Reinforcement Learning and Synergistic Control of the ACT Hand

Eric Rombokas, Mark Malhotra, Evangelos Theodorou, Emo Todorov and Yoky Matsuoka

Abstract—Tendon-driven systems are ubiquitous in biology and provide considerable advantages for robotic manipulators, but control of these systems is challenging because of the increase in dimensionality and intrinsic nonlinearities. Researchers in biological movement control have suggested that the brain may employ “muscle synergies” to make planning, control, and learning more tractable by expressing the tendon space in a lower-dimensional virtual synergistic space. We employ synergies which respect the differing constraints of actuation and sensation, and apply path integral reinforcement learning in the virtual synergistic space as well as the full tendon space. Path integral reinforcement learning has been used successfully on torque-driven systems to learn episodic tasks without using explicit models, which is particularly important for difficult-to-model dynamics like tendon networks and contact transitions. We show that optimizing a small number of trajectories in virtual synergy space can produce comparable performance to optimizing the trajectories of the tendons individually. The six tendons of the index finger and eight tendons of the thumb, each actuating four degrees of joint freedom, are used to slide a switch and turn a knob. The learned control strategies provide a method for discovery of novel task strategies and system phenomena without explicitly modeling the physics of the robot and environment.

Index Terms—Biologically inspired control, Synergies, Reinforcement Learning, Tendon driven control.

1 INTRODUCTION

Tendon-driven systems provide unique advantages for space-constrained mechanisms like robotic hands. They allow actuators to be placed at a distance from the end-effector, reducing inertia and allowing for faster movements and improved force control [1]. Further, they have little intrinsic backlash and friction.

Compared to torque-driven systems, however, the inverse dynamics solution for a tendon-driven system is more complicated. Because tendons may only transmit force by pulling, a single tendon cannot control a one-degree-of-freedom joint. This increases the dimensionality of the problem not only in the degrees of freedom available, but in the number of control algorithm parameters necessary to handle the complexity. Tendon routing over irregular bone shapes exhibits nonholonomicity, nonlinear pose-dependent variation of the moment arms, and the interaction of tendons through a complex web of sub-tendons is generally hard to measure and model. Nonlinear effects are compounded by incomplete sensing; human joint angle sensation is achieved indirectly by sensors in the muscles and skin, not by direct measurement of joint angle.

It is hypothesized that neural systems surmount these challenges in part by reducing the dimensionality of the problem. Implicit in this hypothesis is the assumption that control will be more successful or easier to learn in the reduced-dimension space, but the particulars of the control algorithms used by the brain are still unclear. We have developed a technique for the tendon-driven Anatomically Correct Testbed (ACT) robotic hand [2] which expresses the tendon space in a lower-dimensional virtual synergistic space. These “motor synergies”
provide a first-order model which captures the dynamics of actual sensor recordings of the robot in the environment. Data-driven construction of observation (applying to sensors) and actuation (applying to actuators) synergies allows dimensionality reduction, but it remains necessary to learn control.

In this work we use Policy Improvement with Path Integrals or (PI$^2$) [3], [4] for learning control. This is one of a class of model-free reinforcement learning algorithms which learn movement control without having to explicitly model uncertain task dynamics. It also allows samples of the state space to be concentrated around an initial demonstrated or previously learned strategy. Advantages like these are critical for dealing with the immense dimensionality and stochasticity encountered by tendon-driven biological and robotic systems in unstructured environments. Previously, application domains have been limited to torque-driven robotic systems, but the additional challenges of tendon actuation constitute an important and previously unmet challenge for model-free reinforcement learning.

A pilot study has shown promise using PI$^2$ to learn a full tendon-space control policy for a switch-sliding manipulation task [5]. Here we extend and expand those results to confront a more challenging knob-turning task and less informative cost feedback, as well as incorporating tendon synergies. This research is the first successful model-free learning of manipulation tasks using a real-world tendon-driven robot. Inspired by the success of the method on the full-tendon problem, we apply it to control in the reduced-dimensional synergy space. We find that the same manipulation tasks may be learned with comparable performance in the virtual synergy space, but for these experiments we observe no unequivocal advantage compared to full-tendon learning. We observe, however, that the synergy-based controller learns an interesting strategy for knob turning which was not discovered in the full-tendon version of the experiments.

This work provides a successful reinforcement learning approach for manipulation tasks on the tendon-driven ACT hand and extends this method to include synergy-based dimensionality reduction respecting the differing constraints of actuation and sensation. We report insights into unknown tendon phenomena provided by the learned control strategies, and the discovery an improved strategy beyond that seen in the initial demonstration.

2 RELATED WORK
A few standard actuator configurations are employed for tendon-driven robotic hand design, generally allowing tractable mappings between tendon forces and the resultant joint torques [6]. Some tendon-driven designs handle non-negativity by using two tendons per joint [7] [8], or use a single extensor antagonized by a flexor for each joint [9]. Each of these designs makes different engineering choices with the aim of producing a capable robotic hand.

In contrast, the goal of the ACT Hand is to closely mimic human hand biomechanics. The human hand is subject to redundancies, high dimensionality, constrained controls, and profound nonlinearities [10] that make both state estimation and planning/control problems a great deal more challenging. The tendons interact in complex and difficult-to-model ways, and as they slide over the irregular surfaces of the bones these interactions change nonlinearly. Neuroscientists and biomechanics researchers discuss methods by which the brain and body surmount these challenges, and in particular here we consider the Synergy Hypothesis.

The Synergy Hypothesis is a much-debated concept in the study of neuromuscular control, and addresses the dimensionality challenge of human-like tendon and joint dynamics. The core of the idea is that the nervous system reduces the degrees of freedom of the control, learning, or planning problem. Rather than independently activating muscles, groups of muscles are used in a coordinated manner corresponding to the kinematic [11], [12] or dynamic [13] regularities of the system itself or a task in the environment [14].

Studies have shown that synergies are consistent with aspects of neuromuscular control: for instance, frog leg wiping trajectories can be composed from premotor drive pulses [15] and a small subspace of possible movements is
used by people when manipulating an object [11]. A number of matrix factorization algorithms have been applied to such analyses [16], but here we use a method developed for the ACT Hand which differentiates between observation and actuation [2]. The reduction of actuation dimensionality must be non-negative due to the tendon properties, but the reduction of observation dimensionality need not be.

In [3], [17], [4] the generalized path integral control framework was presented and applied to stochastic dynamics with state dependent control transition and diffusion matrices. Policy Improvement With Path Integrals (PI²) is an iterative version of path integral control. PI² was applied to variable stiffness control (equivalent to autonomously tuning PD gains in a 6DOF manipulator)[18], learning a full-body skill on a humanoid in simulation [19], manipulation skill acquisition on the PR2 [20], hierarchical reinforcement learning [21], learning robust manipulation [22], and learning to grasp under uncertainty [23]. In all of the aforementioned work, however, PI² was applied only to torque-driven systems.

3 Framework

The Anatomically Correct Testbed (ACT) robotic hand mimics the interactions among muscle excursions and joint movements produced by the bone and tendon geometries of the human hand. This mimicry results in a robotic system sharing the redundancies and nonlinearities of the biological hand [24] [10].

The ACT hand uses 24 motor-driven tendons to control a thumb, index finger, middle finger, and wrist. Each segment of these fingers is machined using human bone data, and is accurate in surface shape, mass, and center-of-gravity to the human equivalent. The extensor mechanisms are webs of tendons on the dorsal side of the fingers and are crucial for emulating dynamic human behavior [25]. As each tendon is pulled by a motor, it is routed through attachment points mimicking human tendon sheaths and following the contours of the bones. Since these bone shapes are complicated surfaces, the effective moment arm the tendon exerts on the joint varies with joint angle [10].

Control is learned by a direct reinforcement learning method, PI², beginning with a demonstrated trajectory which yields initial policy parameters θ. The policy provides a set of desired trajectories, which are followed by an underlying controller, the performance of which is lumped with the task dynamics as illustrated in Figure 1. Each revision of θ we refer to as a trial. A sample trajectory is queried from the system by sampling δθ, and actually performing an episode of the task using the resulting θ. Each of these exploratory executions of the task is a rollout. To revise θ at the end of a trial, each sampled control variation is weighted according to the cost encountered by the corresponding rollout.

Trajectories are represented via Dynamic Movement Primitives (DMPs), which are a set of dynamic equations used to generate smooth movement trajectories. These evolve as a point attractor to the goal, and the shape of the tra-
jectories can be changed and learned according to a set of parameters $\theta$ [26].

Fig. 2: Effects of varying $\theta$. For clarity, this primitive uses only two basis functions, evenly spaced along the phase of the movement. The unforced transformation system dynamics are evident as the black solid line, and two example forced trajectories are pictured in red and blue-dashed, where $\theta_0 = [0.0], \theta_1 = [-1.0 \ 0.0]$, and $\theta_2 = [1.0 \ -1.0]$. Positive and negative values of theta perturb the trajectory above and below the unforced dynamics, as seen at the beginning of trajectories 1 and 2. For $\theta$ values of zero, the trajectory will converge as an attractor to the unforced dynamics, as seen at the end of trajectory 1.

The DMP consists of two sets of differential equations, the canonical and transformation system which are coupled through a nonlinearity [26] which delivers a learnable perturbation on the natural attractor dynamics. The canonical system is formulated as $\frac{1}{\tau} \dot{x}_t = -\alpha x_t$. This can be thought of as a describing the phase of the movement. Starting from the initial state $\phi_0 = 1$, the phase $\phi$ converges monotonically to zero.

The transformation system consists of the following two differential equations:

$$\tau \dot{z} = \alpha_z \beta_z \left( g + \frac{f(\phi)}{\alpha z \beta z} - y \right) - \alpha z z \tag{1}$$

$$\tau \dot{y} = z$$

Where $y_t$ and $\dot{y}_t$ are position and velocity, and $\alpha_z, \beta_z, \tau$ are time constants which affect the flexibility and stability of the attractor. The nonlinear coupling or forcing term $f \in \mathbb{R}$ is:

$$f(\phi) = \sum_{i=1}^{M} \frac{K(\phi_t, c_i) \theta_i \phi_t (g - y_0)}{K(\phi_t, c_i)} = \Phi_P(\phi)^T \theta \tag{2}$$

where $M$ is the number of the basis functions. The basis functions $K(\phi_t, c_i)$ are Gaussian kernels: $K(\phi_t, c_i) = \exp \left( -0.5 h_j (\phi_t - c_j)^2 \right)$ with bandwidth $h_j$ and center $c_j$. [26]. The vector $\Phi_P(\phi) \in \mathbb{R}^{M \times 1}$ contains all the Gaussian kernels. Control of the DMP shape, then, is achieved by changing the weights $\theta \in \mathbb{R}^{M \times 1}$ on each basis function. Since the basis functions are functions of phase, each weight can be thought of as how much perturbation the transformation function receives at a particular phase of the movement. Figure 2 shows an example.

For the tendon-space experiments, the learning loop sets desired tendon length trajectories, via DMPs, which are then achieved by a PID controller. The synergistic controller sets desired observation synergy positions, again via DMPs, and achieves them through direct torque control using activation synergies, as described in Section 3.2. For both these cases ( tendon-space and virtual synergy space ), we learn four tasks, as described in Section 4.

### 3.1 \textit{PT}^2 Learning

Consider the stochastic optimal control problem of minimizing the cost function:

$$V(x) = \min_u J(x, u) = \min_u \int_{t_0}^{t_N} \mathcal{L}(x, u, t) dt \tag{3}$$

subject to the nonlinear stochastic dynamics:

$$dx = \alpha(x) dt + C(x) u dt + B(x) \delta \omega \tag{4}$$

with $x \in \mathbb{R}^{p \times 1}$ denoting the state of the system, $u \in \mathbb{R}^{p \times 1}$ the control vector and $\delta \omega \in \mathbb{R}^{p \times 1}$ brownian noise. The function $\alpha(x) \in \mathbb{R}^{p \times 1}$ is the drift, which can be a nonlinear function of the state $x$. The matrix $C(x) \in \mathbb{R}^{p \times p}$ is the control and $B(x) \in \mathbb{R}^{p \times p}$ is the diffusion matrix. Under the optimal controls $u^*$ the cost function is equal to the value function $V(x)$. \( \mathcal{L}(x,u,t) \), the immediate cost, is expressed as:

$$\mathcal{L}(x, u, t) = q(x, t) + \frac{1}{2} u^T Ru \tag{5}$$

The immediate cost has two terms: the first $q(x_t, t)$ is an arbitrary state-dependent cost, and the second is the control cost with weight $R \in \mathbb{R}^{p \times p}$. The function $q(x, t)$ is a nonnegative definite function of $x$ and $t$. The matrix $R$ is a symmetric positive definite matrix.
integral optimal control takes the form:

\[ \text{to be the most difficult step in the theory of} \]

is to find the value function which turns out

be solved under an exponential transformation
tic optimal control and planning problems can

and KL control in [30] have shown that stochas-

equation is a partial differential equation that

Bellman (HJB) equation [27], [28]. The HJB

the optimal controls are specified as follows:

\[ V \]

space with small values of

is to push the systems towards parts of the state

function. Thus, the outcome of optimal control

negative direction of the gradient of the value

controls should move the systems towards the


\[ \text{TABLE 1: Policy Improvement With Path Integrals} \]

- **Given:**
  - An immediate state dependent cost function \( q(x_i) \)
  - Time horizon \( N \), weights \( M \) and \( R \), parameter \( \lambda \), and noise variance \( \Sigma = \lambda R^{-1} \)
  - **Repeat** until convergence of the trajectory cost:
    - Create \( K \) roll-outs of the system from the same start
      state \( x_0 \) using stochastic parameters \( \theta + \delta \theta \), at every
      time step \( t \), where \( \delta \theta \) are sampled from a gaussian
distribution with variance \( \Sigma \).
    - For \( k = 1 \ldots K \), compute costs and weights:
      - \( S(\tau_i) = \phi(x_N) + \sum_{j=1}^{N-1} (q_j + \delta \theta_j^T M \delta \theta_j) dt \)
      - \( P(\tau_i, k) = e^{-\frac{1}{2} S(\tau_i, k)} \)
    - For \( i = 1 \ldots (N - 1) \), compute:
      - \( \delta \theta^* (x_{i+1}) = \sum_{k=1}^{K} P(\tau_{i,k}) \delta \theta(i, k) \)
    - Time averaging
      - \( \delta \theta_T^* = \frac{1}{N-1} \sum_{i=1}^{N-1} \delta \theta^* (x_{i+1}) \)
    - Update \( \theta \leftarrow \theta + \delta \theta_T^* \)

\( \mathbb{R}^{p \times p} > 0 \). To find the optimal control \( u(x, t) \) one

needs to know the value function \( V(x, t) \). Then

the optimal controls are specified as follows:

\[ u(x, t) = -R^{-1}C(x)^T \nabla_x V(x, t) \]  

(6)

The equation (6) tell us that the optimal

controls should move the systems towards the

negative direction of the gradient of the value

function. Thus, the outcome of optimal control

is to push the systems towards parts of the state

space with small values of \( V(x, t) \). The last step

is to find the value function which turns out

to be the most difficult step in the theory of

nonlinear optimal control. The difficulty arises

from the fact that to find the value functions

\( V(x, t) \) one has to solve the Hamilton-Jacobi-

Bellman (HJB) equation [27], [28]. The HJB

equation is a partial differential equation that

is nonlinear and second order and its solutions

can be found for low dimensional problems.

Recent work on path integral work in [3], [29]

and KL control in [30] have shown that stochas-

tic optimal control and planning problems can

be solved under an exponential transformation

of the value function.

In [3], [4] it has been shown that the path

integral optimal control takes the form:

\[ u_{PI}(x_{i}) dt = \lim_{dt \to 0} \int P(\tau_i) \delta \omega_{ti} \]  

(7)

where \( \tau_i \) is a trajectory in state space starting

from \( x_i \) and ending in \( x_N \), so \( \tau_i = (x_i, \ldots, x_N) \).

The probability \( P(\tau_i) \) is defined as

\[ P(\tau_i) = \frac{e^{-\frac{1}{2} S(\tau_i)}}{\int e^{-\frac{1}{2} S(\tau_i)} d\tau_i} \]  

(8)

where the term \( S(\tau_i) \) is defined as:

\[ S(\tau_i) = \phi(x_N) + \sum_{j=1}^{N-1} (q(x_j) dt + \delta \omega_{ji}^T M \delta \omega_{ji}) \]  

(9)

and \( M = B(x)^T (C(x)R^{-1}C(x)^T)^{-1} B(x) \). 

Essentially the optimal control is an average of

variations \( \delta \omega \) weighted by their probabilities.

This probability is inversely proportional to the

path cost according to (8) and (9). Low-cost

paths have high probability and vice versa.

The goal of PI\(^2\) is not to solve the optimal

control problem at once since that would re-

quire full sampling of high dimensional state

spaces. Instead we start with an initial policy

which is iteratively updated (Table 1). Start

with an initial policy parameterized as \( u = \theta^T \Phi(x, t) \). New trajectories can be generated by

sampling policy parameters \( \delta \omega_{ti} = \delta \theta_{ti} \). The cost for each resultant trajectory is computed

based on (9), allowing computation of the op-
timal variation \( \delta \theta_t^* \) according to (6). The policy

parameters are updated as \( \theta = \theta + \delta \theta_T^* \) where

\( \delta \theta_T^* \) is the time average of \( \delta \theta_t^* \).

An important element of PI\(^2\) is that optimal

controls only partially depend on the stochastic
dynamics. In particular they do not depend

on the drift of the stochastic dynamics in (4),

and in that sense it is a model-free method.

In addition, with the use of DMPs we are also

able to represent desired trajectories without
directly modelling the robot and environment.

PI\(^2\) as formulated above optimizes only a

single policy parameter vector \( \theta \), but it is easy
to consider multiple simultaneous PI\(^2\) agents

working in concert, each learning the policy

for a single dimension or tendon. The agents
do not interact directly, but instead learn to
coordinate their action through their shared
dependence on task performance.
Fig. 3: Observation (left) and activation synergies (right). These represent 99% of the kinematic variance of the knob task demonstration. Tendons FDS and FDP act as flexors; EI, RI, and PI act as extensors and ab/aductors; and the lumbrical (LUM) is an abductor but switches from extensor to flexor depending on finger posture and which joint is considered.

3.2 Synergistic Control

Although the manipulation tasks presented here involve unmodeled and stochastic dynamics, they are kinematically fairly simple. The demonstrated trajectories show a large degree of coupled motion among the tendons. This regular structure in the tendon trajectories suggests that applying task-specific dimensionality reduction could improve learning speed and convergence without much loss in performance. Additionally, testing synergies for control in the context of reinforcement learning can provide different insights than analysis of animal behavior.

We review here a synergistic control framework [2] that 1) reduces the dimensionality of observation and control, 2) enables controllability in this reduced space, and 3) provides an effective method for adaptively revising the forward model to combat any model mismatch.

Consider a tendon-driven system composed of \( n \) torque-driven motors, each with an encoder to measure displacements, \( L \). First-order system dynamics can be expressed in discrete state space formulation as:

\[
L_{k+1} = AL_k + BT_k, \tag{10}
\]

such that \( L_k \in \mathbb{R}^{n \times 1} \) is the state vector at timestep \( k \), \( T_k \in \mathbb{R}^{n \times 0} \geq 0 \) is the non-negative control vector, \( A \in \mathbb{R}^{n \times n} \) is the state transition matrix, and \( B \in \mathbb{R}^{n \times n} \) the control matrix.

Let the observation synergy \( S \in \mathbb{R}^{s \times n} \) with \( (s < n) \) be an orthonormal matrix that is used to project \( L \) into a \( s \)-dimensional virtual synergistic space. Define \( x \in \mathbb{R}^s \) as the “synergy position”: \( x = SL \), a low-dimensional representation of the measured tendon lengths. Since \( S \) is orthonormal, the inverse operation is easily computed. A target in tendon length space can be reconstructed from the target synergy position by: \( \hat{L} = S^T \hat{x} \).

This definition differs slightly from standard postural synergies [11]: tendon lengths are the input rather than joint angles. This is analogous to the human’s proprioceptive feedback from muscle spindles [31], considered the primary input for awareness of joint position in the absence of vision [32], [33]. Consequently, tendon-space synergies may account for both inter-joint coupling and for the coupled movement of multiple tendons that results from a single joint movement.

The activation synergy \( W \in \mathbb{R}^{n \times w} \) with \( (s < w < n) \) is non-negative matrix used to project the “synergy activation,” \( u \in \mathbb{R}^w \), into torque commands for each of the \( n \) motors: \( T = Wu \). Thus, activating a single component of \( u \) produces a weighted activation of a set of motors.

Applying both the observation and control reduction models significantly simplifies the system. Now, first-order system dynamics are expressed in discrete state space as:

\[
x_{k+1} = \hat{A}x_k + \hat{B}u_k. \tag{11}
\]

The state transition and control matrices are no longer \( n \times n \), but instead \( \hat{A} \in \mathbb{R}^{s \times s} \) and \( \hat{B} \in \mathbb{R}^{s \times w} \).

Joint-space tendon control methods[1] can be applied to the virtual synergistic space system with the synergy position substituted for joint angle and the synergy activation substituted for tendon force. Analogously to joint torque, the "synergy torque" can be defined as \( \tau_S = J^T u \) where \( J \) is the Jacobian. Assuming first-order dynamics in the virtual synergistic space, \( \hat{B} = J^T \), \( \hat{A} \) is the identity matrix, and \( \tau_S \) can be estimated from the change in synergy position:
\( \delta x = J^T u \). The system is then virtual-space-controllable if the non-negative rank of \( \tilde{B} > s \).

A simple controller is achieved using the efficient non-negative least squares algorithm [34] at every timestep to select synergy activations to achieve target synergy position displacements:

\[
\hat{u} = \arg\min_u \| J^T u - \hat{\delta}x \| \text{ subject to } \hat{u} \geq 0 \quad (12)
\]

This algorithm minimizes the norm of the synergy activation and produces smooth control.

The Jacobian is estimated offline by linear regression from a set of identification data:

\[
J_0 = (UU^T)^{-1}UX^T, \quad (13)
\]

where \( U \in \mathbb{R}^{w \times N} \) is the matrix of input synergy activations, with \( N > w \) and \( X \) is the \( s \times N \) matrix of measured synergy torques over \( N \) timesteps. If the estimated Jacobian has full non-negative rank this indicates that the virtual-space system should be controllable.

A model-adaptive controller can then revise the estimated Jacobian as a trajectory is tracked through synergistic control. The revision is done using with a batched, damped recursive least squares [35] formulation of the form:

\[
J_{k+1} = J_k - c((UU^T)^{-1}U(J_k^T U - X)^T) \quad (14)
\]

where \( U \) and \( X \) are recorded over time horizon \( N \) and the damping coefficient \( c \) controls adaptation speed. The time horizon can be lengthened to combat noise and avoid non-singular matrix inversion (this condition is caught and \( J \) is not altered).

4 MANIPULATION TASKS

For all four tasks, the first step is a demonstration, in which a human moves the hand through the motion of either pushing the switch or turning the knob. The tendon excursions (6 index finger tendons and 8 thumb) produced by this externally-powered example grossly resemble those required for the robot to complete the task, but simply replaying them would not result in successful task completion for two reasons. Firstly, during demonstration the tendons are not loaded, which changes the configuration of the tendon network in comparison to when it is actively moving. More importantly, the tendon trajectories encountered during a demonstration do not impart information about the necessary forces required to accommodate task dynamics.

For the virtual synergy space experiments, these demonstrations are decomposed by applying PCA to determine the observation synergies \( S \), and decomposed by a rectified-PCA NMF algorithm [2] to find the activation synergies \( W \). The number of components are selected so that at least 99\% of the variance is captured, resulting in \( s=1 \) for Task 1, \( s=2 \) for Tasks 2 and 3, and \( s=4 \) for Task 4. As an example, Figure 3 shows the resulting synergies for Task 2.

The initial control parameters \( \theta_o \in \mathbb{R}^{50} \) of the DMPs (Section 3) are fit for each tendon or actuation synergy separately. This fit is achieved by initializing a random control \( \theta_i \), and apply-
ing \( PI^2 \) learning in simulation, in which the cost function reflects how closely the resultant DMP trajectory matches the observed tendon lengths or synergy positions of the demonstration.

Each of the switch tasks has different kinematic and dynamic requirements. Figure 4 illustrates the start position and the relative orientation of hand and switches.

**Task 1**: the index finger must flex from an extended position, contact a standard household sliding dimmer switch, and push it down. Transitioning from positioning the body in air to manipulation of an external object is notoriously difficult to model.

**Task 2**: the index finger must flex to perform a contact transition and exert force on a standard household rotary dimmer knob, then turn the knob clockwise. This is more complex than the first task because the required direction of force for knob-turning is tangent to the knob, but normal force must also be applied throughout the motion to prevent slipping.

**Task 3**: the index finger must rotate the knob using less informative cost feedback. The cost function (Eq. 5) used for the first two tasks reflects the continuous nature of the task by evaluating task completion (switch or knob position) every time step. The state-dependent cost term \( q_0(x,t) \) takes the form:

\[
q_0(x,t) = q v_t^2 + q_{term} v_{tN} \quad (15)
\]

where \( v_t \) is the voltage reading from the knob at time instance \( t \), \( q \) is the instantaneous state cost coefficient, \( q_{term} \) is the terminal state cost coefficient, and \( v_{tN} \) is the terminal state corresponding to the switch position after movement completion.

Many natural tasks, however, do not provide feedback until their completion, such as switching a on-off lightswitch instead of dimmers. To evaluate the possibility of learning in this more difficult situation, the third task is to turn the knob using \( v_t = 0 \). In this situation, there is no advantage for switching faster or earlier, but instead only the end position of the knob contributes to cost.

**Task 4**: we increase the dimensionality of the learning problem by adding the thumb (actuated by eight tendons) into the knob task. This fourth task is to turn the knob using both thumb and index finger, under the continuous cost function (Equation 15). Therefore Task 4 is like Task 2, but also using the thumb.

5 **Results**

The initial policies, which are equivalent to simply playing back the demonstration using a generic gain, are barely capable of moving the switches. For all tasks, however, the system improves performance to near the maximum allowed by the kinematics of the pose.

**Task 1**, the sliding switch, exhibits similar learning for the full-tendon and synergy controllers. The initial policy moves the switch about 0.7 cm, but after 10 trials the switch is pushed to the end of its range (2.75 cm). Additional improvement in performance is achieved after trial 10, as the movement occurs earlier and the reference trajectories become more aggressive, presumably so they can generate the necessary forces to accommodate the task dynamics. The converged learned strategies improved slightly more for the synergy case.

Using the index finger to rotate the knob (Task 2), the finger is initially able to slightly move the knob (5° for both controllers), but slips across the surface without moving it for the majority of the movement. Learning progresses steadily, and by trial 40 for the full-tendon case and trial 25 for the synergy controller, there is little improvement in the rotation (about 35°); further improvement was due to increased speed and earlier onset.

**Task 3** uses only a terminal cost, and learning is less steady, requiring longer to converge and producing a final policy with less emphasis on speed. Minimizing a terminal-only cost is a more difficult learning problem because it is hard to properly attribute the result of a rollout to each of the many changes in parameters. Instantaneous costs provide extra information that makes learning easier; for instance, parts of the policy that immediately result in knob displacements are rewarded. The learned policies for both full-tendon and synergy controllers are more variable than in the other experiments,
Fig. 5: Task 4 before (left column) and after (right column) learning in the full tendon space. Tendon trajectories (top row) indicate the excursions of the tendons from resting position. Excursion in the negative direction denotes pulling-in of the tendon. Change in tendon length is dominated by the two flexors (FDP and FDS, denoted cyan and magenta) and the main extensor (EI, denoted blue). The motor torques (middle row), however, demonstrate that tendons which are not moving as much are contributing significantly to the forces used in the task. Torques are limited from below at 250 mA in the spooling-out (negative) direction and 1800 mA in the pulling-in direction. The rotation of the knob is presented on the bottom row.

meaning that playback of the same policy produces a more stochastic outcome in the task.

**Task 4:** An example of tendon trajectories (top row), torques (middle row), and knob movement (bottom row) for the full-tendon controller are presented in Figure 5. Before learning (left column), only 1 degree of knob rotation is performed, but after learning (right column) the knob is rotated to 35° using a much more aggressive torque behavior and resulting in greater tendon displacements. Figure 6 is the equivalent for the synergy controller, depicting synergy positions (top row), torques (middle row), and knob rotation (bottom row).

Cost improvements from initial to learned policy as learning progresses are shown in Figure 7 for the full-tendon experiment and Figure 8 for synergies. Cost was assessed every third trial by performing three rollouts using the current best control vector. Because the tasks have different dimensionality and durations, the raw values of the costs are not meaningful for comparison. Even learning the same task from a longer demonstration would appear to have higher cost, because the cost-to-go is integrated over time. Instead we normalize the costs of the learned policy to the initial cost to show learning.

6 Discussion
We show that it is feasible to learn challenging manipulation tasks containing contact, force,
Fig. 6: Task 4 before (left column) and after (right column) learning in the synergy space. Synergy positions (top row) are the virtual synergy positions as in Section 3.2. The synergy torques (middle row) illustrate the learned pre-contact adduction of the index finger which allows greater knob rotation. The rotation of the knob is presented on the bottom row.

and motion in either the full-tendon or a reduced synergy space, without a model or sensing of tendon constraints and contact dynamics. This experiment establishes that model-free reinforcement learning is a way forward for learning a variety of tendon-driven manipulation tasks, and may seamlessly be integrated with biologically relevant dimensionality reduction. Further experiments could analyze the space of meta-parameters for consideration of learning speed or asymptotic performance, but this research demonstrates the utility of the general class of algorithms.

Rather than confirming or rejecting the synergy hypothesis for brains directly, we are interested in gaining insight into the question by demonstration of the relevant algorithms in actual hardware. Synergies are postulated from an analysis perspective, explaining but not generating behavior, often without suggesting what mechanisms learn to control the synergies. The full-tendon experiments reported here show that there exist methods for learning control in surprisingly high dimensions (14 tendons each using 50-dimensional control vectors). Discussion of motor synergies would benefit from considering the algorithms operating on the state representation, rather than axiomatically assuming dimensionality reduction to be necessary.

Nonetheless, there appear to be advantages to using the synergistic controller. The normalized reduction in costs was greater for the synergistic controller for three of the four experiments, with learning on the fourth potentially not yet converged to its asymptotic value. Further experimentation is required to ascertain whether this is exemplary of the relative ease of learning control for the two systems, but it is possible that the synergies help to shape control sampling according to the regularities of the hardware.
Fig. 7: Reduction of costs as learning progresses. Solid line indicates mean cost for three unexploratory rollouts every third trial; the shaded area is standard deviation. Costs are normalized to the before-learning cost.

More definitively, the synergistic controller learned a novel (not seen in the demonstration) strategy for improving the performance of the knob task. The learned strategy for both Tasks 2 and 4 include an initial movement which is distinct from the post-contact knob-rotating movement. This feature corresponds to a small adduction of the index finger so that contact is made further away from the kinematic goal, allowing for a greater range of motion than provided in the demonstration. Total achieved knob rotation, therefore, is slightly greater than in the full-tendon experiments. The ability to exploit unforeseen properties of the environment is part of what makes direct reinforcement learning on an embodied robot so appealing. While this strategy is available to the full-tendon controller, it was not found, indicating that something about synergies made it easier to discover this trick.

Another example of how the learned strategies can provide insight into the hardware is found in the full-tendon solution for Task 4 (see Figure 5). At 2.3 seconds, the Radial Interosseous tendon (purple, negative displacement, upper-right pane) abruptly takes in slack. This tendon acts as an extensor and abductor of the Metacarpal Phalangeal joint (“knuckle”) near the start position for the task, which would contribute to the finger not firmly pressing the knob. As the finger abducts further, however, the tendon becomes purely an abductor, and contributes to the motion without lifting the finger from the knob. Observing the learned control strategy for this tendon serves as a novel way of discovering the complex role it plays, without having to decide beforehand what the relevant phenomena are.

This exemplifies how biomimetic robots can not only take inspiration from nature to achieve engineering goals, but can also provide insights back into the systems which they mimic.

REFERENCES


