pick up or compensate for the work normally done by the wrist motor because their activation is proportional to the reach error.

The block diagram shows the flow of information in the reach synergy (**Figure 6B**). We can describe the system using the terminology of Latash (2008) or Latash et al. (2007): the performance variable is *reach*, and it is maintained at its reference level y_e by varying elemental variables—activations o1, o2 and o3, of motors M1, M2, and M3, respectively. The activations are calculated by weighting the output R_o of the controller, and summing the signals with outputs from other systems, here control of height z and wrist angle δ. These sums (o1, o2, and o3) are used as activations of motors, as pulse-width-modulated signals from the microcontroller to the motor driver chip.

Task IV: Further Perturbations: Tilting the Tablet, Using Tools, Rotating Point-of-View

We applied static disturbances or perturbations to controlled variables either directly or indirectly to examine the adaptiveness and robustness of the robot control architecture. In the normal condition, without additional perturbations (**Figure 7A**), the writing tablet is horizontal, the wrist is mobile, the marker is on the tip of the hand of the robot, and the visual coordinate system is roughly aligned with the proprioceptive coordinate system along the × and y axes. The angle of the pen to the tablet is sensed and maintained at 0 degrees (pen is perpendicular to the tablet, while the hand is parallel to the tablet) by moving the wrist joint. The pressure of the pen to the tablet is sensed and maintained at or near 50%.

In the tilted tablet condition (**Figure 7B**), the end of the tablet distal to the robot was lifted to make the tablet close an angle of 30° to the surface. This perturbation challenged the pen pressure and pen angle control systems because the pressure control system needs to continuously modify the height of the hand in order to keep the pressure at 50% and still move toward the reference in the visual space. The pen angle control system needs to modify the wrist angle so that the pen is always orthogonal to the tablet surface. The plots of pen angle and pressure show that those variables were maintained near their reference values despite the perturbation, with somewhat more error than in the normal condition.

In the tool use task (**Figure 7C**), we added a 12 cm long plastic piece to the tip of the arm and moved the marker forward to the end of the plastic piece, creating a situation resembling tool use, as now the tip of the "tool" was tracking the reference. We moved the camera about 12 cm forward to keep the workspace in the visual field. The robot performed the task without learning or reprogramming visual transformations. In the next task, we rotated the visual field by 30° (**Figure 7D**) by rotating the camera and keeping the robot in place. This amount of rotation is near the limits of performance—the robot performed the task with higher amounts of error, visible on the patterns (red) on the plots.

The perturbations summary plot shows the average absolute error as the average distance of the hand marker from the
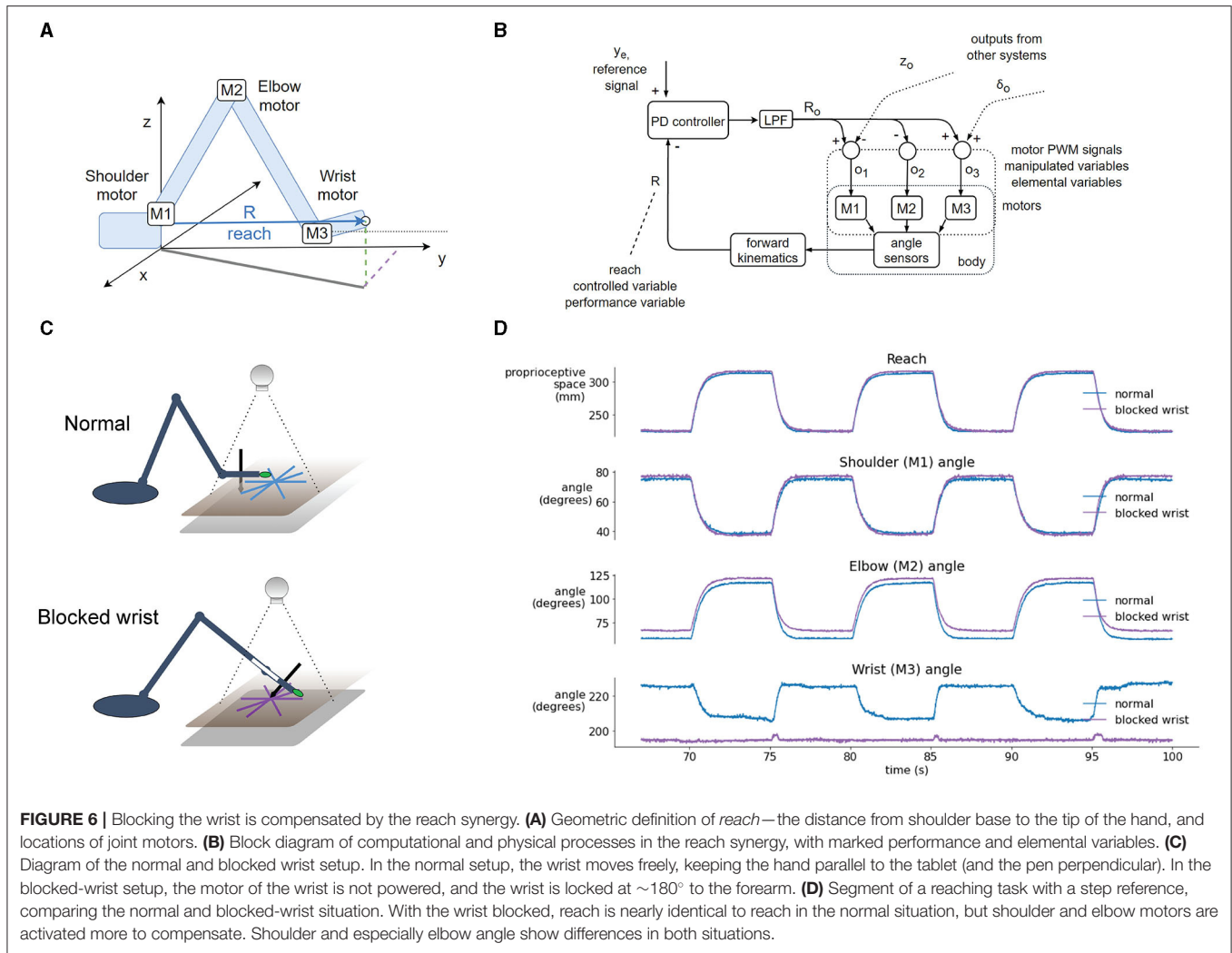
FIGURE 6 | Blocking the wrist is compensated by the reach synergy. **(A)** Geometric definition of *reach*—the distance from shoulder base to the tip of the hand, and locations of joint motors. **(B)** Block diagram of computational and physical processes in the reach synergy, with marked performance and elemental variables. **(C)** Diagram of the normal and blocked wrist setup. In the normal setup, the wrist moves freely, keeping the hand parallel to the tablet (and the pen perpendicular). In the blocked-wrist setup, the motor of the wrist is not powered, and the wrist is locked at ∼180° to the forearm. **(D)** Segment of a reaching task with a step reference, comparing the normal and blocked-wrist situation. With the wrist blocked, reach is nearly identical to reach in the normal situation, but shoulder and elbow motors are activated more to compensate. Shoulder and especially elbow angle show differences in both situations.

reference in all perturbation tests across different tasks. The error grows with distance or the size of the step in the step-reference task. In pseudorandom tracking, the error grows with "difficulty," where more difficult tasks have a higher magnitude of high-frequency signals. In the ellipse tracking task, the error grows with frequency or with the speed of the reference. Thus, our robot arm is highly robust to external perturbations, akin to human movement.

DISCUSSION

We have demonstrated how and explained why a custom-made robot arm (**Figure 1**) with a hierarchical control architecture (**Figure 2**) based on simulations by Powers (1999, 2008) displays basic features characteristic of biological movement. Being robust to noise and delays, the robot's behavior complies with isochrony and displays bell-shaped velocity profiles in a reaching task (**Figure 3**). In tracking a moving target at high speeds, the robot complies with the so-called speed-curvature power law of human movement (**Figure 4**).

We must acknowledge that the proposed control architecture does not fully explain the production of reaching and tracking trajectories since (i) the reaching trajectories of the robot are isochronous for different reach distances, while humans may change the reaching duration according to speed or accuracy demands of the task (Fitts, 1954); and (ii) the tracking of elliptic reference trajectories does not reproduce the geometrical trace or exact speed profile of the reference, with the amplitude of movement falling with frequency, and the shape reducing in size.

However, it is clear that the hierarchical organization of the control systems afford a lot of flexibility to the robot arm even without learning algorithms or online optimization. Moreover, we have demonstrated adaptive behavior to structural perturbations such as blocking the robot's wrist (**Figure 6**) and to environmental configuration disturbances such as tilting the writing tablet, extending the hand with a tool, or rotating the visual field (**Figure 7**). The main reason for such a robust behavioral emergence seems to be the choice of the controlled variables and the hierarchical arrangement of the control systems. We further discuss these findings below.

FIGURE 7 | Diagrams of perturbation conditions and a plot of tracking performance in different tasks. **(A)** the normal condition diagram and plots of pen angle and pressure from a pseudorandom tracking task, reference in red and measured variable in blue. **(B)** The tablet is tilted 30°, distal part is lifted, plots show the pen angle, and pressure in a pseudorandom tracking task, reference in red and measured variable in blue. **(C)** Diagram of the tool use task, a 12 cm plastic piece is attached to the hand of the robot, and the maker placed on the end. **(D)** Diagram of the camera rotation task; the camera is rotated by 30° when compared to the "normal" conditions. **(E)** Diagram of the blocked wrist. **(F)** Average distance (absolute error) between the marker and the reference in the visual space, across different conditions and tracking tasks. Bellow the plot are robot trajectories for each perturbation condition and task.

Frequency Response and Stability Despite Noise and Delays

The frequency response of the robot's visually guided behavior (**Figure 3**) shows how the arm behaves in response to input signals of different frequencies, where the "input" is the reference or setpoint input to the hand tip position in the visual space, and the "response" is the hand tip position in visual space. We can see that the system is stable for inputs of any frequency because

both the phase and gain margins are positive. The system has a slightly overdamped second-order response (simplified model of the robot arm discussed later in more detail), and it acts as a low pass filter with the gain crossover frequency of ∼0.08–0.1 Hz. This means that magnitudes of all input signals above this frequency will be attenuated.

Transport delays in the human visuo-manual loops in tracking pseudorandom targets are approximated at 100–150 ms (Viviani et al., 1987; Parker et al., 2017). Visual loops of the present robot also contain large signal transport delays, 180–200 ms, that come from camera latency and refresh rate, as well as serial protocol communication delays between the microcontroller and the PC. Delays in feedback loops cause a frequency-dependent phase shift so that at high frequencies of input, the phase shift might cause the actuator action to add to the error, creating positive feedback as opposed to negative feedback where the error is reduced, which may cause instability and oscillation. Long transport delays are often cited as the main reason for relatively complex delay-compensation schemes such as forward models (e.g., Miall et al., 1993; Kawato, 1999; Desmurget and Grafton, 2000). A forward model, given the efference copy of the motor command, estimates the state of the arm at current time, instead of waiting for the delayed feedback signals. However, we show that an alternative, simpler scheme might work.

Stability can be achieved by reducing the bandwidth of the system—trading the bandwidth for stability—using low pass filter elements in the outputs of controllers (**Figure 2**). This maintains the visual loop gain high when the input frequency is low and reduces the effective gain for high-frequency inputs to avoid positive feedback (Powers, 2008). Additionally, the "real world" is highly unpredictable, and various disturbances acting on the arm would invalidate any prediction made by the forward model that takes only the motor command into account. The present architecture avoids the problem by always using feedback signals, affected by both motor action and environmental disturbances, as representations or parameters in calculation of actual state.

The robot arm is still capable of producing movements with high peak speed (**Figure 3**), while movements with both a short duration and a high peak speed might be produced if the movement was stopped by an obstacle. This might be a mechanism involved in fast, short movements of human hands in e.g., pressing piano keys.

Most of the effects of sensory noise in the high-gain visual level systems seem to be averaged over time by the low-pass filter and do not affect the movement, especially at low speed, and don't require additional compensation mechanisms. In the lower levels, sensory noise does not affect movement because the gains are low.

Isochrony, Bell-Shaped Velocity, and the Speed-Curvature Power Law

In humans, isochrony was found in drawing figures of different sizes (Viviani and McCollum, 1983; Viviani and Flash, 1995). It was also found in macaques in natural settings (Sartori et al., 2013), but not consistently in laboratory settings (Castiello and Dadda, 2019). In the Fitts tapping task, the time of movement is related to the so-called index of difficulty, and isochrony is

present not for all movements, but for tasks of the same index of difficulty (Guiard, 2009). The tapping task illustrates the speed-accuracy tradeoff: the faster we move, the less accurate our movements will be. Then, in order to preserve accuracy, presumably, when aiming for smaller targets, we slow down our movements.

We have found that the robot performs isochronously all movements in the step-reference task, regardless of travel distance or direction (**Figure 3**); for all movements in this task, the average reaching time was 1.47 s. In comparison, e.g., Fitts (1954) reports times from 0.180 to 0.580 s for reaching across comparable distances; depending also on the size of the target or the required accuracy. This illustrates that for a direct comparison with human movement, the robot arm would need to have (1) higher force-producing capabilities and (2) presumably a higher-level system that would be able to perceive target sizes and modify produced speeds accordingly. The arm might also be made closer in size to the human arm: with 12 cm forearm and 12 cm upper arm it is approximately a third of the size of the adult human arm.

In humans, during rapid straight-line hand movements, the speed profile is not constant. The movement starts slow, accelerates to a point, then decelerates to a stop, forming a bell shape over time or distance. Some researchers report symmetrical bell shapes where the peak speed is in the middle of the movement (Flash and Hogan, 1985), and some report less symmetrical profiles, namely short acceleration and longer deceleration phase in human (Soechting, 1984; Atkeson and Hollerbach, 1985) and in primate reaching (Inoue et al., 2018). In the step-reference task, we found that the speed profiles are bell-shaped, asymmetrical with a short acceleration phase and longer deceleration, and with the maximum speed scaled with the extent of the movement (**Figure 3**). The peak speeds for the robot arm were in the range from 50 mm/s for 4 cm distance to 300 mm/s for 16 cm distance; while in human reaching peak speeds are higher, e.g., Soechting (1984) reports speeds of 650–1,300 mm/s for reaching across 30 cm. Without the modifications to the robot arm system mentioned in the previous paragraph, it is not possible to make more direct comparisons.

We interpret this speed profile emerging as a consequence of programming visual level loops as proportional-derivative controllers. Since there was no trajectory planning or online optimization, our results show that it is possible to achieve such profiles with a simple control architecture, supported also by our mass-spring-damper simulation.

There were no accuracy requirements, but we found that increasing the frequency and speed in the ellipse tracking task decreased the accuracy, suggesting a speed-accuracy tradeoff in the movement of the robot. This tradeoff seems to be caused by several factors: (i) the low-pass filtering properties of the arm, resulting from its inertia, relatively low power of the actuators and also explicit low-pass filter elements, and (ii) the increased influence of lower-level non-linearities and control system interactions on the behavior of the robot because the errors on the lower levels were not corrected fast enough.

The speed-curvature power law with a two-thirds coefficient is observed in rapid elliptical movement in humans. This phenomenon can be roughly described as movement at lower

speed in areas of high curvature and relatively higher speed in areas of low curvature (Viviani and Terzuolo, 1982; Lacquaniti et al., 1983). The production of a power law trajectory is not obligatory in principle because the hand might take many of the infinite possible trajectories along the same path. However, the set of possible trajectories is limited by the physical properties of the hand and the environment. For instance, the hand will never move instantaneously from point A to point B, as there is a limit to the force produced by muscles.

Remarkably, when the robot was tracking a high-frequency elliptic reference, we found the speed-curvature power law in the measured movement, even when it was not present in the reference input. This result is consistent with the optimization of jerk in the movement planning phase (Viviani and Flash, 1995; Huh and Sejnowski, 2015). However, we seem to be getting an optimal jerk trajectory "for free," without an explicit optimization algorithm. This result shows that the speed-curvature power law can be achieved without explicitly optimizing jerk or smoothness either in the planning phase or online.

However, the robot did not accurately follow the path component of the reference. It seems that the emergence of the power law comes from the *failure* of the robot to accurately follow the high-frequency non-power law position reference, which is interesting. This can be viewed as a consequence of low-pass filtering the reference signal by the robot arm system. The position references in x and y dimensions have high-frequency components that get filtered out, leaving single-frequency sinusoid fundamentals, conforming to the power law. Similar results were obtained in simulations by Gribble and Ostry (1996) and Schaal and Sternad (2001) where low-pass filtering non-power-law input signals produced power law trajectories.

Related to this result, several studies with human participants have shown that it is difficult to accurately track targets that don't follow the speed-curvature power law (Viviani et al., 1987; Viviani, 1988; Viviani and Mounoud, 1990). However, the subjects did accurately follow the path component of the trajectory and the rhythm of the target. We further discuss possible mechanisms in the section "higher levels."

A Second Order Simple Model Accounts for the Robot's Behavioral Features

Human behavior in tracking pseudorandom targets can be accurately modeled by a first-order model with three constant parameters (see review in Parker et al., 2020). The step-response and frequency response in humans is also modeled by second-order models, bang-bang control, surge control, or the Crossover model (compared in Müller et al., 2017) with various tradeoffs in simplicity and accuracy of modeling.

Here, in turn, we modeled the behavior of the robot itself with a second-order system, a mass on a spring with damping, with three constant parameters (**Figure 5**) described by the system of Equations (3) and (4). Once the parameters were estimated, the model closely reproduced robot position and velocity in visual space in the frequency response task, in the step-reference task, and in tracking elliptic references. The model displayed isochrony and bell-shaped speed profiles in the step-reference

reaching task and the power law in the ellipse tracking task (**Figure 5**).

The fact that the model captures all these features of robot behavior with just three parameters is surprising given the multi-level control architecture, the non-linearities in the lower levels, and differences in motors in each joint. This finding points to an interesting property of hierarchical systems: higher-order loops may appear as linear systems regardless of non-linearities at lower levels. Higher levels provide reference signals to the lower-level systems, so the lower systems are part of the "plant" from the perspective of the higher systems. A certain range of variations and non-linearities in the plant will be hidden in the behavior of the high-level loop.

If the system for tracking a visual target appears to higher levels of the brain just like the robot arm visual control systems appear to us as experimenters, movement control might be relatively simple for the higher brain structures. As postulated by Viviani and Mounoud (1990), all voluntary movement might be a special case of pursuit tracking, where the only difference from conventional tracking is that the target is internal. The hypothetical higher-level system would only need to specify the virtual target, which is identical to the hand position reference in the robot arm.

Higher Levels of Control

In optimal feedback control theory, as well as in industrial robotics control, the solution to the problem of producing a trajectory might involve forward or inverse models, online optimization with a changing horizon, and similar schemes. Those methods are very powerful, especially when coupled with modern computers, precise actuators, and relatively noise-free environments. However, from an academic perspective, they are criticized for not being empirically refutable or biologically plausible (Powers, 2008; Scott, 2012; Feldman, 2015). In the framework of hierarchical perceptual control (Powers, 1973), higher levels should be controlling variables more abstract than lower levels, and also work more slowly having a larger time constant and longer transport delays.

One hypothesis arising from the analysis of straight and curved movements of the robot arm is that the present architecture is missing higher levels of control. In the present architecture, the visual position references are set by the experimenter. A hypothetical higher level would be taking the role of the experimenter and would attempt to control or maintain a high-level variable at a desired value by using the position reference as the manipulated variable, much like the current visual position control loops manipulate references to proprioceptive variables to bring visual position to the desired value. The visual position reference would be equal to the produced visual position in the steady state, but not necessarily during transients.

Harris and Wolpert (1998) show that minimizing the variance of the hand trajectory from the desired path over a set of movements (accuracy) for a given speed, or minimizing the speed for a given desired accuracy, reproduced the bell-shaped speed profiles and the two-thirds power law. Their result was achieved in the framework of planning an optimal trajectory and

executing it in an open-loop manner; however, we propose that their cost functions for speed and accuracy might be treated as explicit higher-order controlled variables that set visual position references. This would amount to independent control of path accuracy and average speed.

Alternatively, elliptic or straight-line movement can be produced rhythmically. There is some evidence that control of movement amplitude and frequency are developed independently. For instance, 5-year-old children occasionally produce sinusoidal movements that match the amplitude but not the frequency of the target, while other children match the frequency but not the amplitude (Mounoud et al., 1985). This might suggest a closed-loop pattern generator, similar to a phase-locked loop, that produces a patterned reference for the position control system (see Matic and Gomez-Marin, 2019). More research is needed to further elucidate these questions.

Controlled Variables in a Hierarchy Explain Adaptivity to Perturbations

The ability of humans and animals to achieve the same task outcome using different motor means has been termed as the phenomenon of *motor equivalence*. The problem it poses to motor control theories is the apparent rapid selection of correct means from the space of all possible means. While motor synergies and hierarchical control are proposed as the solution for the problem (Bernstein, 1967), the concept of synergy in motor control literature is defined in many different ways (see review by Bruton and O'Dwyer, 2018). Our reach control system fits the definition of Latash et al. (2007), and we termed it the *reach synergy*.

To probe the system, we blocked the wrist of the robot and put it through the same battery of tests as in the normal condition: the step reference reaching task and tracking pseudorandom and elliptic targets. Without any reprogramming or autonomous learning algorithms, the robot still performed the task with similar performance to the normal condition (**Figure 6**), with the exception of pen angle not being controlled (as the pen was fixed perpendicularly to the hand, and the wrist was blocked). Wrist blocking was also modeled in the DIRECT model (Bullock et al., 1993) where they argued that fast adaptations to losing a degree of freedom probably exclude complex planning as a relevant mechanism as it would take too much time, and the same effect can be achieved by simpler schemes. Our result seems to be consistent with the *minimum intervention principle* (Todorov and Jordan, 2002b) where the task-level variable of reaching toward a goal is maintained, and the variability caused by blocking the wrist is taken up by task-irrelevant variables of elbow and shoulder joint angles. The minimum intervention principle may emerge from an online movement optimization algorithm; however, here we achieve the same result without optimization, by having a flexible control hierarchy.

Maintaining pen pressure and angle are indeed important skills in handwriting. Measures of quality of control have been linked, for instance, to dysgraphia as a diagnostic criterion (Mekyska et al., 2017). Modeling contact forces in model-based control and optimal feedback control is still an open problem. Control systems for pen pressure and pen angle were implemented for this robot as slow, but precise systems in the higher level. The precision afforded by the pressure control was compensating the lack of precision in the lower-level proprioceptive systems. We tested pressure and angle control systems by tilting the graphics tablet by 30° and keeping the reference for the pen angle toward the tablet at 0° and the reference for pressure at 50%. Next, the robot performed the battery of tracking tests (**Figure 7**), and we found the performance close to the normal condition. The robot automatically adjusted the height of the tip of the hand and the angle of the wrist in order to maintain the angle and pressure references. In this case, we can see that precise pressure control is crucial for maintaining the pen on the tablet, similarly to human handwriting.

The visual coordinate system was "retina-based" in the sense that the two-dimensional visual field recorded by the camera was the working space of the robot. It was somewhat primitive, as it could only find the location of a green marker in two dimensions. However, it was robust to perturbations. The robot arm and the camera, and their respective proprioceptive and visual coordinate systems were only roughly aligned to begin with, and this was sufficient for normal operation. In the first test, we extended the arm with a 12 cm long piece of plastic and put a marker on the tip of the plastic instead of the tip of the hand, simulating writing with a longer pen or reaching with a stick. The visual system had no information about the size of the stick, or for that matter, the size of the robot arm or the configuration of its angles, but only the location of the marker. This was sufficient to enable the robot to track the reference with this "tool."

In the second test, we rotated the camera by 30°. This made the relationship between visual location proprioceptive location variables more non-linear than in normal operation. The performance in this test was somewhat worse than in normal condition (**Figure 7**), but the tasks were still successfully performed: in the step-reference condition, the hand tip reached the reference position and settled at that position, and in target tracking, the hand tip followed the reference signal in a similar manner to the normal situation.

These perturbations to the visual system do not greatly affect performance because there are no explicit coordinate transformations between the visual and kinesthetic loops. All transformations are implicit: the higher levels tell the lower levels to "move until the higher-level reference state is achieved." Moving in approximately the right direction seemed to be enough. As discussed, most non-linearities in the lower levels were absorbed by the high-gain higher-level control systems, at least in the low frequency, low-speed movement.

The geometric and kinematic definitions of controlled variables used in the arm were selected and adapted to fit with this specific robot arm, largely based on the previous computer simulations. While we suspect similar variables might be found in human arm control, we have no direct evidence to support the claim. Following the performance of the robot arm in this study, we suggest that an architecture featuring hierarchical arrangement of controlled variables might be a plausible solution for biological arm control.

Limitations and Perspectives

A limitation of the present study is the lack of direct comparison of robot behavior to human behavior in the same tasks, and instead comparing invariances and trends. The mechanical and sensory properties of the arm were not on par with the human sensory-motor system to allow such a comparison. With the aim of creating a higher fidelity model of a visually, tactually, and proprioceptively controlled human arm, the improvements would make the arm slightly faster and the sensors more numerous, but maybe not more precise. The improvements would not remove transport delays or noise, because those properties are present in biological arm control systems.

Mechanically, backlash in the geartrain of the motors (also known as slop or play) seems to be a major obstacle for human-like movement, as it puts a hard limit on the precision and bandwidth of the system that cannot be improved by higher quality sensors. With all the non-linearities, slowness, and fatigability in human muscles, human joints are backlash-free. Therefore, a higher fidelity model should put an emphasis on removing the backlash from the joints, perhaps by tendon-driven actuators.

The visual system of the present robot is a crude approximation of the human visual system's object detection in two dimensions. Accurate modeling of visual delays should be maintained, but the resolution and refresh rate could be improved, as well as adding stereo vision for three-dimensional localization. Improvements in the same direction could be made to proprioceptive and haptic sensory systems. In sum, such improved systems would allow testing hypotheses of lower, spinal-level sensorimotor loops, and their interaction with higher-level visual or proprioceptive loops, multi-sensory integration etc.

Additionally, as in studies of human movement, the results are influenced by low-pass filtering the data in the analysis stage, and should be taken with some reserve.

CONCLUSION

This research has shown that in a robot arm system with a hierarchical control architecture based on simulations by Powers (1999, 2008) several features characteristic of biological movement naturally emerge. The robot is robust to noise, delays and some nonlinearities. We found isochrony and bell-shaped velocity profiles in straight reaching movements and the speed-curvature power law in the fast drawing of ellipses. We showed how they can be achieved without trajectory planning, learning or online optimization. We also showed that a hierarchy of controlled variables can produce a motor equivalence phenomenon, where the robot performs the same visual task either with the wrist freely moving or with the wrist blocked. The system also adapts to different angles of the graphics tablet tilt by relying on pressure and pen angle control. Moreover, the system adapts to extending the arm with a tool, and to rotations of the visual field. Overall, we have demonstrated that our 4 DOF robot arm reproduces important features of human movement and therefore presents an appealing platform upon which to build and test further models of adaptive behavior, while providing insight into feasible mechanisms of human arm control.

DATA AVAILABILITY STATEMENT

The raw data supporting the conclusions of this article will be made available by the authors, without undue reservation.

AUTHOR CONTRIBUTIONS

PV and AM conducted a preliminary study. AG-M and AM redesigned the project in its current form and edited the final version of the manuscript. AG-M supervised the research. AM built the arm system, wrote the control software, did the experiments, analyzed the data, performed the modeling, and wrote the first complete manuscript draft. All authors approved the final version of the manuscript.

FUNDING

ACKNOWLEDGMENTS

REFERENCES

Atkeson, C. G., and Hollerbach, J. M. (1985). Kinematic features of unrestrained vertical arm movements. *J. Neurosci.* 5, 2318–2330. doi: 10.1523/JNEUROSCI.05-09-02318.1985

Bernstein, N. A. (1967). *The Co-ordination and Regulation of Movements*. Oxford: Pergamon Press.

Brooks, R. (1986). A robust layered control system for a mobile robot. *IEEE J. Robotics Automat.* 2, 14–23. doi: 10.1109/JRA.1986.1087032

Brooks, R. A. (1992). "Artificial life and real robots," in *Proceedings of the First European Conference on Artificial Life*, Cambridge, MA. 3–10.

Bruton, M., and O'Dwyer, N. (2018). Synergies in coordination: a comprehensive overview of neural, computational, and behavioral approaches. *J. Neurophysiol.* 120, 2761–2774. doi: 10.1152/jn.00052.2018

Bullock, D., Grossberg, S., and Guenther, F. H. (1993). A self-organizing neural model of motor equivalent reaching and tool use by a multijoint arm. *J. Cogn. Neurosci.* 5, 408–435. doi: 10.1162/jocn.1993.5.4.408

Castiello, U., and Dadda, M. (2019). A review and consideration on the kinematics of reach-to-grasp movements in macaque monkeys. *J. Neurophysiol.* 121, 188–204. doi: 10.1152/jn.00598.2018

Cisek, P. (2019). Resynthesizing behavior through phylogenetic refinement. *Attent. Percept. Psychophys.* 81, 2265–2287. doi: 10.3758/s13414-019-01760-1

Cisek, P., and Kalaska, J. F. (2002). Modest gaze-related discharge modulation in monkey dorsal premotor cortex during a reaching task performed with free fixation. *J. Neurophysiol.* 88, 1064–1072. doi: 10.1152/jn.00995.2001

Desmurget, M., and Grafton, S. (2000). Forward modeling allows feedback control for fast reaching movements. *Trends Cogn. Sci.* 4, 423–431. doi: 10.1016/S1364-6613(00)01537-0

Feldman, A. G. (2015). *Referent Control of Action and Perception. Challenging Conventional Theories in Behavioral Neuroscience.* Springer Science+Business Media, New York. doi: 10.1007/978-1-4939-2736-4

Fitts, P. M. (1954). The information capacity of the human motor system in controlling the amplitude of movement. *J. Exp. Psychol.* 47:381. doi: 10.1037/h0055392

Flash, T., and Hogan, N. (1985). The coordination of arm movements: an experimentally confirmed mathematical model. *J. Neurosci.* 5, 1688–1703. doi: 10.1523/JNEUROSCI.05-07-01688.1985

Floreano, D., Ijspeert, A. J., and Schaal, S. (2014). Robotics and neuroscience. *Curr. Biol.* 24, R910–R920. doi: 10.1016/j.cub.2014.07.058

Gribble, P. L., and Ostry, D. J. (1996). Origins of the power law relation between movement velocity and curvature: modeling the effects of muscle mechanics and limb dynamics. *J. Neurophysiol.* 76, 2853–2860. doi: 10.1152/jn.1996.76.5.2853

Guiard, Y. (2009). "The problem of consistency in the design of Fitts' law experiments: consider either target distance and width or movement form and scale," in *Proceedings of the Sigchi Conference on Human Factors in Computing Systems*, Boston, MA. 1809–1818. doi: 10.1145/1518701.1518980

Hadjiosif, A. M., Krakauer, J. W., and Haith, A. M. (2021). Did we get sensorimotor adaptation wrong? Implicit adaptation as direct policy updating rather than forward-model-based learning. *J. Neurosci.* 41, 2747–2761. doi: 10.1523/JNEUROSCI.2125-20.2021

Harris, C. M., and Wolpert, D. M. (1998). Signal-dependent noise determines motor planning. *Nature* 394, 780–784. doi: 10.1038/29528

Hughlings Jackson, J. (1884). The Croonian lectures on evolution and dissolution of the nervous system. *Brit Med J.* 1:703. doi: 10.1136/bmj.1.1213.591

Hughlings Jackson, J. (1958). "Evolution and dissolution of the nervous system," in *Selected writings of John Hughlings Jackson*, Vol. 2, ed J. Taylor (London: Staples Press), 45–75.

Huh, D., and Sejnowski, T. J. (2015). Spectrum of power laws for curved hand movements. *Proc. Natl. Acad. Sci. U.S.A.* 112, E3950–E3958. doi: 10.1073/pnas.1510208112

Inoue, Y., Mao, H., Suway, S. B., Orellana, J., and Schwartz, A. B. (2018). Decoding arm speed during reaching. *Nat. Commun.* 9:5243. doi: 10.1038/s41467-018-07647-3

Kawato, M. (1999). Internal models for motor control and trajectory planning. *Curr. Opin. Neurobiol.* 9, 718–727. doi: 10.1016/S0959-4388(99)00028-8

Lacquaniti, F., Terzuolo, C., and Viviani, P. (1983). The law relating the kinematic and fgural aspects of drawing movements. *Acta Psychol.* 54, 115–130 doi: 10.1016/0001-6918(83)90027-6

Latash, M. L. (2008). *Synergy.* New York, NY: Oxford University Press. doi: 10.1093/acprof:oso/9780195333169.001.0001

Latash, M. L., Scholz, J. P., and Schöner, G. (2007). Toward a new theory of motor synergies. *Motor Control* 11, 276–308. doi: 10.1123/mcj.11.3.276

Loeb, G. E. (2012). Optimal isn't good enough. *Biol. Cybern.* 106, 757–765. doi: 10.1007/s00422-012-0514-6

Lopes, G. C. (2016). A robust role for motor cortex. *bioRXiv.* PhD Thesis. doi: 10.1101/058917

Lou, J. S., and Bloedel, J. R. (1988). A new conditioning paradigm: conditioned limb movements in locomoting decerebrate ferrets. *Neurosci. Lett.* 84, 185–190. doi: 10.1016/0304-3940(88)90405-3

Lou, J. S., and Bloedel, J. R. (1992). Responses of sagittally aligned Purkinje cells during perturbed locomotion: synchronous activation of climbing fiber inputs. *J. Neurophysiol.* 68, 570–580. doi: 10.1152/jn.1992.68.2.570

Matic and Gomez-Marin (2019). "Trajectory production without a trajectory plan," in *9th International Symposium on Adaptive Motion of Animals and Machines (AMAM 2019)*. Lausanne, Switzerland.

McNamee, D., and Wolpert, D. M. (2019). Internal models in biological control. *Ann. Rev. Control Robotics Autonomous Syst.* 2, 339–364. doi: 10.1146/annurev-control-060117-105206

Mekyska, J., Faundez-Zanuy, M., Mzourek, Z., Galaz, Z., Smekal, Z., and Rosenblum, S. (2017). Identification and rating of developmental dysgraphia by handwriting analysis. *IEEE Transact. Human-Machine Syst.* 47, 235–248. doi: 10.1109/THMS.2016.2586605

Merel, J., Botvinick, M., and Wayne, G. (2019). Hierarchical motor control in mammals and machines. *Nat. Commun.* 10:5489. doi: 10.1038/s41467-019-13239-6

Miall, R. C., Weir, D. J., Wolpert, D. M., and Stein, J. F. (1993). Is the cerebellum a smith predictor? *J. Mot. Behav.* 25, 203–216. doi: 10.1080/00222895.1993.9942050

Morimoto, J., and Kawato, M. (2015). Creating the brain and interacting with the brain: an integrated approach to understanding the brain. *J. R. Soc. Interface* 12:20141250. doi: 10.1098/rsif.2014.1250

Mounoud, P., Viviani, P., Hauert, C. A., and Guyon, J. (1985). Development of visuomanual tracking in 5-to 9-year-old boys. *J. Exp. Child Psychol.* 40, 115–132. doi: 10.1016/0022-0965(85)90068-2

Müller, J., Oulasvirta, A., and Murray-Smith, R. (2017). Control theoretic models of pointing. *ACM Transact. Computer-Human Interact.* 24, 1–36. doi: 10.1145/3121431

Parker, M. G., Tyson, S. F., Weightman, A. P., Abbott, B., Emsley, R., and Mansell, W. (2017). Perceptual control models of pursuit manual tracking demonstrate individual specificity and parameter consistency. *Attent. Percept. Psychophys.* 79, 2523–2537. doi: 10.3758/s13414-017-1398-2

Parker, M. G., Willett, A. B., Tyson, S. F., Weightman, A. P., and Mansell, W. (2020). A systematic evaluation of the evidence for Perceptual Control Theory in tracking studies. *Neurosci. Biobehav. Rev.* 112, 616–633. doi: 10.1016/j.neubiorev.2020.02.030

Plooij, F. X., and van de Rijt-Plooij, H. H. (1990). Developmental transitions as successive reorganizations of a control hierarchy. *Am. Behav. Sci.* 34, 67–80. doi: 10.1177/0002764290034001007

Powers, W. T. (1973). *Behavior: The Control of Perception.* Chicago, IL: Aldine.

Powers, W. T. (1999). A model of kinesthetically and visually controlled arm movement. *Int. J. Hum. Comput. Stud.* 50, 463–479. doi: 10.1006/ijhc.1998.0261

Powers, W. T. (2008). *Living Control Systems III: The Fact of Control.* Simulations accompanying the book. Available online at: http://www.livingcontrolsystems.com/lcs3/content_lcs3.html (accessed October, 2021).

Prescott, T. J., Redgrave, P., and Gurney, K. (1999). Layered control architectures in robots and vertebrates. *Adapt. Behav.* 7, 99–127. doi: 10.1177/105971239900700105

Profeta, V. L., and Turvey, M. T. (2018). Bernstein's levels of movement construction: a contemporary perspective. *Hum. Mov. Sci.* 57, 111–133. doi: 10.1016/j.humov.2017.11.013

Sartori, L., Camperio, A., Bulgheroni, M., and Castiello, U. (2013). Reach-to-grasp movements in *Macaca fascicularis* monkeys: the Isochrony Principle at work. *Front. Psychol.* 4:114. doi: 10.3389/fpsyg.2013.00114

Schaal, S., and Sternad, D. (2001). Origins and violations of the 2/3 power law in rhythmic three-dimensional arm movements. *Exp. Brain Res.* 136, 60–72. doi: 10.1007/s002210000505

Scott, S. H. (2004). Optimal feedback control and the neural basis of volitional motor control. *Nat. Rev. Neurosci.* 5, 532–545. doi: 10.1038/nrn1427

Scott, S. H. (2012). The computational and neural basis of voluntary motor control and planning. *Trends Cogn. Sci.* 16, 541–549. doi: 10.1016/j.tics.2012.09.008

Shadmehr, R., and Krakauer, J. W. (2008). A computational neuroanatomy for motor control. *Exp. Brain Res.* 185, 359–381. doi: 10.1007/s00221-008-1280-5

Soechting, J. F. (1984). Effect of target size on spatial and temporal characteristics of a pointing movement in man. *Exp. Brain Res.* 54, 121–132. doi: 10.1007/BF00235824

Todorov, E., and Jordan, M. (2002b). A minimal intervention principle for coordinated movement. *Adv. Neural Inf. Process. Syst.* eds Becker, Thrun, Obermayer, MIT Press (2003). 15, 27–34.

Todorov, E., and Jordan, M. I. (2002a). Optimal feedback control as a theory of motor coordination. *Nat. Neurosci.* 5, 1226–1235. doi: 10.1038/nn963

Uno, Y., Kawato, M., and Suzuki, R. (1989). Formation and control of optimal trajectory in human multijoint arm movement. *Biol. Cybern.* 61, 89–101. doi: 10.1007/BF00204593

Viviani, P. (1988). "Two-dimensional Visuo-manual Tracking Beyond The System Analysis Approach," in *Proceedings of the 1988 IEEE International Conference on Systems, Man & Cybernetics*, Vol. 1, (IEEE), Beijing, China. 404–412.

Viviani, P., Campadelli, P., and Mounoud, P. (1987). Visuo-manual pursuit tracking of human two-dimensional movements. *J. Exp. Psychol.* 13, 62–78. doi: 10.1037/0096-1523.13.1.62

Viviani, P., and Flash, T. (1995). Minimum-jerk, two-thirds power law, and isochrony: converging approaches to movement planning. *J. Exp. Psychol.* 21, 32–53. doi: 10.1037/0096-1523.21.1.32

Viviani, P., and McCollum, G. (1983). The relation between linear extent and velocity in drawing movements. *Neuroscience* 10, 211–218. doi: 10.1016/0306-4522(83)90094-5

Viviani, P., and Mounoud, P. (1990). Perceptuomotor compatibility in pursuit tracking of two-dimensional movements. *J. Mot. Behav.* 22, 407–443. doi: 10.1080/00222895.1990.10735521

Viviani, P., and Terzuolo, C. (1982). Trajectory determines movement dynamics. *Neuroscience* 7, 431–437 doi: 10.1016/0306-4522(82)90277-9

Webb, B. (2001). Can robots make good models of biological behaviour?. *Behav. Brain Sci.* 24, 1033–1050. doi: 10.1017/S0140525X01000127

Whelan, P. J. (1996). Control of locomotion in the decerebrate cat. *Prog. Neurobiol.* 49, 481–515. doi: 10.1016/0301-0082(96)00028-7

Wolpert, D. M., and Kawato, M. (1998). Multiple paired forward and inverse models for motor control. *Neu net.* 11:1317–1329. doi: 10.1016/S0893-6080(98)00066-5