

University of Groningen

## Non-linear adaptive control inspired by neuromuscular systems

Schomaker, Lambert; Timmermans, Jordi; Banerjee, Tamalika

*Published in:*  
Bioinspiration & biomimetics

*DOI:*  
[10.1088/1748-3190/acd896](https://doi.org/10.1088/1748-3190/acd896)

**IMPORTANT NOTE:** You are advised to consult the publisher's version (publisher's PDF) if you wish to cite from it. Please check the document version below.

*Document Version*  
Publisher's PDF, also known as Version of record

*Publication date:*  
2023

[Link to publication in University of Groningen/UMCG research database](#)

*Citation for published version (APA):*

Schomaker, L., Timmermans, J., & Banerjee, T. (2023). Non-linear adaptive control inspired by neuromuscular systems. *Bioinspiration & biomimetics*, 18(4), Article 046015. <https://doi.org/10.1088/1748-3190/acd896>

**Copyright**

Other than for strictly personal use, it is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license (like Creative Commons).

The publication may also be distributed here under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license. More information can be found on the University of Groningen website: <https://www.rug.nl/library/open-access/self-archiving-pure/taverne-amendment>.

**Take-down policy**

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

*Downloaded from the University of Groningen/UMCG research database (Pure): <http://www.rug.nl/research/portal>. For technical reasons the number of authors shown on this cover page is limited to 10 maximum.*

PAPER • OPEN ACCESS

# Non-linear adaptive control inspired by neuromuscular systems

To cite this article: L Schomaker *et al* 2023 *Bioinspir. Biomim.* **18** 046015

View the [article online](#) for updates and enhancements.

## You may also like

- [Values in the backyard: the relationship between people's values and their evaluations of a real, nearby energy project](#)  
Goda Perlaviciute, Robert Görsch, Marieke Timmerman et al.
- [IPMC mechanoelectrical transduction: its scalability and optimization](#)  
D Pugal, P Solin, A Aabloo et al.
- [Blind backscattering experimental data collected in the field and an approximately globally convergent inverse algorithm](#)  
Andrey V Kuzhuget, (UNCC/ChalmersGU team), Larisa Beilina et al.

# Bioinspiration & Biomimetics



## PAPER

# Non-linear adaptive control inspired by neuromuscular systems

## OPEN ACCESS

### RECEIVED

24 February 2023

### REVISED

6 May 2023

### ACCEPTED FOR PUBLICATION

24 May 2023

### PUBLISHED

8 June 2023

Original Content from this work may be used under the terms of the [Creative Commons Attribution 4.0 licence](https://creativecommons.org/licenses/by/4.0/).

Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.



L Schomaker<sup>1,\*</sup> , J Timmermans<sup>1</sup> and T Banerjee<sup>2</sup>

<sup>1</sup> Department of Artificial Intelligence, Groningen Cognitive Systems and Materials Center, University of Groningen, Nijenborgh 9, 9747 AG Groningen, The Netherlands

<sup>2</sup> Zernike Institute of Advanced Materials, Groningen Cognitive Systems and Materials Center, University of Groningen, Nijenborgh 4, 9747 AG Groningen, The Netherlands

\* Author to whom any correspondence should be addressed.

E-mail: [l.r.b.schomaker@rug.nl](mailto:l.r.b.schomaker@rug.nl)

**Keywords:** neuromorphic control systems, bioinspired control, biomimetic systems, artificial motor units, memristors, electronics, machine learning

## Abstract

Current paradigms for neuromorphic computing focus on internal computing mechanisms, for instance using spiking-neuron models. In this study, we propose to exploit what is known about neuro-mechanical control, exploiting the mechanisms of neural ensembles and recruitment, combined with the use of second-order overdamped impulse responses corresponding to the mechanical twitches of muscle-fiber groups. Such systems may be used for controlling any analog process, by realizing three aspects: Timing, output quantity representation and wave-shape approximation. We present an electronic based model implementing a single motor unit for twitch generation. Such units can be used to construct random ensembles, separately for an agonist and antagonist ‘muscle’. Adaptivity is realized by assuming a multi-state memristive system for determining time constants in the circuit. Using SPICE-based simulations, several control tasks were implemented which involved timing, amplitude and wave shape: The inverted pendulum task, the ‘whack-a-mole’ task and a handwriting simulation. The proposed model can be used for both electric-to-electronic as well as electric-to-mechanical tasks. In particular, the ensemble-based approach and local adaptivity may be of use in future multi-fiber polymer or multi-actuator pneumatic artificial muscles, allowing for robust control under varying conditions and fatigue, as is the case in biological muscles.

## 1. Introduction

The current successes in deep learning neural networks (Sejnowski 2018) also pose new challenges. Whereas the carbon footprint of a training procedure in artificial neural networks is very high, the required computations are far from efficient on a von Neumann computer architecture. Large vectors containing input and output patterns need to be copied between storage and processor modules. However, the essential operation is a mere matrix-vector multiplication, once the data sits present. The computation of a weighted sum and a resulting final non-linear operation, however, can be realized by patterns streaming through a neuromorphic material, at a fraction of the energy dissipation. The biological

brain is an example of such a complex material. However, there are more examples of impressive, analog processing than the neural computation within the neocortex of the brain itself. A large component of neural activity is concerned with the non-linear, analog control of movement. Biological evolution in vertebrates has lead to a versatile system for neuromechanical control (i.e. the muscular system) consisting of contractile fibers and the associated spinal and cerebellar control mechanisms. In this study, we will propose an electronic system for the control of complex analog systems that is inspired by a number of essential properties of its biological counterpart. The basic tenet is not so much in realizing a final mechanical pathway but in proposing possible mechanisms for control, which potentially can be reused at several

electrical as well as soft-mechanical levels in a neuromorphic control system performing an analog control task.

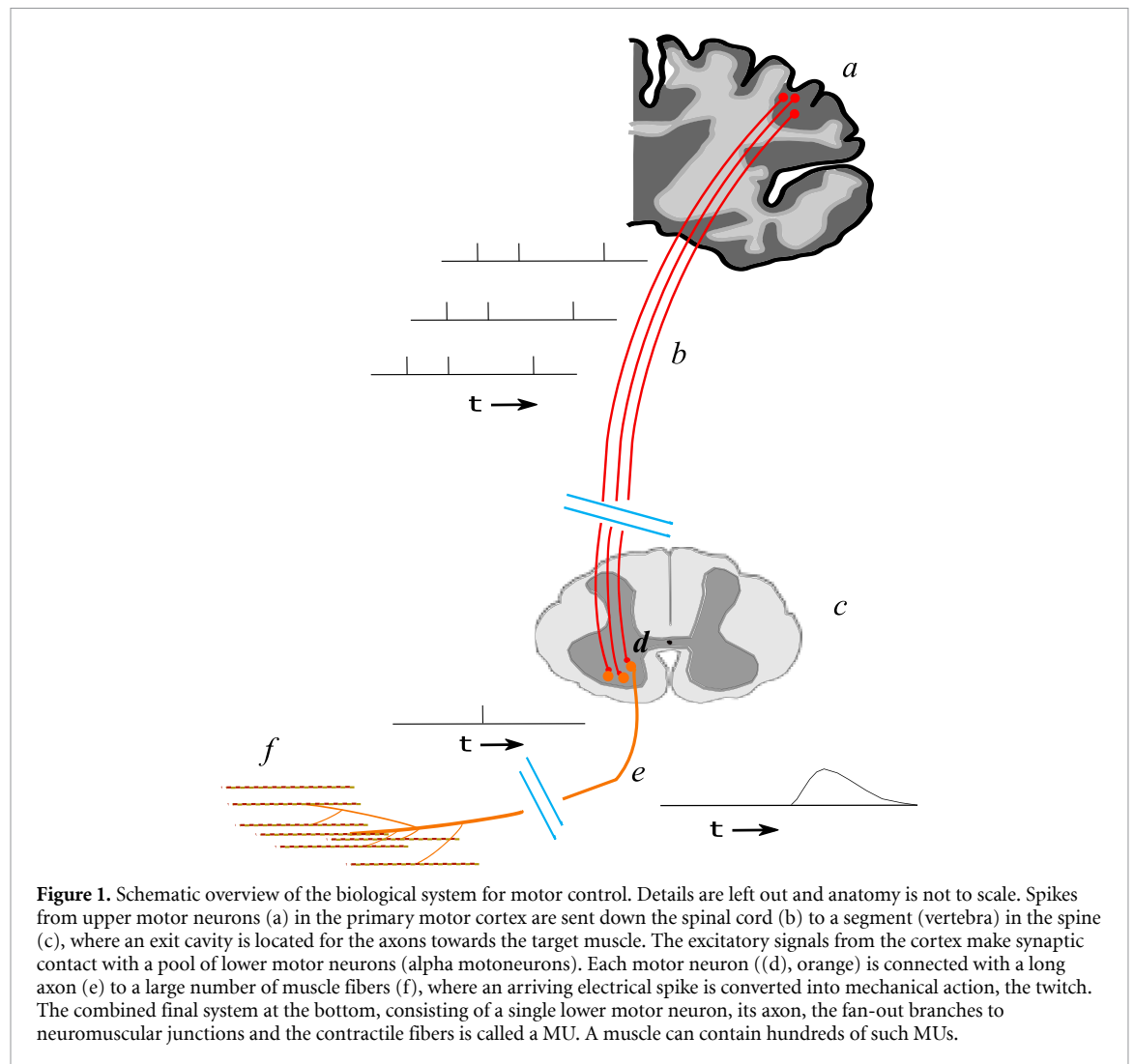
### 1.1. Related work in biologically inspired movement control in robotics

In recent literature, there is an increased interest in biologically inspired methods for movement control (DeWolf 2021). Apart from the minimization of energy requirements, biological systems provide highly effective examples of adaptive and fault-tolerant control under heterogeneous external conditions and in the face of internal resource depletion. The biological inspiration has an influence on several levels in the general movement-control architecture. In DeWolf *et al* (2016), a spiking neuron model of the motor cortices and cerebellum of the motor control system for a three-link simulated planar arm is presented. The recurrent error-driven adaptive control hierarchy model addresses the overall control architecture and the spike-based learning process. However, the model does not explicitly incorporate the essential electro-biomechanical nature of the end effector, the muscular system. Spike-based control also has attractive properties for the adaptive control of, e.g. a robotic arm mounted on a wheel chair (Ehrlich *et al* 2022) in robotic support of activities of daily living. In this study, Intel's Loihi chip (Davies *et al* 2018) was used. Although it is spike based and interesting because of its adaptive properties, the general approach heavily relies on direct control of electric motors per joint. We believe it is important to explore the possibilities for ballistic, twitch-based control as is the case in both biological muscles and would be required for new contractile materials (Mirfakhrai *et al* 2007, Leng *et al* 2021) which allow for the exploitation of intrinsic compliance. The intrinsic stiffness of traditional joint-angle control by electric motors has an advantage in industrial applications. However, the downside is that compliant control (Liang *et al* 2014, Li *et al* 2017) for service cobots must be computed in detail, centrally. New actuator technologies will require innovative control schemes. In Abadía *et al* (2021), the important aspect of movement control and planning in systems with intrinsic delays is addressed. Such time delays play an important role in sensorimotor anticipation and in learning. The paradigm concerns force control of a teleoperated Baxter robot arm, where safety requires an adequate handling of delays. This research (Abadía *et al* 2021) mainly addresses central control mechanisms. We will also address the issue of time delay in the current study, which is aimed more at peripheral processes. These (DeWolf *et al* 2016, Abadía *et al* 2021, Ehrlich *et al* 2022) and other studies (Ehrlich and Tsur 2021) would be suitable in delivering a central control architecture that acts as a front end for the peripheral biomimetic control proposed in the current study. It should be noted that

peripheral end-effector control is not the only possible application domain for the work proposed here. An additional goal of this study is to provide an interface from short-lasting spike events to internal state responses with a prolonged temporal effect, as occurs in the biological system, when a discrete spike induces a smooth gradient in the state of the receiving system. We expect that there are many application possibilities for adaptive, neuromorphic approximation of time functions. In the next sections we will explore properties of the biological motor control system.

### 1.2. Ensemble models in biological motor control

Motor control in vertebrates is characterized by an architecture where spiking control signals from a 'higher' processing level in the central nervous system, i.e. the primary motor cortex, are sent downward (Betts *et al* 2013) (figure 1). Passing through the spinal cord, the terminal connections end up within a vertebra of the spine, in the ventral horn, where a pool of alpha motoneurons receives the excitation signals from the upper control level. At this point, the motoneuron pool is then activated, leading to (a) recruitment (Liddell and Sherrington 1925) of silent neurons when their activation threshold is exceeded and (b) leading to a modulation of the spiking frequency of neurons that are already active (Kanosue *et al* 1979, Van Boxtel and Schomaker 1983). These two mechanisms allow for coarse, recruitment-based force control as well as fine regulation of required intermediate-force levels by means of altering the firing rate. The recruitment process is organized according to the 'size principle' (Henneman 1957), with smaller motor neurons connected to small fibers have a low threshold and are resistant to fatigue, whereas large motor neurons have a high threshold, activating high-force, fast-twitch (FT), but fatigable muscle fibers. The force jump that would be caused by a discrete recruitment step of a strong motor unit (MU) can be smoothed, i.e. filled in, by spike, i.e. twitch trains of increasing frequency, from other MUs, yielding smooth control that is realized by means of a discrete-event and discrete-force system. In order to obtain stable and predictable results, the system is also equipped with a negative feedback system, measuring the actual mechanical effect of a motor command in terms of muscle-fiber length variations (Hulliger 1984) as well as the force exerted on muscle tendons (Houk and Henneman 1967). In this way, the system is able to generate the required force, acceleration, speed and positional control patterns, even in the face of external disturbances and resource depletion in the muscle fibers (Sahlin 1986, Prochazka 1996, Schlink *et al* 2021). In short: The biological neural system has solved the problem of continuous, analog control where the basic actions are discrete short-lasting spike events; It has solved the problem of activating groups of contractile elements while avoiding step-wise jumps in the delivered output force; and

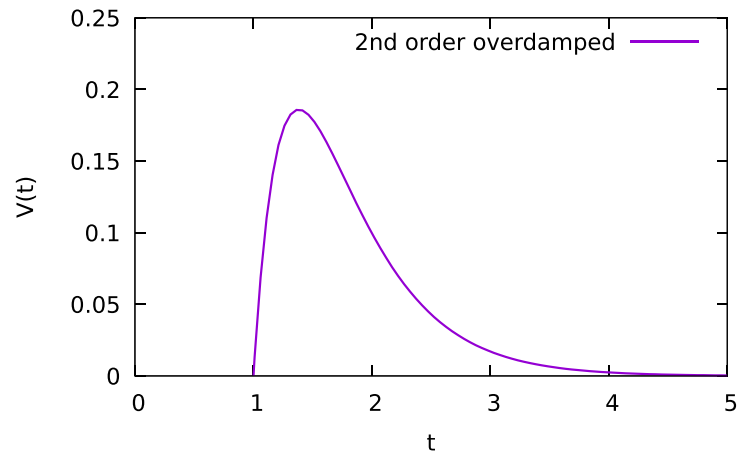


it has solved the problem that output elements may be subject to perturbations and drift during task execution. Last but not least, the biological motor system is adaptive as regards new motor tasks, the effects of resource depletion during task execution, and, at a slower pace, skeletal growth and ageing of the biological systems involved. While not all of these properties are necessary or easy to implement in new neuromorphic systems, we will explore a number of facets that lend themselves to a hybrid emulation in analog electronics and digital simulation.

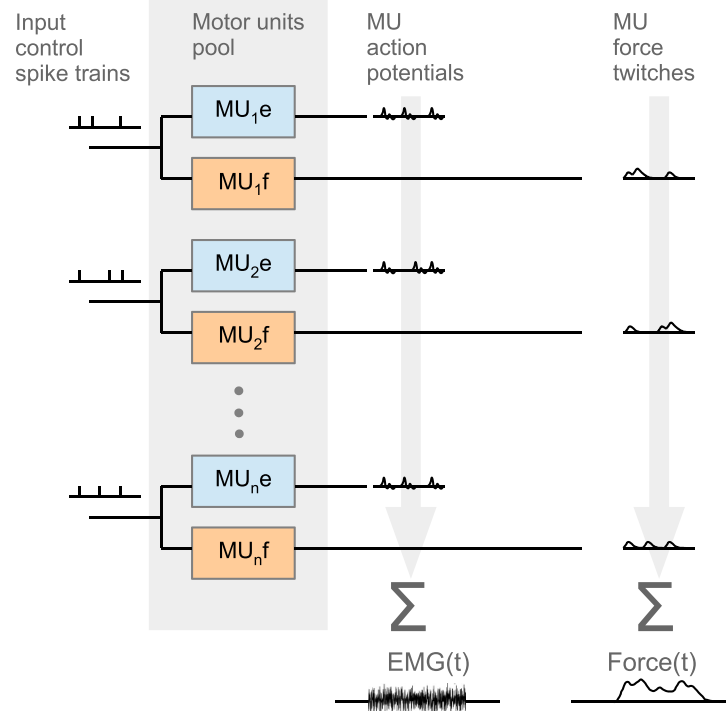
An early *ensemble approach* to modeling muscular control in the electrical domain was given by De Luca (1979)<sup>3</sup>. The muscle consists of many separate groups of fibers, MUs, in humans ranging from a few dozen MU in small facial muscles up to 1900 MU in the calf muscle (m. gastrocnemius) (Feinstein *et al* 1955). A single MU actually consists of the combination of: (1) one motoneuron in the spine; (2) its axon towards the muscle; (3) the branches of the terminal arborization within the muscle; 4) the end plates as

the neuromechanical junctions positioned on 5) the spatially distributed contractile muscle fibers. A single motoneuron excites from 100 to 1000 muscle fibers, depending on the muscle (Buchthal and Schmalbruch 1980). A discharge of the motoneuron in the ventral horn (figure 1) leads to a spike traveling towards the end plates, there leading to a spatiotemporally dispersed depolarisation which can be electrically measured on the surface of the muscle as a MU action potential (MUAP). Mechanically, the spike arrival can be measured as a muscle shortening, in the free-hanging condition, or as a force pulse in the mechanically constricted (isometric) condition. The shape of the MU *twitch* corresponds to a second-order overdamped impulse response (Crochetiere 1967, Milner-Brown *et al* 1973, Tax and van der Gon 1991) (figure 2). Whereas the summation, over motor units, of the MUAPs yields a signal measured as the electromyogram (EMG), the summation of the twitches (Burke *et al* 1976, Simony *et al* 2010, Gerritsen 2020, Smith *et al* 2020), similarly, leads to the compound force output of the muscle as shown in figure 3.

<sup>3</sup> ibi, figure 5, p 321.



**Figure 2.** Example of a second-order overdamped impulse response to an input spike (not shown) which occurs at  $t = 1$ . This function approximates the summed mechanical force effect of the contracting muscle fibers belonging to a MU, on the arrival of a spike from the alpha motoneuron.



**Figure 3.** The muscle, modeled as a motor-unit pool receiving spikes from an upper controlling level, where each MU can be viewed as a filter operator in the electrical and force domain. The total muscle activity is spatiotemporally summed, leading to an observable EMG and an overall force-time function. The interesting aspect for neuromorphic computing is that the input can be seen as gross multi-channel spike-train control on a pool of units, with an overall functional effect in the force domain characterized by a carefully timed and shaped wave pattern. Short inter-spike intervals, whether within a MU spike train, or as the consequence of spatio-temporal summation over input spike trains, lead to a temporal fusion of twitches, allowing a wide variety of output force patterns.

### 1.3. How to exploit mechanisms in biological control within neuromorphic computing?

The aim of this study is not to model all forms of continuous, sustained control, but to focus on motor tasks involving short-lasting ballistic movements or analog bursting. By using the second-order overdamped impulse response (figure 2) to a spike, i.e. a

pulse with a very short duration, the effect of that input event is spread over time, with a smeared-out duration which is useful in relation to the analog-control requirements of a particular motor task. In brief, the idea is to use the 2nd order impulse response as a base function for a neuromorphic universal function approximator. In an electronic implementation,

this can be realized by putting two ‘RC’ filters in series, realizing an impulse response:

$$h(t) = \frac{k}{\tau_2 - \tau_1} (\exp(-t/\tau_2) - \exp(-t/\tau_1)) \quad (1)$$

where  $\tau_1$  and  $\tau_2$  are time constants.

Advantages of this approach:

- Bridge the gap between short-lasting spikes on the input and long-lasting analog wave output as required by a movement task;
- Ability of the twitch impulse response to be used as a base function for general function approximation;
- Requiring only simple electronic circuits;
- In which the behavior is controlled by adaptive memristors.

#### 1.4. Relation with traditional machine-learning methods for function approximation

Universal function approximators can be realized in many ways, ranging from polynomials, Fourier transform, additive mixtures of Gaussians (GMM) (Gauvain and Lee 1994), multilayer perceptrons (Hornik *et al* 1989), or other methods. There are, however, peculiarities caused by the used basis function in such approaches. For instance, Gaussian mixtures are only an effective function approximator for probability densities, i.e.  $p \in [0, 1]$ . Similarly, twitches (i.e. 2nd order overdamped impulse responses) will be unipolar assuming a positive input impulse, as required by the electronics. Therefore, modeling using only positive ‘humps’ is not sufficient, because in control problems, the controller output needs to be signed (i.e. requiring a bipolar control action  $a \in [-s, +s]$ ). Twitches, i.e. 2nd order ‘humps’ (figure 2) are a very skewed approximation of a Gaussian, but cheap to generate because the only requirement is a cascade of two RC filters which can be realized by a wide range of neuromorphic materials having capacitive, or more generally, state-persistent properties of the leaky-integrator type.

The required bipolarity for control can be realized by assuming both an agonist and an antagonist MU pool, with a reversed sign on the base function. In muscular systems, the sign reversal is the natural consequence of a geometric structure that is organized around a joint, the agonist and antagonist muscles being connected to opposing locations around a hinge (joint) in a kinematic chain (Hartenberg and Denavit 1964).

We will propose a computing architecture for analog control tasks requiring short-lasting (ballistic) wave shapes with appropriate timing, amplitude, and adaptive wave shapes. The modeling will be done in two levels: (1) an electronic circuit is proposed using NGSPICE (Cox *et al* 1992, Strasnick *et al* 2021) to model a single MU, allowing for the timed generation of 2nd-order overdamped impulse responses; (2) an

overall architecture is proposed, solving bipolar (positive and negative) control signals by summing outputs over ensembles of MUs; (3) an overall simulation is realized in Python to instantiate the NGSPICE simulations and collect their output; (4) heuristic, STDP-inspired learning rules are presented to adapt memristor resistances to a task; (5) several tasks will be used to evaluate the capabilities of this biologically inspired analog control system.

The proposed approach has three desired properties in mind:

- The total system of circuits needs to be flexible in the sense that it needs to be able to learn to perform multiple analog control tasks involving timings and wave shapes;
- The circuit needs to operate using current memristive devices having a limited set of resistance values, usually only two;
- The circuit has low power requirements;

To show that ensembles of MU circuits can learn to perform dynamic control tasks, we will test whether it can be trained to control:

- (i) A cart-pole (inverted pendulum) task (Donaldson 1960, Blitzer 1965, Barto *et al* 1983)  
Task focus: (*timing and amplitude in 1D*);
- (ii) Playing the whack-a-mole game (Gutiérrez *et al* 2006, Wikipedia 2022)  
Task focus: (*timing and amplitude in 1D*);
- (iii) We also want to show the ability of the circuit to generate wave shapes representing pen-tip trajectories for handwritten characters (Schomaker 1992)  
Task focus: (*trajectory formation in 2D*).

Finally, we will discuss the modeling results and the possible implications for future neuromorphic and biomimetic control systems.

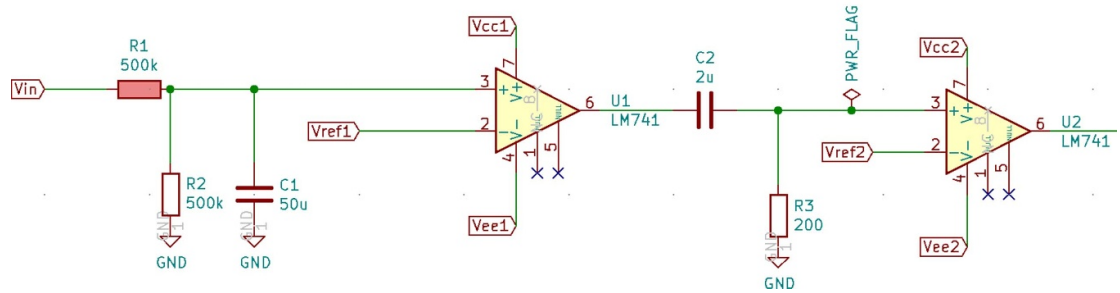
## 2. Methods

### 2.1. Circuit

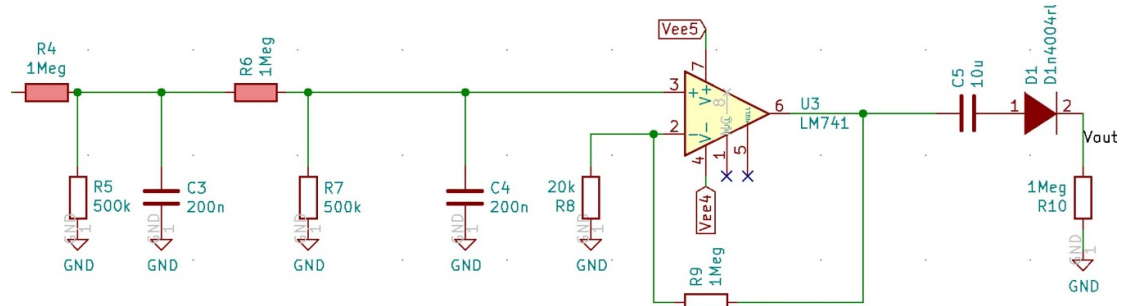
The basic component of the approach is an electronic circuit representing a single MU, the MU circuit, for generating a delayed twitch with a particular amplitude. An ensemble of several such circuits is assumed to represent a ‘muscle’, i.e. a complete analog controller for one output dimension. Two such ensembles can operate in an agonist/antagonist mode, allowing for bipolar control.

#### 2.1.1. Delayed-pulse generator

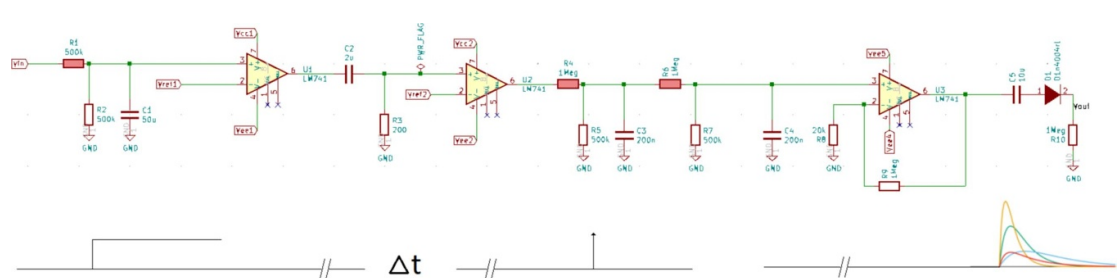
The first part of the circuit is the time-delayed spike generator. This part of the circuit will turn a basic activation stimulus (a step pulse, from rest) into a delayed pulse, represented by a narrow, high amplitude block pulse. This brief-lasting spike is assumed



**Figure 4.** First part of the circuit which generates delayed narrow block pulse (‘Dirac’ pulse in the muscle fiber model).



**Figure 5.** Second part of the circuit, generating a muscle twitch-like signal from a narrow block pulse input.



**Figure 6.** Overall diagram of twitch-generating circuit representing a MU. The first part of the circuit generates a delayed pulse while the second one generates a second-order overdamped impulse response. The red resistances are variable, i.e. memristive devices.

to be a physically realizable version of the idealized (Dirac 1947) pulse, with an area of one and a duration approaching zero seconds, as is used in signals & systems theory. It represents a single input spike of the input trains as shown in figure 3. The delayed pulse generator can be seen in figure 4. A memristor-capacitor combination realizes a charging curve leading to a step response by the first opamp, which is translated into a spike by the second opamp, not unlike a leaky-integrate and fire (LIF) neuron model (Abbott 1999).

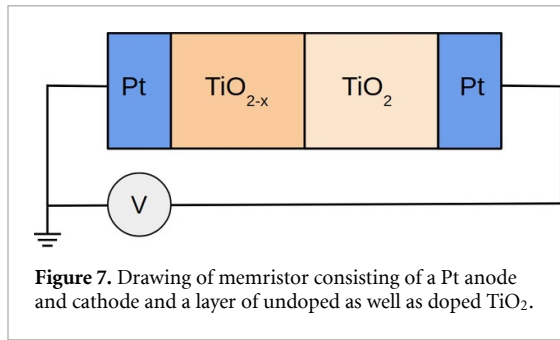
### 2.1.2. Muscle-twitch generator

The second subcircuit is the twitch generator, which converts the short-lasting spikes from the first part of the circuit into longer-lasting spikes that look like the

tension over time of a muscle twitch. The conversion from a spike to a twitch is realized by filtering the narrow block pulse with a second-order RC filter. The double RC filter generates a second-order overdamped impulse response. The subcircuit can be seen in figure 5.

### 2.1.3. Complete single-MU circuit

The complete ‘single-MU’ twitch generator circuit is shown in figure 6. It has only three memristors. In total, the single circuit has 16 passive components and three standard opamps (LM741). The components have been selected for a proof of concept, where notably some capacitors (C1, C5) are dimensioned to allow for modeling a relatively slow real-time process. When embedded in an ensemble, additional



memristors are needed to obtain a weighted-sum output over MU circuits to allow a variation of output patterns.

#### 2.1.4. Memristive devices

The memristor, coined by Chua (1971) and produced in the HP Labs (Strukov *et al* 2008) is typically a non-linear device, of which the microscopic internal state can be modified and measured as a change in the resistance. Their ability to co-locate memory and compute in the same element makes them akin to biological synapses.

The memristor is a two-terminal device whose resistance can be switched between several states depending on the applied voltage. The classical memristor consists of two electrodes that sandwiches a binary resistive oxide, such as  $\text{TiO}_2$  as seen in figure 7.

Resistive switching is activated when the electric field due to the applied voltage leads to the movement of oxygen vacancies whose density at either interface with Pt can be controlled by the applied voltage. The higher concentration of oxygen vacancies at  $\text{TiO}_{2-x}$  ( $x$  denoting the vacancy concentration) on the left side leads to the formation of an ohmic contact with Pt and a Schottky interface with Pt on the right side. The barrier height and width modulation related to the movement of the oxygen vacancies, with the applied voltage is associated with the different resistance states that such memristors exhibit.

A non-volatile memristor will retain its resistive state when no voltage is applied. This way, the memristor can be regarded as a component with memory. Non-volatile memristors are usually bistable, although research has been done on memristive systems with a higher number of states (Ielmini and Waser 2016, Sun *et al* 2021). Bistable memristors have an ON/Low resistive and OFF/High resistive state, also referred to as LRS and HRS respectively.

In neuromorphic systems, the goal is to realize local adaptation at individual synapses. In this study, we first want to investigate whether memristors with a limited number of resistive states can be used for analog tasks. It is not clear, *a priori*, that a coarse quantization of resistances (read: ‘neural-network weights’) still allows for sufficiently fine control in such tasks. Therefore, updates to the memristor parameters are,

for the time being, realized by using an external controller which allows for a stochastic exploration of resistance configurations. The external controller can control the weight changes of the entire system or subsystems within the system. The controller receives input information from devices that measure the physical properties of the system, which are different across different tasks. These properties are converted into a control signal, based on heuristics, that contain a direction of update (i.e. an increase or decrease in resistance) and what subset of resistors to update. The resistances then change with only a set value. The controller’s stochasticity determines which resistors are chosen to be updated, namely; at every instance where updates are performed, depending on the task, the resistors have a 20% or 50% chance of being adjusted. A flowchart of the process can be seen in figure A1.

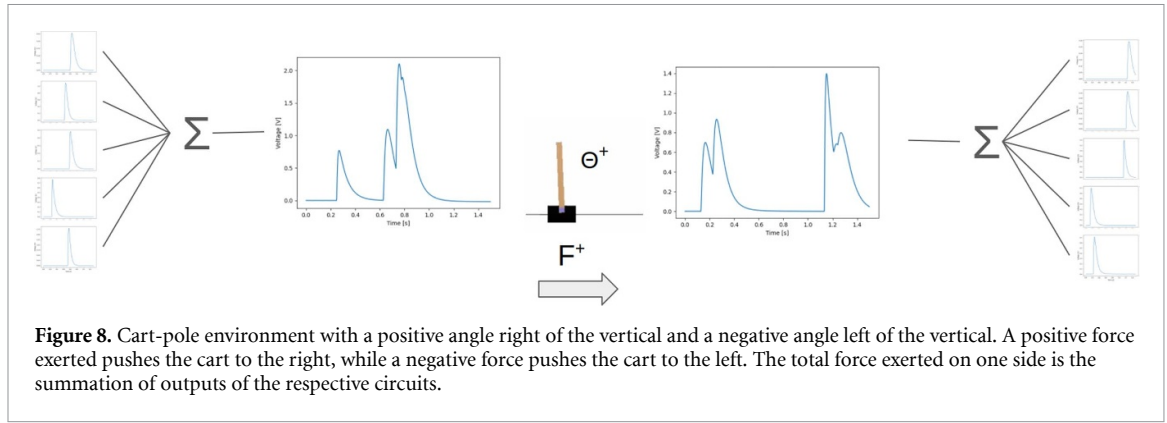
#### 2.2. Ensemble architecture

Similar to the muscle-modeling approach, a summation over individual twitches from MU circuits is used to obtain the driving signal. The architecture used for this contains multiple circuits with a common activating step input, where only the variable resistance values will be different between circuits. The circuits are split up into antagonistic groups, meaning one group of circuits will contribute positively while the other contributes negatively. By means of a long-lasting block pulse stimulus, the circuits in an ensemble are stimulated and will produce differently shaped and delayed second-order overdamped voltage pulses. The individual pulses are summed per group and represent the force an actuator has to exert, i.e. in the simulation the exerted force is assumed to be equal to the voltage (i.e. value  $[V] = \text{value } [N]$ ).

Three different ensemble architectures are used; single-bin homogeneous updating (SBHoU) ensemble, multi-bin homogeneous updating (MBHoU) ensemble and the MU recruiting-inspired MBHeU ensemble. The SBHoU ensemble consists of one group of an antagonistic pair of circuits where either all agonist or all antagonist circuits will activate. The MBHoU ensemble consists of eight groups of circuits (four agonists and four antagonists) of which each activates depending on circumstances. The MBHeU ensemble consists of eight groups of circuits in which one or more activate depending on the circumstances. A schematic visualization can be found in figure B1 together with a more detailed explanation in appendix B.

#### 2.3. Control tasks

In this section, our main task: the cart-pole or inverted-pendulum task is described. The whack-a-mole task is described in appendix C and a trajectory-generation task for pen-tip movement in handwriting is described in appendix D. The control tasks address approximation problems in timing, amplitude and wave-shape domains.



The cart-pole task represents a typical inverted-pendulum control problem. It is used to evaluate the timing and amplitude control capabilities of our model. The cart-pole system consists of a pendulum attached to a cart using a hinge weight. The pendulum needs to be balanced by the movement of the cart, initiated by either turning the wheels or applying a force to the sides of the cart. In the current model, the task does not require an initial upswing from a down-hanging orientation. The cart-pole is simulated using the OpenAI Gym cart-pole environment (Brockman *et al* 2016). This environment utilizes the frictionless version of the equations of motions of the cart-pole described in Barto *et al* (1983), shown in equations (2) and (3).

$$\ddot{\Theta} = \frac{g \sin(\Theta) + \cos(\Theta) * \frac{-F - m_p * l * \dot{\Theta}}{m_c + m_p}}{l * (\frac{4}{3} - \frac{m_p * \cos^2(\Theta)}{m_c + m_p})} \quad (2)$$

$$\ddot{x} = \frac{F + m_p * l * (\dot{\Theta}^2 * \sin\Theta - \ddot{\Theta} * \cos(\Theta))}{m_c + m_p} \quad (3)$$

where  $g$  = gravitational acceleration ( $9.8 \text{ (m s}^{-2}\text{)}$ ),  $m_c$  = mass of the cart,  $m_p$  = mass of the pole,  $l$  = distance from hinge to center of mass of the pole, and  $F$  = horizontal force applied to the center of mass of the cart.

Different machine learning algorithms to control the inverted pendulum have been explored in simulation under which fuzzy logic (Becerikli and Celik 2007), reinforcement learning (Lillicrap *et al* 2019) and spiking neural networks (Kang and Banerjee 2018). The problem is also tackled in real life by reinforcement learning and a hybrid system of memristors and transistors (Wang *et al* 2019), where the memristors were used to speed up matrix multiplications and introduce stochasticity. The cart-pole is controlled in a simulated environment. We use the circuit outputs to generate force, and the OpenAI Gym cart-pole environment (Brockman *et al* 2016) to simulate the cart-pole. The environment with annotations and outputs of force-generating circuits can be seen in figure 8. The cart in this environment is moved by a pushing force that pushes to the left or the

right. We call the ensemble pushing right the agonist ensemble and the ensemble pushing left the antagonist ensemble.

When the pole flips from a vertical position to a positive angle compared to this position in a clockwise direction, the agonist circuits that have to push the cart receive a 10 V block pulse until the pendulum flips to the other side. The circuits that are not activated also do not contribute in pushing the cart anymore (e.g. when the pole flips from a positive to a negative angle, the agonist circuit ensemble does not contribute any force anymore).

The pole is assumed to fall, ending a run in a failure, whenever the pole exceeds an angle of  $20^\circ$  relative to the vertical. A run is assumed to finish successfully whenever the pendulum is balanced for 60 s. Unlike in the common benchmark setups, we do not set a horizontal limit for where the cart is able to move.

Monte Carlo search will be performed to explore the performance of different ensemble schemes in pole balancing. The physical properties are arbitrarily chosen; The mass of the pole is 0.2 kg, the mass of the cart is 0.3 kg and the pole length is 1 m. The activated bins with their corresponding angular velocity ranges can be seen in table B2. One run consists of 90 balancing attempts where the 90 initial conditions for the pole are each of the elements of the Cartesian product between the following two sets;

$$\begin{aligned} \Theta &= \{-0.25, -0.15, -0.1, -0.05, -0.01, 0.01, \\ &\quad 0.05, 0.1, 0.15, 0.25\} \text{ (rad)} \\ \dot{\Theta} &= \{-0.25, -0.15, -0.1, -0.05, 0, 0.05, 0.1, \\ &\quad 0.15, 0.25\} \text{ (rad s}^{-1}\text{)}. \end{aligned}$$

The score of each run is the average balancing time over the attempts.

By evaluating the results from the Monte Carlo search we will hypothesize what ensemble could be used best to utilize in combination with a more goal-oriented learning rule which we call event timing-based plasticity. We will compare these combinations of ensembles and learning rules by means of two different tasks. The first task is a simple pendulum balancing task where the pendulum has to be balanced from an initial angle of approximately  $0^\circ$  and

an angular velocity between  $-0.05$  and  $-1.5$  or  $0.05$  and  $1.5$  ( $\text{rad s}^{-1}$ ). The second task is balancing the pendulum with initial positions as seen in the Monte Carlo search (i.e. an angle-angular velocity combination from the set of 90 initial states).

In both tasks, when the pendulum either crosses the vertical again or falls, the resistances are updated. Taking the time between two consecutive pendulum crossings to be  $t$ , the resistances update according to the following rules;

- Resistances should increase when the pendulum is pushed too fast and too hard, resulting in lower amplitudes and bigger time delays.
- Resistances should not change when  $t$  is within the ideal region.
- Resistances should decrease when the pendulum is pushed too slow and not hard enough, resulting in higher amplitudes and smaller time delays.

An ideal time between two crossings is assumed to be between  $0.1$  s and  $1$  s. Resistances are only allowed to change  $-10\text{ k}\Omega|10\text{ k}\Omega$  or  $0\ \Omega$ . The probability of change will be determined by the external controller's stochasticity. Afterward, the task in which the pendulum has a vertical initial position is extended to determine the working range of the system. This is done by varying the physical properties of the cart-pole system.

Besides controlling the pendulum, playing whack a mole and performing handwriting will also be explored. The details of the tasks will be described in appendix B.

## 2.4. Optimization methods

Optimization methods are used to teach or optimize a system with respect to a goal. Several optimization methods are used in this research. This section explains the optimization methods used; Monte-Carlo search (random search), the Widrow-Hoff learning rule (Widrow and Hoff 1960) and a rule inspired by STDP.

### 2.4.1. Spike-time and event-time dependent plasticity

Neurons are plastic in nature and update their connection strengths based on local information in the form of relative spike timings. Relative time differences in spikes can either cause an increase in connection strength, a decrease in connection strength or can leave the connection strength unchanged. Different STDP schemes are available, of which each has its own advantages and disadvantages. The STDP algorithms avoid global optimization and optimize locally. Local updating is an attractive property because it eliminates the need for additional connections or 'third leads', into an electronic circuit. However, the current study is first focused on the general principle of neuromuscular-inspired analog control, evaluating it on a number of tasks. However, it is also useful to

borrow from STDP the event-time principle as the basis for memristor updates. Actually, there exists a wide range of proposed STDP curves beyond the classic '1/x' shape (Buchanan and Mellor 2010).

In this study, we will use a more abstract version, dubbed event-time dependent plasticity (ETDP), which is based on deriving update signals for a memristor update from relative timing requirements of a particular task. The basic and discrete update actions are: 'resistance up', 'resistance unchanged', and 'resistance down'. The assumption of ETDP is that it is possible to determine an event in the task execution, of which the (relative) timing is related to success or failure. Because our proposed MU circuit in itself does not require a large crossbar with many memristors on the crossings, the disadvantage of the circuit-global nature of ETDP is manageable, although admittedly not negligible. For the tasks where ETDP is used, an explicit time event can be detected and used as an update signal.

## 2.5. Overall approach

We can model a single MU by treating it as a mechanism that produces a delayed spike with respect to a stimulus after which it filters this spike using a second-order overdamped transfer function (figure 2). This model is inspired by the neural firing timing property of which the importance is suggested by Mainen and Sejnowski (1995). Controlling a dynamical system using only a single signal however, may not be viable for systems that require fine-grained control. In order to achieve more fine-grained control, it is preferable to sum the activation over an ensemble of units. As seen in figure 3, multiple MUs can either be added up in the electrical domain in terms of MUAPs or they can be added up in terms of their twitch forces. We will use the latter to control our system. By timing the twitch forces we perform ballistic control on the system. We do this by simulating an ensemble of MUs, which work together to control systems. Each MU in our system will be simulated by an NGSPICE model (Cox *et al* 1992, Nagel *et al* 2011), which is an open-source tool from the SPICE<sup>4</sup> family of algorithms for detailed simulation of electronic circuits. Similar to the model that adds up multiple twitches to generate more complex force patterns, we will add up multiple twitch-like circuit outputs to generate more complex voltage patterns. These voltage patterns are then digitally transformed into force patterns, used as driving forces in the simulations.

## 3. Results

### 3.1. Validation and calibration

Before being able to perform the three major experimental tasks, some validation and calibration steps

<sup>4</sup> <https://en.wikipedia.org/wiki/SPICE> (28 April 2023).

need to be performed on the MU-circuit ensemble approach.

### 3.1.1. Validating the timing capability of a single motor-unit circuit

In order to test the basic timing capability of the first stage of a single motor-unit circuit, a number of pilot evaluations were done to verify the dependency on the resistance of memristor R1. The circuit is fed with a step function, from the resting state (0 V). A monitoring controller sets a predefined delay and starts an iterative update of R1 until the timing of the output signal is correct, starting with a predefined initial resistance (100 k $\Omega$ ). The resistance value for a delay in the range of 0–1200 ms can be estimated in about 20 steps for coarse update steps of + or –10 k $\Omega$ . After convergence from 21 steps onward, the obtained MAE was 16.5 ms and the RMS error 20.8 ms. An approximation with  $|\Delta R|$  values which gradually become smaller leads to a more accurate timing but required a longer iteration sequence of 30 steps and assumes a finer memristance resolution. More details can be found in appendix E.1. It was concluded that this performance was a sufficient basis for the actual timing tasks (inverted pendulum and whack-a-mole).

### 3.1.2. Validating the shape-fitting capability of an ensemble of twitch generators

Similarly, before embarking on more complicated tasks, we needed to evaluate the capability of an ensemble of electronic twitch-generator circuits to approximate an arbitrary wave shape. Five different transient time functions were used. Three unipolar wave shapes: sawtooth pulse (increasing), reverse sawtooth (decreasing) and block pulse. Furthermore, a bipolar wave shape was evaluated, i.e. a damped sinusoidal impulse response. All signals were approximated using random weight initialization and the Widrow–Hoff learning rule. Please note that the basic architecture allows for bipolar signals in all cases because there is both an agonist and antagonist pool. For the unipolar targets, the system needs to learn to suppress negative output. Results show that with 100 circuits (50 for positively signed twitches, 50 for negatively signed twitches, a decent fit can be reached in about 1000 iterations. Figure 9 shows the results for the damped sinusoid. The approximation was also acceptable for the difficult ramp slope and the discrete jumps in the sawtooth and block pulse targets. Table 1 gives the RMS errors for different wave shapes. An overview of these pilot evaluations can be found in appendix E.2.

### 3.1.3. Validating a basic inverted-pendulum task ('1 s experiment')

The cart-pole task represents a typical inverted-pendulum control problem. It is used to evaluate the timing and amplitude control capabilities of our model. The goal was set to push the cart such that the

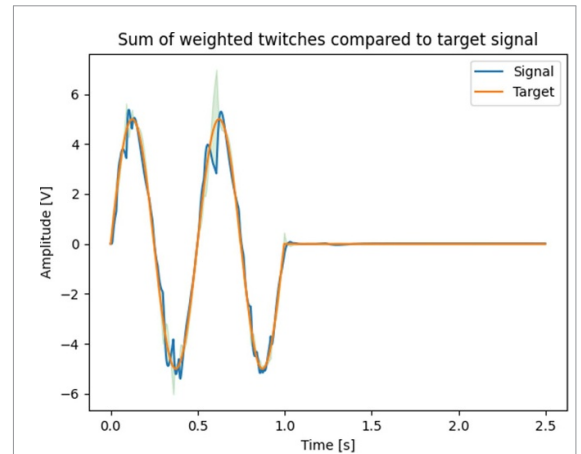


Figure 9. Damped sinusoid fitted with 50 scaled agonist and antagonist circuit outputs.

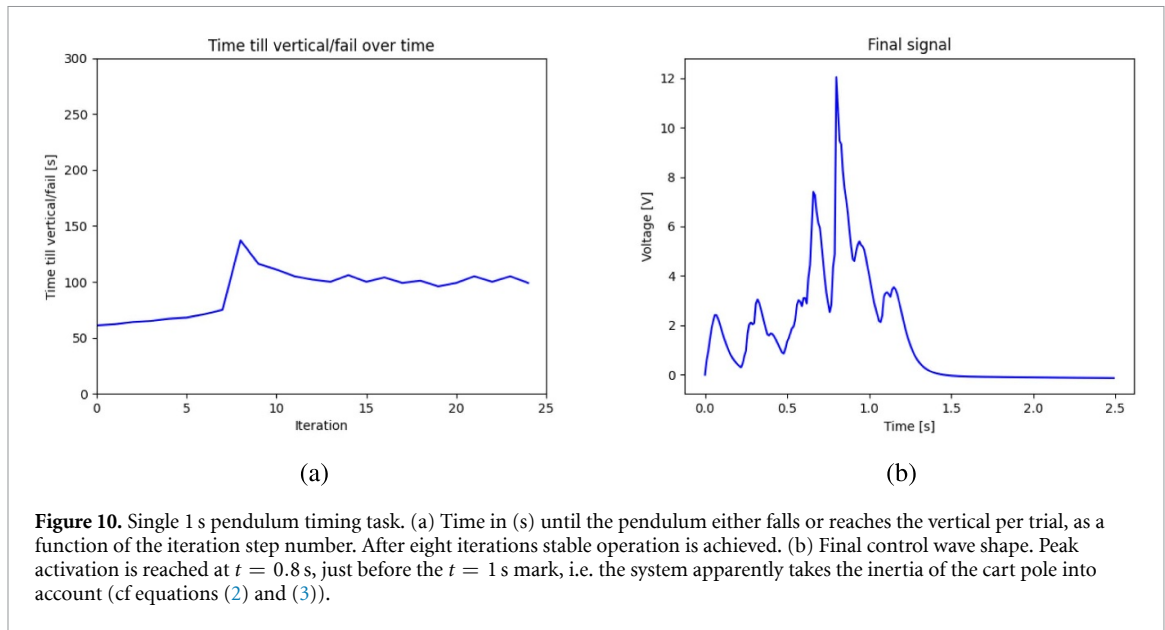
Table 1. Wave shape approximation RMS error in V, for different wave shapes.

Wave	RMS error	Vpp
Block pulse	0.31	5
Damped sine	0.51	8
Reverse sawtooth	0.74	5
Sawtooth	0.76	5
Sine	0.29	10

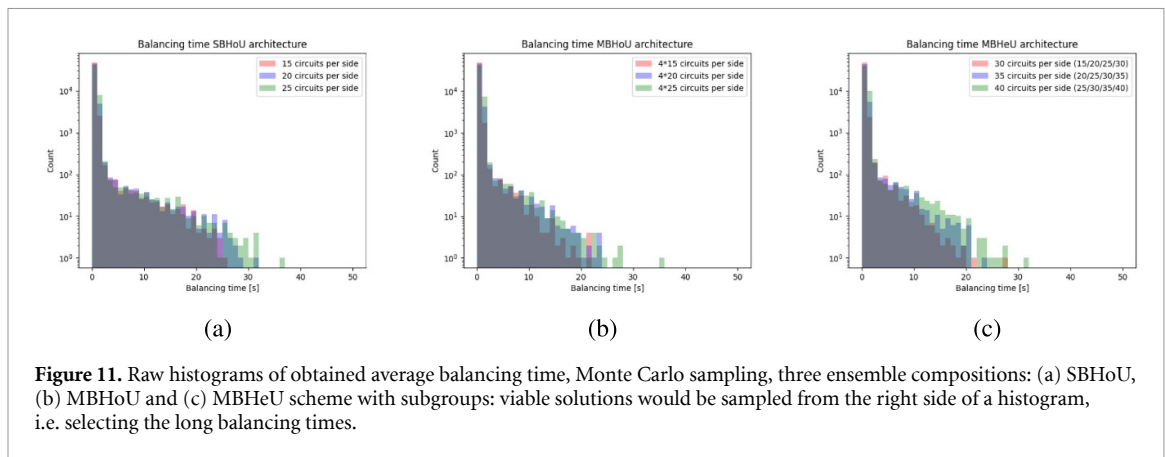
Table 2. Update rules of the 1 s experiment.

Event	Resistance change
Pendulum falls	Decrease
Pendulum crosses vertical over 1 s	Decrease
Pendulum crosses vertical under 1 s	Increase

pole crosses the vertical, in one second from an initial angle of 0.15 (rad), and initial pendulum angular velocity of 0 (rad s<sup>-1</sup>). The goal event time window for the pendulum crossing the vertical was from 1.00 s to 1.01 s. After every attempt, when the pendulum fell or reached the vertical, resistances change either  $2\text{ k}\Omega| - 2\text{ k}\Omega$  or  $0\ \Omega$  with an equal probability (i.e. 50/50). The rules can be seen in table 2. The timing and cumulative circuit output of a single run can be seen in figure 10. After iteration 8, the system has learned to let the pendulum cross the vertical. The final signal of the run can be seen in figure 10(b). Here we can see that the cart is nudged just a little, in the beginning. Just before the 1 s mark however, at  $t = 0.8\text{ s}$ , the cart is pushed hard by two peaking forces. In summary, the system initially attempts to push the pendulum to the vertical and when this is achieved it optimizes the timing given the inertia characteristic of the pendulum, yielding a stable performance after ten trials. The experiment is also performed using different numbers of circuits in order to see whether the number of signals increases or decreases convergence. The average convergence time with error bars can be seen in figure E4. Here we



**Figure 10.** Single 1 s pendulum timing task. (a) Time in (s) until the pendulum either falls or reaches the vertical per trial, as a function of the iteration step number. After eight iterations stable operation is achieved. (b) Final control wave shape. Peak activation is reached at  $t = 0.8$  s, just before the  $t = 1$  s mark, i.e. the system apparently takes the inertia of the cart pole into account (cf equations (2) and (3)).



**Figure 11.** Raw histograms of obtained average balancing time, Monte Carlo sampling, three ensemble compositions: (a) SBHoU, (b) MBHoU and (c) MBHeU scheme with subgroups: viable solutions would be sampled from the right side of a histogram, i.e. selecting the long balancing times.

can see that with an increasing number of circuits, the convergence time (number of steps) decreases as well as its variation over training sessions. We have now shown that besides timing the delayed step, we can also time and shape the pendulum-control action using an ensemble of circuits.

After embarking on the validation of basic functionality with acceptable to good results, we can now address the results for the major tasks.

### 3.2. Inverted-pendulum task: ensemble configurations

All three ensemble architectures are compared. For every ensemble type we performed 3 tests using a different number of circuits. For every run (one ensemble setting), we sampled 50 000 times from the 10 000 randomly generated circuit outputs (section 3.1.2). For every sample we let this combination of outputs balance the pendulum starting with 90 different combinations of angles and angular velocities as mentioned in section 2.3. We thus performed  $50\,000 \times 90 = 4\,500\,000$  trials to get an impression of every setting's performance. For every sample

(containing 90 trials) the score is equal to the average balancing time over all 90 trials. For each ensemble composition, the histogram of obtained balancing times over all the trials is shown in figure 11. Please note that vertical axes are on a logarithmic scale.

Since we are only interested in balancing the pendulum for a long time, comparing the three different methods is done by looking at the number of occurrences in the higher balancing times ( $>20$  s).

The first histogram shows the SBHoU ensemble and can be seen in figure 11(a). We can see that there are a vast number of trials that end up in the one or two-second balancing time bin. We can also see that especially the 25 circuits per side ensemble resulted in several trials that balanced the pendulum longer than 20 s.

For the MBHoU ensemble, seen in figure 11(b), we can see that again there are a large number of combinations that result in one to two-seconds average balancing time. The 25 circuits ensemble performs best, like in the SBHoU experiment. This time however there were fewer trials resulting in a balancing time higher than 20 s.

**Table 3.** Number of average balancing times above 20 s per ensemble size.

Ensemble size	SBHoU	MBHoU	MBHeU
I	21	4	2
II	50	8	7
III	56	15	21
Total	127	27	30

The third histogram, seen in figure 11(c), shows the MBHeU ensemble. Also here, many combinations of circuits were capable of reaching a one to two-second average balancing time. However, the setting with the largest number of circuits (40) scores best but there are only a few solutions yielding balancing times that are higher than 20 s.

The numbers of top balancing times are shown in table 3. Here we can see that in the extreme cases, the SBHoU ensemble performs best, followed by the MBHeU solution, which is in turn followed by the MBHoU ensemble type by only a small margin. We can also see that the setting with the most circuit performs best for every ensemble type.

When selecting from the range of acceptable solutions (min. 20 s of balancing), the single-bin/homogeneous ensemble (SBHoU, panel (a) in figure 11) thus seems to perform best (56 acceptable solutions). Of the multi-bin architectures, the MBHoU approach requires more circuits than the MBHeU approach, but surprisingly the latter approach performs better (21 vs 15 solutions). Therefore, the multi-bin homogeneous ensemble with many circuits and an anticipated high energy demand (MBHoU) will not be chosen, in the next experiments.

### 3.3. Pendulum balancing using ETDP

In this experiment, we will compare two ensemble types on a pendulum-balancing task. As opposed to plain Monte-Carlo search, we will apply a more goal-directed form of plasticity, ETDP. The reason for also exploring the MBHeU scheme while the SBHoU ensemble performed best is not only because of its similarity to the biological process, but also due to the expectation that our event-time approach would potentially be able to learn to balance the pendulum for a wider dynamic range of initial settings (initial angle, initial angular velocity, cart mass, pendulum mass and pendulum length) than is the case in a homogeneous ensemble.

The best scheme for the MBHeU approach is the  $25 \rightarrow 25 + 5 \rightarrow 25 + 5 + 5 \rightarrow 25 + 5 + 5 + 5$  circuits scheme (see table F1 and appendix B for an explanation). In order to compare the SBHoU approach in a fair manner, we choose the same number of circuits for the SBHoU approach as for the MBHeU approach, which is 80 (40 per side).

Ten different runs consisting of a maximum of 500 resistance change opportunities are performed

**Table 4.** Fraction of perfectly balanced states given initial pendulum condition and circuit ensemble architecture.

	SBHoU	MBHeU
Vertical start	$1 \pm 0$	$1 \pm 0$
Non-vertical start	$0.7 \pm 0.2$	$0.97 \pm 0.02$

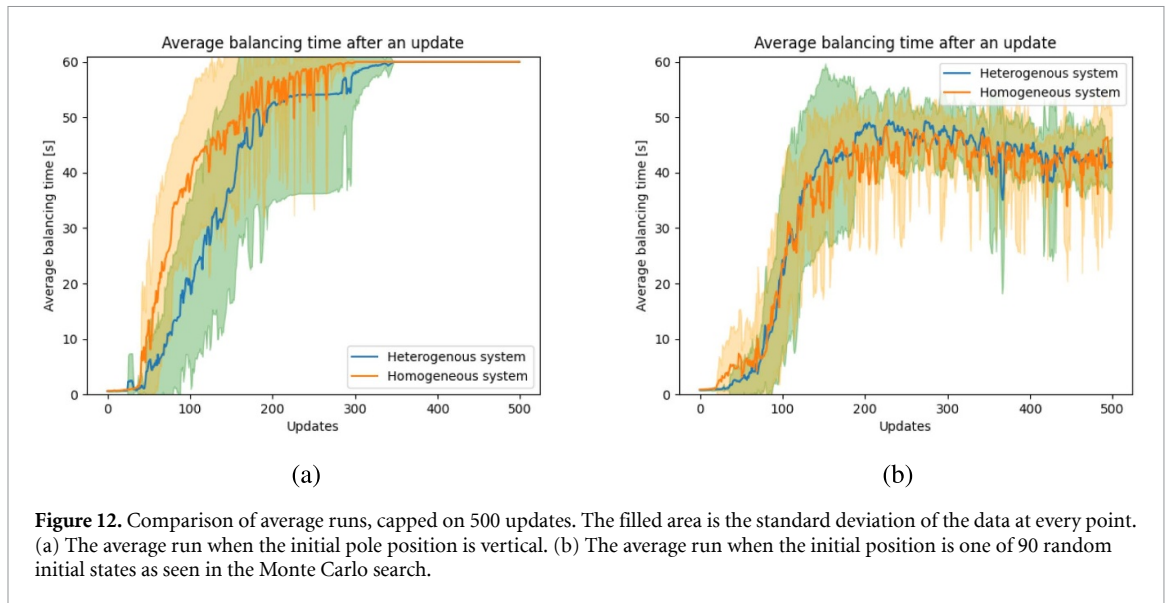
for both a random starting angle and starting angular velocity as well as a vertical start. The pendulum has a length of 1 m, a weight of 0.2 kg and the weight of the cart is 0.3 kg. Resistances are only allowed to change  $-10\text{ k}\Omega|10\text{ k}\Omega$  or  $0\text{ k}\Omega$  with a 20% chance of changing, which is determined by an external controller. After every update, the performance is measured by taking the average balancing time, calculated over all 90 states in the case of the random starting state and calculated over 60 different initial angular velocities<sup>5</sup>  $\text{rad s}^{-1}$  in the case of the vertical starting state. The best performing combination of circuits (i.e. the combination of circuits of which the average balancing time is the highest) are saved for every run. The number of perfectly balanced (60 s) states is counted for each of those circuits and is then averaged. The comparison can be seen in table 4.

We can see that in the vertical start setting, both architectures are able to perfectly balance all test settings. In the non-vertical start setting, however the single bin homogeneous update architecture is able to only balance 2/3rd of the settings while the multi bin heterogeneous update architecture is able to balance 97% of the initial conditions. The MBHeU architecture is significantly better in the non-vertical start setting than the SBHoU architecture  $t(18) = -4.266, p = 0.0005$ .

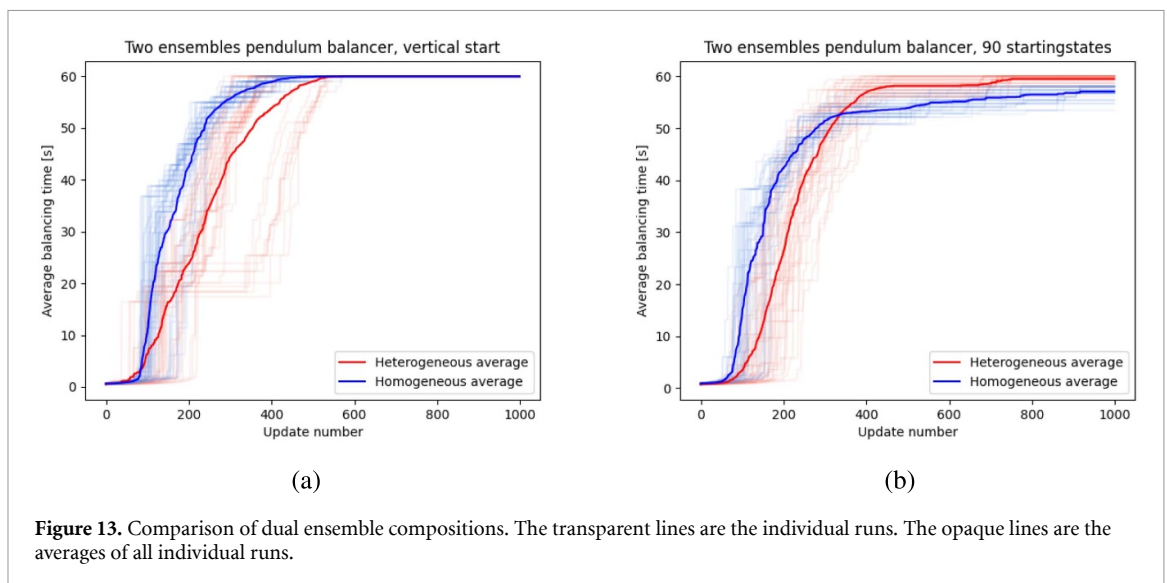
From the same experiment, the average balancing time per update is kept track of and is plotted in figure 12.

In the vertical start runs, we can see that the homogeneous updating architecture learns faster than the heterogeneous one, while both will converge to a configuration where the system is able to perfectly balance all starting conditions. In the harder task where the initial pole position is one of 90 states as seen in the Monte Carlo search, the heterogeneous system seems to peak higher and generally seems to have a higher variance. It appears that the working range (pendulum angle and angular velocity) needs to be limited during the training procedure, because configuration which might be impossible to optimize for while also keeping the system optimized for other configurations, may cause divergence in the training due to a competition between opposing controllers.

<sup>5</sup>  $\{-1.5, -1.45, -1.4, -1.35, -1.3, -1.25, -1.2, -1.15, -1.1, -1.05, -1, -0.95, -0.9, -0.85, -0.8, -0.75, -0.7, -0.65, -0.6, -0.55, -0.5, -0.45, -0.4, -0.3, -0.25, -0.2, -0.15, -0.1, -0.05, 0.05, 0.1, 0.15, 0.2, 0.25, 0.3, 0.35, 0.4, 0.45, 0.5, 0.55, 0.6, 0.65, 0.7, 0.75, 0.8, 0.85, 0.9, 0.95, 1, 1.05, 1.1, 1.15, 1.2, 1.25, 1.3, 1.35, 1.4, 1.45, 1.5\}$ .



**Figure 12.** Comparison of average runs, capped on 500 updates. The filled area is the standard deviation of the data at every point. (a) The average run when the initial pole position is vertical. (b) The average run when the initial position is one of 90 random initial states as seen in the Monte Carlo search.



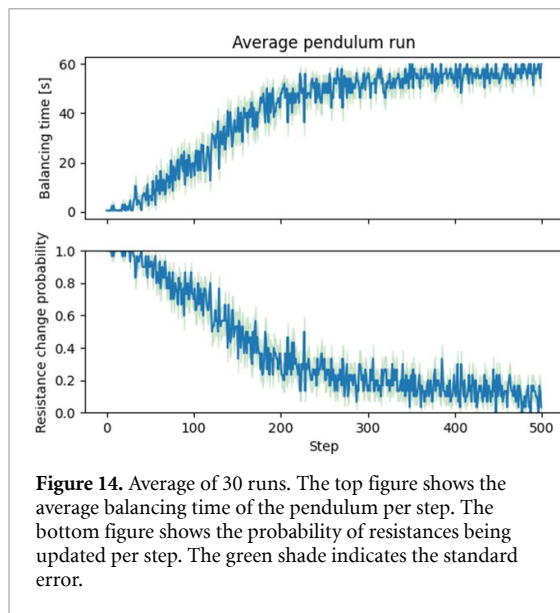
**Figure 13.** Comparison of dual ensemble compositions. The transparent lines are the individual runs. The opaque lines are the averages of all individual runs.

This divergence can be seen especially in the heterogeneous system by means of the decreasing trend after peaking. By using two ensembles that interchangeably learn until they perform better than the other while the other does not learn, learning is more stable. Performance of dual ensembles, consisting of combinations of all 10 previous runs for all 4 settings can be seen in figure 13.

In the case of the vertical starting state, we can see that on average the homogeneous updating system learns faster but the heterogeneous updating system does not lag far behind and reaches the 60 s mark approximately 100 updates later. In the case of the 90 starting states, the homogeneous system again learns faster on average but is surpassed by the heterogeneous system after around 350 updates. the heterogeneous architecture on average balances better than the homogeneous one.

A **single** run consisting of 500 optimization steps with a pendulum length of 1 m, a pendulum weight of

0.2 kg and a cart weight of 0.3 kg is performed. Every optimization step consists of one or more updates, which happen when the pendulum drops or crosses the middle. The resistances are allowed to change  $-10 \text{ k}\Omega | 10 \text{ k}\Omega$  depending on the required direction or  $0 \Omega$  with a 20% probability. One step is over when the pendulum falls or is balanced for 60 s. An example of a single run can be seen in appendix F. The starting angles of the pendulum are approximately  $0^\circ$  and the starting angular velocities are randomly initialized between  $-1.5$  to  $-0.05 \text{ (rad s}^{-1}\text{)}$  and  $0.05$  to  $1.5 \text{ (rad s}^{-1}\text{)}$ . Using the same settings, 30 runs have been performed. The average and standard error of these 30 runs are shown in figure 14. We can see that especially in the first few steps, the pendulum takes some time to be balanced. Later, the learning suddenly speeds up and finally slows down again. Over time, the system is able to balance the pendulum fairly well with an average of the maximally reachable 60 s on a regular basis.



**Table 5.** All values of variable parameters of the pendulum.

Pendulum length (cm)	40, 100, 200
Pendulum weight (kg)	0.2, 0.5, 1
Cart weight (kg)	0.2, 0.5, 1

In order to show that the system can also learn to balance the pendulum for a longer amount of time, a single run with the same settings as the previous runs is performed with a cap of one hour (3600 s). This run can be seen in figure F3.

The working range of the system is explored by varying the simulated physical properties of the cart and the pole (Brockman *et al* 2016). Twenty-seven (27) different combinations of parameters are explored. Three parameters are varied; length of the pendulum, mass of the pendulum and mass of the cart. The parameters and their values can be seen in table 5.

For every combination of values, the optimization process is run five times. Meaning, in total 135 simulations are performed. The system is judged on three criteria; consistency, learning speed and average balancing time.

- The consistency is a measure of how well the system performs once it has reached a balancing time of 60 s the first time. This is calculated by dividing the total number of times it reaches 60 s by the total number of trials performed after the first time the system reached 60 s. It thus shows how well the system is able to keep performing once it has found its first solution.
- The learning speed is the number of trials the systems needs before it is able to reach 60 s for the first time.
- The average balancing time is the average balancing time over the entire run.

The evaluations are plotted in figures 15, G1 and G2. The same data, but instead plotted per pendulum length can be seen in appendix G. These plots are less dense and easier to comprehend.

We will review the average balancing time in figure 15 in more detail. We can see that a fairly wide range of physical parameter settings results in an average balancing time of over 30 s. The system starts to fail to learn when the cart mass is 1 kg and the pendulum length is 0.4 m (i.e. where the pendulum length is the longest and the cart mass is the heaviest). This is indicated by green dots on the bottom of the plot. It should be noted that this physical parameter setting requires fast, powerful actions, due to the combination of a short arm and the relatively large mass. On the other side of the spectrum (i.e. a large pendulum and a light cart indicated by red triangles), the system shows signs of a harder time learning.

### 3.4. Handwriting task

The handwriting task is performed on all letters of the alphabet, for 3 different writings per letter, ten times per writing for both an ensemble of block pulses as well as an ensemble of twitches. An example of fitting a single written letter variant can be seen in figure 16.

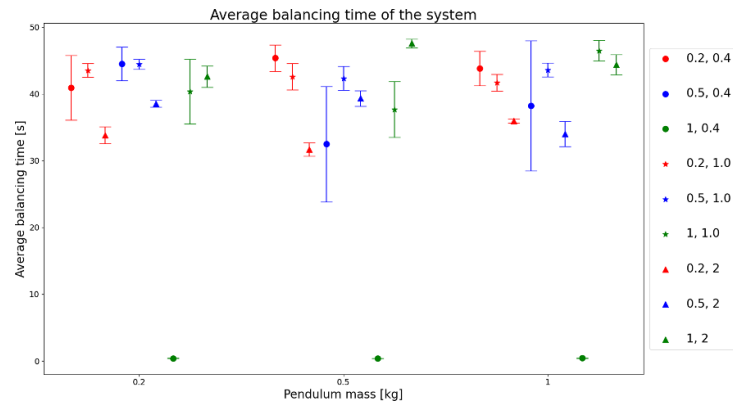
Here we can see that there is quite some variability in the individual fits. Despite this, the average of all fits comes close to the original letter. We can also see that the letter fit by block pulses has sharp corners, while the letter fit by the twitches appears to be more smooth. A histogram of the euclidean distances which shows that the twitches fit the letters better than the block pulses as well as a comparison between individual letter fits can be seen in appendix I).

## 4. Discussion

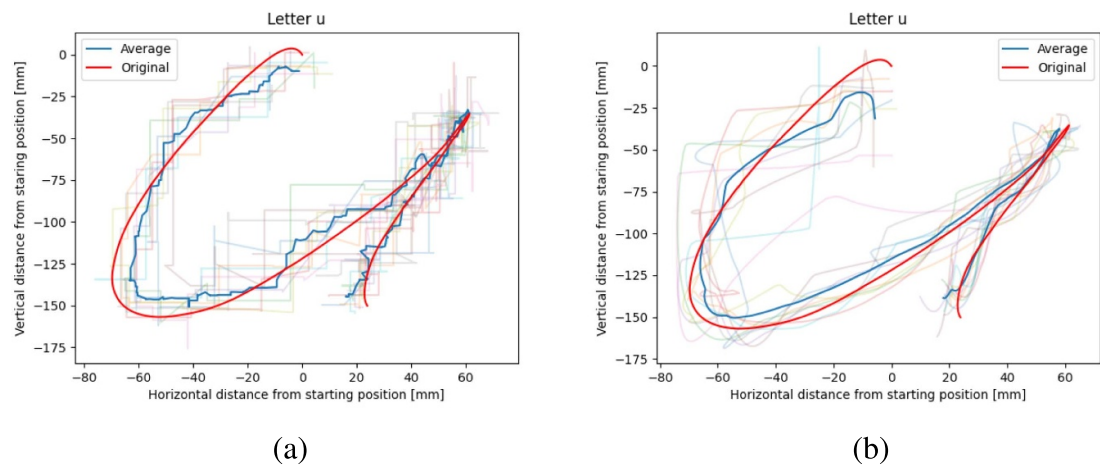
In this study, we propose to take inspiration from biological neuromuscular control mechanisms, both for ‘internal’ neuromorphic computing and for external, multi-actuator mechanical control. A basic electronic circuit is introduced (in NGSPICE) as a synthetic MU, which is able to implement timing, amplitude and a 2nd order overdamped impulse response, by means of a limited number of n-state memristors. By combining several such MUs into an ensemble, arbitrary wave shapes for short-lasting control actions can be realized. The problem of bipolar control signals is solved by defining two pools of motor neurons, one for the ‘agonist’ and one for the ‘antagonist’ direction. The concept is tested on three motor-control tasks involving timing, amplitude and wave-shape requirements: The inverted-pendulum task, the whack-a-mole task, and a pen-tip trajectory generator for handwritten characters.

### 4.1. Modeling results

The proposed approach shows, that in simulation, the three tasks are performed well, with a limited number



**Figure 15.** Average average balancing time over five trials per combination including standard error bars. The physical parameters; cart mass (kg) and pole length (m) are shown in the legend. In three cases, the mass and rod-length parameter combinations were uncontrollable.



**Figure 16.** Example of a single letter variant fit. (a) The letter is fit using block pulses. (b) The letter is fit using twitches. The transparent lines in the background are the individual fits, while the blue line is the average of all the fits.

of motor-unit circuits in an ensemble. Whereas the exact shape of a force burst is less important in the inverted-pendulum task and the whack-a-mole task, the advantage of the ‘twitch’ modelization as a second-order impulse response becomes clear in the handwriting task, where basic block pulses provided a higher RMS error than the twitch function as the basis function in the ensemble.

#### 4.2. Electronics implementation

The current circuit is not a minimal setup and can be simplified in many ways. The goal of this publication is to propose the general principle and leave fundamental improvements to the domain of electronics research. One possibility would be to replace the opamps with field-effect transistors, which also will allow for a good range of usable RC-value combinations due to their high input impedance ( $10^7$ – $10^{12}\Omega$ ). The use of MOSFETS would also reduce the power requirement of the circuit. It would be interesting to find solutions with a minimized component count and optimized routing in a multi-unit design. In analog neuromorphic computing, the temporal task

requirements cannot be overlooked. The component parameters (RC-value combinations) are dependent on the speed of optimal system responses in a particular task. Here, the parameters were geared to simulations of relatively slow processes. Notably, the capacitor is an ‘expensive’ component in a circuit. However, if the spike rates are high and twitch durations are short, low-capacitance MOSCAPs (Arora 1993) could be used in a VLSI design. Alternatively, volatile memristors can be used which have large time-constant values for the decay of a parameter value in the circuit, e.g. resistance (Wang and He 2017).

#### 4.3. Implications for systems with adaptive memristive devices

One of the advantages of a ‘real’ STDP framework is the local learning, on board. In the proposed ETDP model, memristor adaptation needs to be realized by a non-local, external controller. For the MU circuit, five (5) leads would be needed to inject resistance-switching voltages. If these are kept within a reasonable range, damage to the semiconductors

can be avoided. An alternative solution would consist of optically-controlled memristors (Hu *et al* 2021), which have an advantage due to the decoupled operational and training modes and the opportunity for out-of-plane access to the memristive devices on a flat surface, i.e. without additional wiring and routing problems. Since the model is basically concerned with a timed generation of transients, it does not require, in itself, a large-sized 2D crossbar. A 1D linear array of memristors is needed to realize the weighted sum per ensemble, whereas the realization of the bipolar control signal just requires the summation over agonist and antagonist ensemble, for a single control output channel. This does not preclude the use of a crossbar at later or earlier processing stages, it just is not needed within the stage of wave-shape composition by timed twitches.

#### 4.4. Implications for biomimetic actuator design

Whereas the primary goal of this study is to propose neuromorphic approaches for analog control in general, the availability of ensemble-based motor control will be useful for biomimetic control systems, especially if these involve the use of many small actuators with an uncertain, non-linear behavior. For example, in McKibben actuators (Gaylord 1955) or Hasel actuators (Rothmund *et al* 2021), especially if implemented with a large number of contractile elements. A wide range of solutions and patents for (electro-)contractile materials are known (Brock 1991, Mirfakhrai *et al* 2007, Bhatti *et al* 2020, Kuramaya and Ito 2020, Leng *et al* 2021, Phan *et al* 2021, Liu *et al* 2022). Despite a stable research interest over the years, it is difficult to create a breakthrough, because of the non-linearity of control and possible drift and temperature dependencies. A twitch-based controller for individual thin contractile elements that are grouped in parallel, similar to the biological muscle can theoretically compensate for the imperfections (Moghadam *et al* 2015) in the material. For some materials, the second-order filter component in our model can be replaced by the electroactive fiber group, while retaining the population-based timing, amplitude and wave-shape control. In Kuramaya and Ito (2020) an example is given of an artificial muscle exhibiting a second-order overdamped impulse response that appears to be very similar to its counterpart in biological muscles. In this study, we did not address the biological feedback control, which uses muscle length-variation sensors (muscle spindles) and tendon force. Ensemble-based control models (Georgopoulos 1991) play a role, both at the output level and the input level. This means that error signals can be directed back into a motor-unit controller, to compensate for locally non-ideal actuation. As an example, an electro-active polymer with a transparent core can be gauged using laser pulsing using Bragg grating (Hill *et al* 1978, Wolf *et al* 2018), such that the contractile state can be sensed. Other

sensing mechanisms with strain gauges or piezo materials can be envisaged, giving local feedback from individual regions within an artificial muscle. The essence is to design a fine-grained multi-actuator system that is intrinsically prepared for internal wear and tear and external disturbances. This can be realized by point-to-point feedback in an electro-mechanical ensemble system, such that sensors report on the corresponding local regions of contractile elements. This approach is opposed to the traditional technical actuator design which aims at a limited number of presumably perfect components, e.g. using one electromotor per joint. Evidently, a biologically-inspired system will not be the choice if micrometer precision is needed. For many materials, even attaining millimeter precision will be difficult at the scale of humanoid robotics. However, for control of ballistic movements over a wide range of tasks, a spiking analog control model does have attractive properties.

#### 4.5. Incorporation of reflexive feedback

In this work, the overall architecture and the tasks that were used address the feedforward, ballistic type of control, i.e. without considering real-time feedback. In the biological motor apparatus, there is a wide range of sensory (afferent) channels, concerning muscle-fiber length changes from the muscle-spindle sensors, absolute force-level information from the Golgi organ, and nociceptive (pain) signals. For a mechanical agonist/antagonist muscle setup, homonymous excitation and reciprocal inhibition allow for a precise control of mechanical impedance. An example of the highly advanced control mechanisms in the biological system is that the sensitivity of the muscle-spindles is controlled from within even the high level of the motor cortex via the  $\gamma$  motor neuron or fusimotor system, encompassing 30% of the total control signal (Macefield and Knellwolf 2018). Such feedforward mechanisms are especially important for anticipatory control in a system with several delays, in signal propagation as well as in the mechanical force buildup. However, for a fully computational neuromorphic application of the paradigm that uses the twitch ensemble for time function approximation, lack of feedback may not be a limitation and it avoids the complexity in an electronic implementation. Interestingly, the sensory information collection in the dorsal root of the spine also appears to obey recruitment principles, similar to the size principle on the output, i.e. motor-neuron side (Idlett *et al* 2019). Ensemble recruitment also plays a role in central (non-motor) computation (Crowe *et al* 2010, Carrillo-Reid and Yuste 2020). In Khubieh *et al* (2015), it is shown that the recruitment mechanism increases dynamic range and allows for an exploitation of the noise that is present, in pyramidal cortical neural computation, i.e. not only in efferent motor control but also in internal computational processes. Representing

an information quantity using a number of neurons leads to an improved signal-to-noise ratio and dynamic range.

#### 4.6. Gross versus detailed control

An aspect of the fine-grained control in the biological motor system has not yet been addressed. Implicitly, the control signals coming from the upper level, i.e. the primary motor cortex, were assumed to have a gross effect on the motoneuron pool. This means that the details of local adaptation in space (the tissue) and time are left to the computational resources within the segmental level of the spine, allowing the usual recruitment order to take place. This view is consistent with the general approach in, e.g. motor planning in robotics, with a separation of 3D task-space and n-df joint-space planning and abstracted set-point values to be reached by the actuator. However, in the biological system, the central controller can exert, apart from gross influence, also a more detailed control in the recruitment of MUs (Basmajian 1963). If needed, the central controller can choose to modulate the actuator in detail, if a task would require so. In Formento *et al* (2021), subjects were able, after six days of training, to skillfully and independently control three MUs in the biceps to complete a 2D centering task. For intelligent controller systems this means that corrective action for non-ideal components can be realized at two levels: The basic low-level control loop and a higher, strategy-based adaptation of controlling signals, from above. Also here, an ensemble implementation may show its virtue: In the case of many MUs, compensatory strategies by a central controller may be applied with graded effects. The combination of strict hierarchical control in combination with delegated local steering is known as heterarchical control (Cohen 1992).

#### 4.7. Future research

We have presented a first step in the direction of ensemble-based generation of control functions. Several important elements of the biological systems may play a role in more advanced versions of the proposed approach. As an example, for reasons of simplicity, the ‘size principle’ (Henneman 1957) was not deployed in the synthetic MU ensemble. A non-homogeneous distribution of amplitude and time-constant parameter values may allow for achieving a larger dynamic range of responses, both in amplitude and speed of reaction. The distributions of such parameters are highly specific (Elder *et al* 1982) in the human muscular system. In case of new contractile materials, it is evident that the properties of such materials would determine optimal parameter values that need to be determined. Whether a non-homogeneous pool of twitch generators is also beneficial in the case of generic time-function approximation tasks for internal neuromorphic computation is an open research question. In a small experiment using the 20%/80%

rule for m. soleus (Elder *et al* 1982), for slow (Type 1) vs fast (Type 2) synthetic MU distribution in our inverted-pendulum task, no clear benefit of a heterogeneous composition of the ensemble was found (appendix J). The argument of a beneficial increased dynamic range can be countered by the argument that the learning process is simplified in the linear setup of a homogeneous ensemble, where the contribution of the smallest MUs is not larger than the step sizes required for accurate control or precise function approximation. More research will be needed, because there clearly is a potential for any mechanism that is helpful in the reduction of resources (energy, number of electronic components) in neuromorphic control. In artificial systems, the optimalities will likely not fully overlap with those found in the biological system (Dideriksen *et al* 2012). Finally, the approach presented here may have useful implications for neuroprosthetics (Unal *et al* 2018, Lan *et al* 2023, Zhang *et al* 2022).

## 5. Conclusion

This study provided a proof of concept, using hybrid simulation, for the use of a motor-unit twitch circuit that can be used as the basic component in an ensemble setup for generating analog control signals that are timed, with a specific amplitude and wave shape. Results show, that the circuits were well trainable and performed well on three different analog-control tasks: The inverted pendulum, the ‘whack-a-mole’ task and the generation of pen-tip trajectories for letters in handwriting. While the results are preliminary, they open a new view on computational processes within neuromorphic computing hardware but also on control modes for multi-fiber artificial muscle actuators.

## Data availability statement

All data that support the findings of this study are included within the article (and any supplementary files).

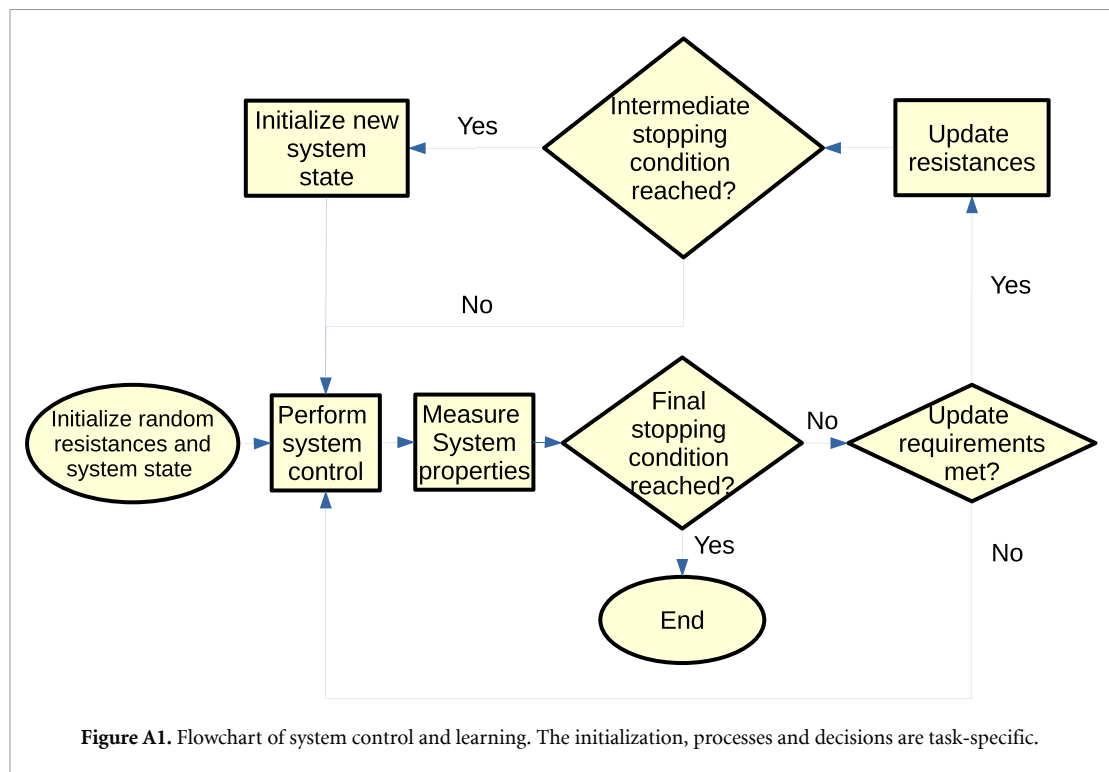
## Acknowledgment

This project was funded by the CogniGron Center, University of Groningen.

## Author contribution statement

Schomaker: general conceptualization, neurophysiology, introduction of NGSPICE modeling, handwriting task, Introduction & Discussion sections; Timmermans: NGSPICE circuit models, experimental task coding and evaluation, Methods & Results sections; Banerjee: materials science and memristor input. Manuscript editing: All authors.

## Appendix A. Control flowchart



## Appendix B. Ensemble compositions for the cart-pole task

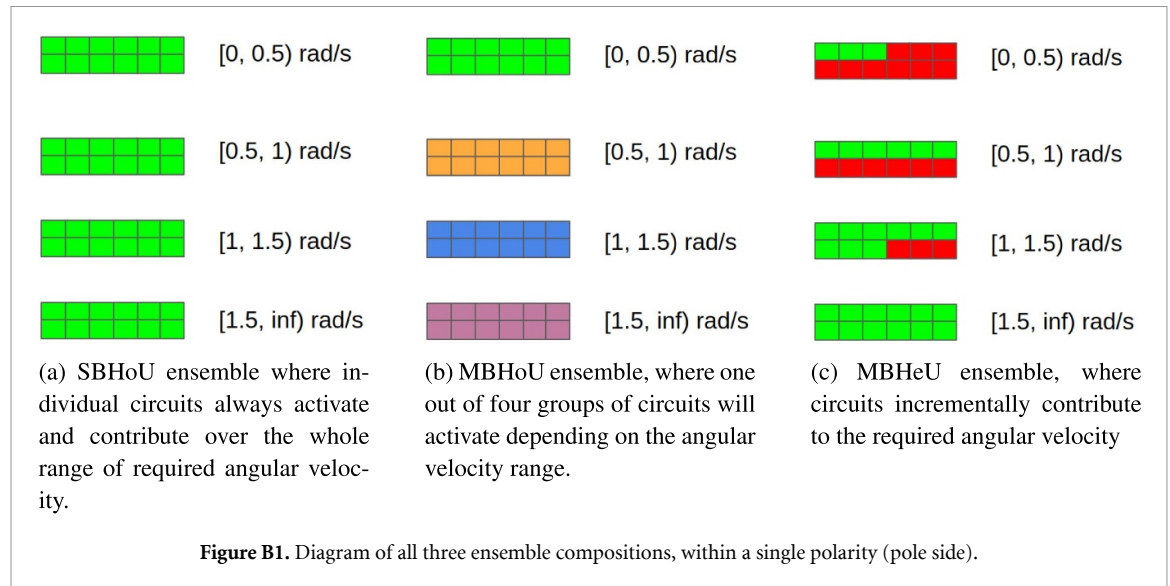
Within the biological muscle, there exist many ways in which MUs are grouped and are involved in a recruitment scheme. For the cart-pole system, we test three different ensembles types, abbreviated as SBHoU, MBHoU and MBHeU.

The SBHoU ensemble of which an example can be seen in figure B1(a), consist of one ensemble for the agonist and one ensemble for the antagonist side. The agonist ensemble will activate when the pole crosses the vertical in the positive angular direction and the antagonist ensemble will activate when the pole crosses the vertical in the negative angular direction.

The MBHoU ensemble (figure B1(b)) consists of multiple ensemble bins per side, where one bin consists of multiple circuits. When the pendulum flips from one side of the vertical to the other, one of the multiple ensemble bins is activated. Which one is activated depends on the instantaneous angular velocity at the point in time of the pendulum crossing the vertical. In practice this can be seen as follows: the angular velocity will be measured at the point

of crossing by a tachometer. This meter generates a certain current. This current then operates a switch. The switch will in turn activate a specific ensemble bin depending on the height of the current, and thus the magnitude of the angular velocity. For the MBHoU ensemble we choose four different circuit bins per side (we thus have eight bins). The first bin will activate when the instantaneous angular velocity is  $[0-0.5) \text{ rad s}^{-1}$ , the second bin will activate when the instantaneous angular velocity is  $(0.5-1] \text{ rad s}^{-1}$ , the rest is visible in table B1 where  $\dot{\Theta}$  is the instantaneous angular velocity at the point of crossing the vertical.

The third approach is the recruitment-based MBHeU approach (figure B1(c)). This approach works the same as the MBHoU ensemble approach except that, with an increasing velocity range, an increasing number of bins will activate (e.g. when the angular velocity is between 0 and  $0.5 \text{ rad s}^{-1}$ , a subset of circuits will activate. When the angular velocity is between 0.5 and  $1 \text{ rad s}^{-1}$ , the same subset will activate together with an additional subset). This method thus activates extra circuits when more force is needed. The activated ensembles per angular velocity range is shown in table B2.



**Table B1.** Multiple ensemble bins approach for ensembles that activate for a specific instantaneous angular velocity range.

Activated bin number (side)	$\dot{\Theta}$ (rad s <sup>-1</sup> )
1 (agonist)	[0, 0.5)
2 (agonist)	[0.5, 1)
3 (agonist)	[1, 1.5)
4 (agonist)	[1.5, $\infty$ )
1 (antagonist)	(0, -0.5)
2 (antagonist)	[-0.5, -1)
3 (antagonist)	[-1, -1.5)
4 (antagonist)	[-1.5, $-\infty$ )

**Table B2.** Recruiting architecture for ensembles that activate for a specific instantaneous angular velocity range.

Activated bin number (side)	$\dot{\Theta}$ (rad s <sup>-1</sup> )
1 (agonist)	[0, 0.5)
1 + 2 (agonist)	[0.5, 1)
1 + 2 + 3 (agonist)	[1, 1.5)
1 + 2 + 3 + 4 (agonist)	[1.5, $\infty$ )
1 (antagonist)	(0, -0.5)
1 + 2 (antagonist)	[-0.5, -1)
1 + 2 + 3 (antagonist)	[-1, -1.5)
1 + 2 + 3 + 4 (antagonist)	[-1.5, $-\infty$ )

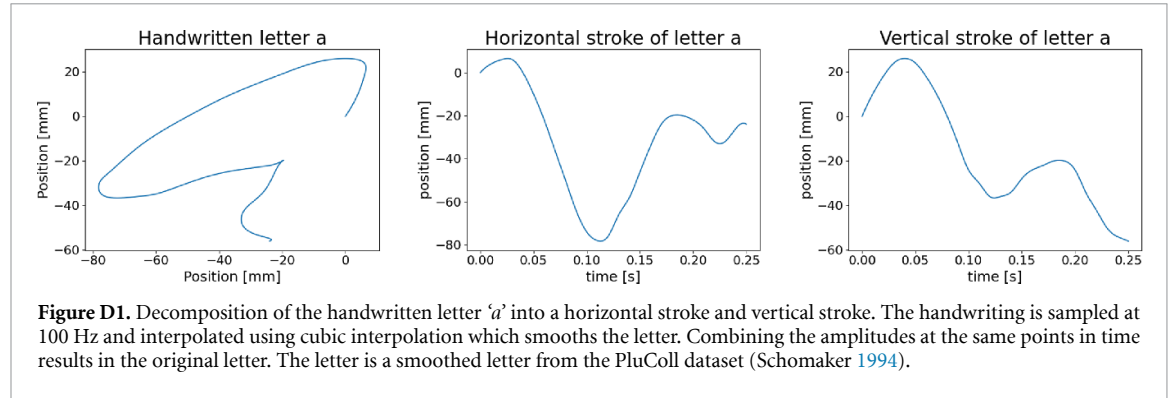
## Appendix C. Whack-a-mole task

In this children's game, a mole pops out from a board with holes and has to be hit with a hammer to gain points. We have simplified the game to one dimension, with a single mole that pops up at a fixed distance from the origin. Rather than ensembles of circuits that alternately activate, all circuits activate simultaneously and voltages are added up to generate a net control voltage. This voltage will drive the hammer forward while net positive and backward

while net negative. The mole is assumed to be hit when the hammer is located above the mole, however, with the requirement of low velocity, implying that a braking action needs to be learned. The update-rules can be seen in table C1. The resistances are only allowed to change  $-50\text{ k}\Omega|50\text{ k}\Omega$  or  $0\text{ k}\Omega$  which will again be determined by the external controller's stochasticity, which allows for a 20% chance of change. Because of the mole's static position, no recruitment is needed to deal with a large dynamic range.

**Table C1.** Resistance update-rules for which a mole. A decrease in resistance reduces time delay and increases amplitude. An increase in resistance results in the opposite.

Condition	Agonist resistances	Antagonist resistances
Hammer exceeds negative $\times$ limit	Decrease	Increase
Hammer exceeds positive $\times$ limit	Increase	Decrease
Hammer in front of mole + time limit exceeded	Decrease	Increase
Hammer past mole + time limit exceeded	Increase	Decrease



## Appendix D. Handwriting task

In some control tasks, block pulses as control signals might suffice. Using a smooth task like handwriting we explore the differences between the twitch shape and a block pulse. Writing a letter can be decomposed into a horizontal and a vertical stroke pattern. Combining the two results in the original letter. An example of the decomposition can be seen in figure D1. By fitting both the horizontal and the vertical stroke, handwriting can be learnt. Please note that small deviations along either one of the axes can already result in an illegible letter if the differential timing is disturbed (Schomaker *et al* 1989). A dataset containing handwritten letters is the PluColl dataset (Schomaker 1994, Vuurpijl and Schomaker 1997), which is available on the Zenodo repository. For all letters of the alphabet, 3 different writing instances per alphabet letter are arbitrarily picked out. The letters are then fit using randomly generated pulses and the Widrow–Hoff learning rule, 10 times per writing pattern for both an ensemble of block pulses as well as an ensemble of twitches. The latter is done to evaluate the assumed advantage of the smooth twitches as opposed to the block pulses as the basis function.

## Appendix E. Preliminary evaluations

### E.1. Single-circuit timing capability, R1 adaptation

The signal timing experiment is performed to see whether we could learn to generate a step signal at a specific moment in time. In order to see whether the

circuit is able to generate a step delay at multiple different times, multiple timing goals have been set. Two different schemes have been used to update resistance R1. In both cases, the initial resistance is set at 100 k $\Omega$  and is updated such that the time delay comes as close as possible to the target time.

The first scheme used is the linear updating scheme where the change in resistance is 10 k $\Omega$  per step or iteration. The time delay of the circuit, taught by a linear updating scheme, seen in figure E1, first goes to a value close to the desired time. It then exceeds this time and starts wiggling around the desired time value because the standard update of 10 k $\Omega$  does not allow for convergence unless the perfect timing is achieved with a resistance value that is a multiple of 10 k $\Omega$ .

The second scheme is the decaying update scheme. In this scheme the initial resistance change is 10 k $\Omega$  and this decays linearly over time, shown in equation (E.1).

$$\Delta R = 10 \text{ k}\Omega * (\text{iterations} - \text{iteration}) / \text{iterations}. \quad (\text{E.1})$$

In the equation,  $\Delta R$  is the change in resistance, 'iterations' is the maximum number of iterations and 'iteration' is the current iteration number. The timing results of the decaying scheme can be seen in figure E2. Using this scheme we can see that at first, the system starts with relatively big updates, getting closer to the desired time fast. Over time the update size decreases, which allows the system to converge.

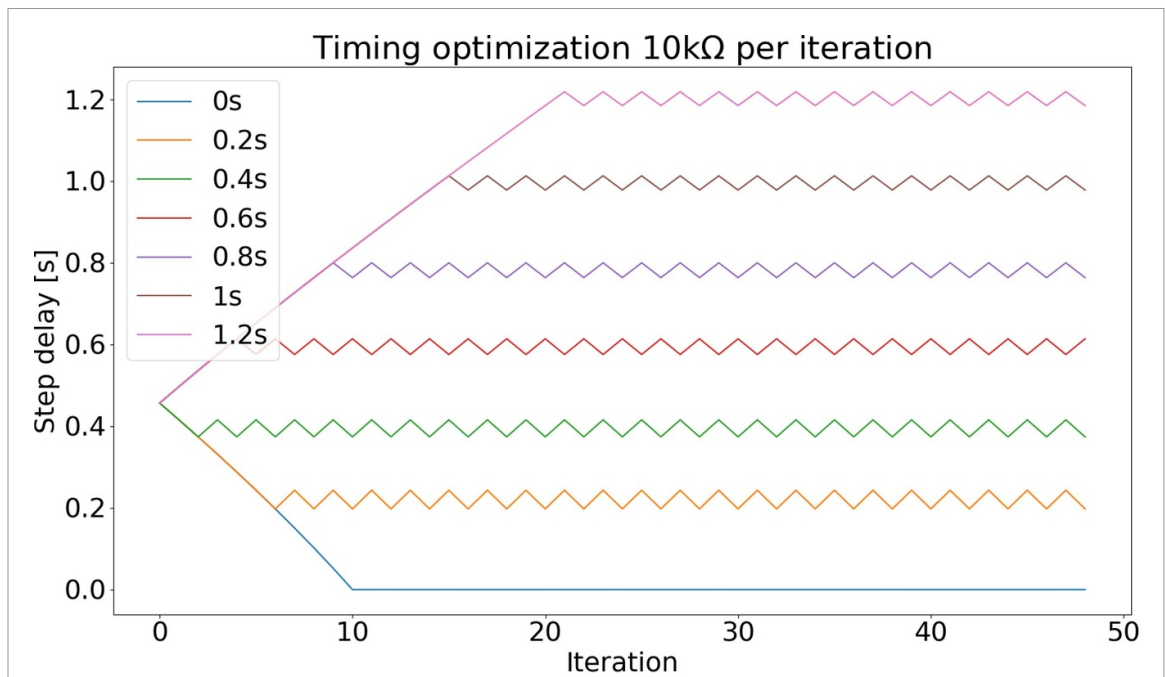


Figure E1. Time delay optimization task for multiple target times with a set update magnitude of 10kΩ. All runs started at 100kΩ.

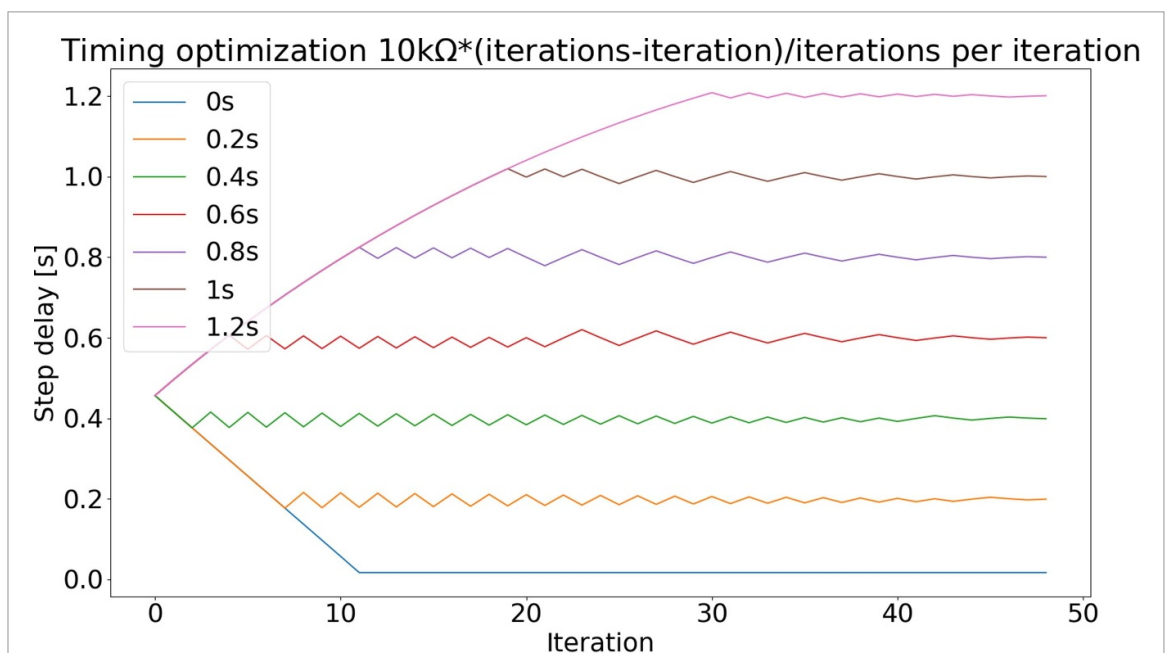
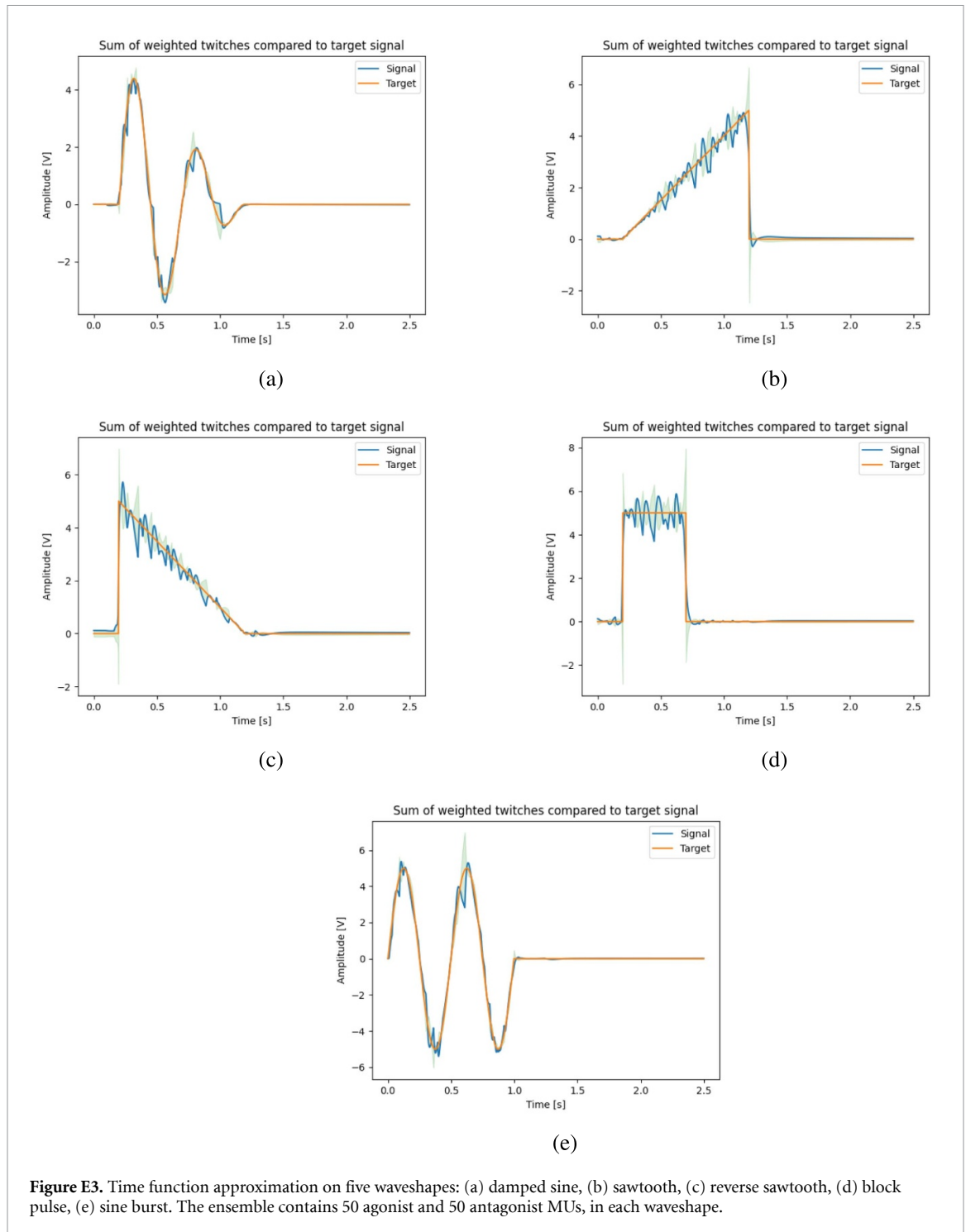


Figure E2. Time delay optimization task for multiple target times with a decaying update magnitude. All runs started at 100kΩ.

## E.2. Evaluation of basic wave-shape fitting

In order to perform a variety of tasks, the system should be able to fit various shapes. Five different shapes have been fit using the Widrow–Hoff learning rule. Before fitting the functions, 10 000 circuit outputs have been generated using random resistance values for R1, R4 and R6. For every waveform, 2000 learning iterations have been performed. For one learning iteration, 100 circuits are sampled, where

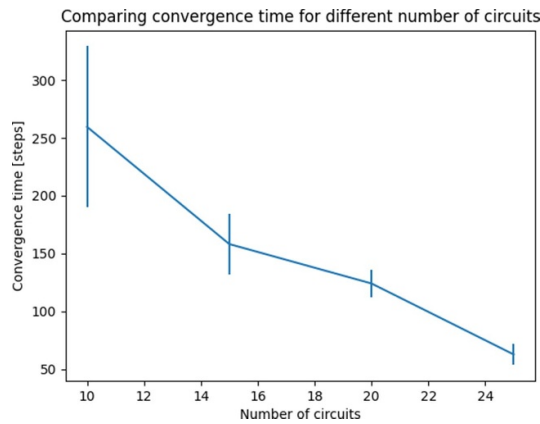
50 of the circuits are used as agonist (contributing positively) and 50 as antagonist (contributing negatively). After sampling, the Widrow–Hoff learning rule is performed for 1000 steps to optimize 102 coefficients (100 circuit outputs, one agonist bias and one antagonist bias) that scale the outputs of every circuit respectively. After every coefficient update, the negative weights are clipped to zero in order to make it more physically plausible (i.e. the agonist should



strictly only contribute positively and the antagonist should strictly only contribute negatively). An issue in fitting the sawtooth is the straight diagonal line. Since the basic twitches only consist of a rounded shape, straight diagonal or horizontal lines are more difficult to fit. This is also visible in for example the block pulse (figure E3(d)). Generally however, considering the variety of signals, the signals fit reasonably well.

### E.3. Cart-pole task, '1 s experiment'

The '1 s' experiment was performed using a different number of circuits in order to see whether the number of signals increases or decreases convergence. The average convergence time with error bars can be seen in figure E4. The convergence duration improves from over 250 steps at 10 circuits down to just above 50 steps when using 25 circuits.



**Figure E4.** Average time it took to learn to get the pendulum straight exactly after one second including error bars. This average time is taken for four ensembles with 10, 15, 20, and 25 circuits, and is performed ten times per ensemble. Convergence is speeding up and becoming more predictable, with an increasing number of circuits in an ensemble.

**Table F1.** Circuit scheme per ensemble. As examples: SBHoU 1 has one ensemble of 15 circuits per side, MBHoU 1 has four ensemble bins (one per angular velocity range) of 15 circuits per side and MBHeU 1 has 15, 20, 25 and 30 circuits for the respective angular velocity range per side. The number of variable resistances is calculated by: number of sides \* number of circuits per side \* number of variable resistances per circuit.

Ensemble name	Ant(agonist) circuit scheme	Number of variable resistances
SBHoU 1	15	$2 * 15 * 3 = 90$
SBHoU 2	20	$2 * 20 * 3 = 120$
SBHoU 3	25	$2 * 25 * 3 = 150$
MBHoU 1	15/15/15/15	$2 * 4 * 15 * 3 = 360$
MBHoU 2	20/20/20/20	$2 * 4 * 20 * 3 = 480$
MBHoU 3	25/25/25/25	$2 * 4 * 25 * 3 = 600$
MBHeU 1	15/15 + 5/15 + 5 + 5/15 + 5 + 5 + 5	$2 * 30 * 3 = 180$
MBHeU 2	20/20 + 5/20 + 5 + 5/20 + 5 + 5 + 5	$2 * 35 * 3 = 210$
MBHeU 3	25/25 + 5/25 + 5 + 5/25 + 5 + 5 + 5	$2 * 40 * 3 = 240$

## Appendix F. Pendulum balancing using ETDP

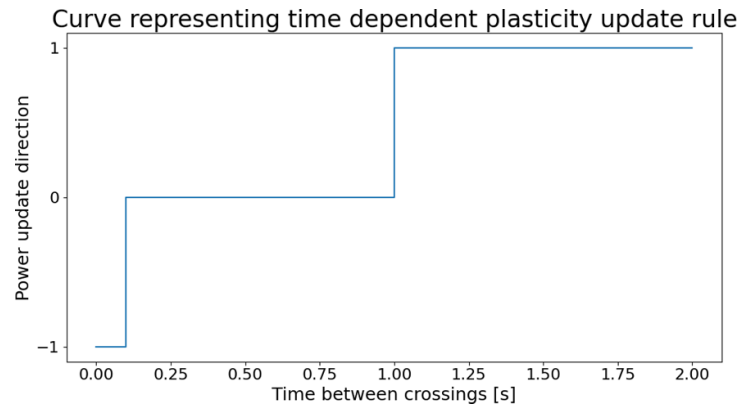
We test three ensemble schemes with three different numbers of resistances per scheme, which can be seen in table F1. The resistances that are updated are R1 for timing and R4 and R6 for shape and amplitude of the signal.

A **single** run consisting of 500 optimization steps with a pendulum length of 1 m, a pendulum weight of 0.2 kg and a cart weight of 0.3 kg is performed. Every optimization step consists of one or more updates, which happen when the pendulum drops or crosses the middle. One step is over when the pendulum falls or is balanced for 60 s. The starting angles of the pendulum are approximately  $0^\circ$  and the starting angular velocities are randomly initialized between  $-1.5$  to  $-0.05$  ( $\text{rad s}^{-1}$ ) and  $0.05$ – $1.5$  ( $\text{rad s}^{-1}$ ).

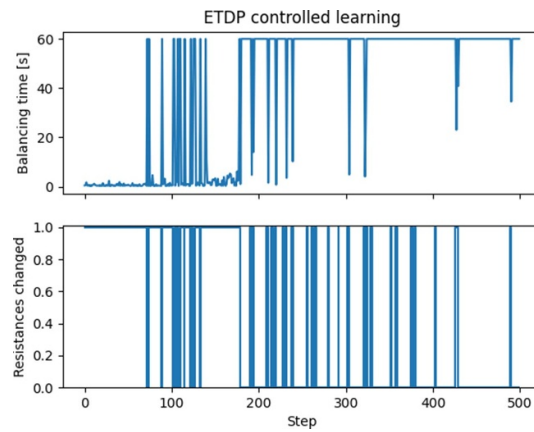
Figure F1 shows the update scenario for the ETDP rule in the pendulum task. For vertical-line crossings that occur much too early ( $t < 0.1$  s) the direction of the memristor adaptation obtains the sign  $-1$ . Within

a dead zone ( $0.1 \text{ s} < t < 1.0 \text{ s}$ ) no updates occur. For  $t > 1.0 \text{ s}$ , the direction of adaptation has the sign  $+1$ . The reason  $0.1 \text{ s}$  is chosen as the lower time limit is that otherwise the learning rule will only allow for an increase in resistance, meaning there is no regularization and it is possible for the resistances to decrease to their minimum value (which results in all circuits generating an  $\approx 8 \text{ V}$  voltage spike at  $t = 0$ ).

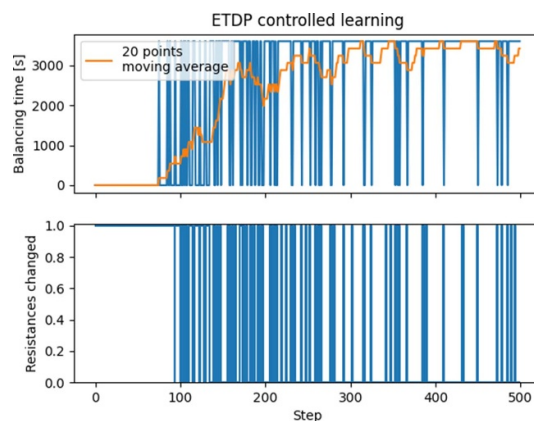
The single run can be seen in figure F2. From this run we can see that the system is helpless initially. It then slowly starts learning and it reaches the balancing time of 60 s more frequently, after around step 180 it barely drops the pendulum. At around 300 steps however, it seems to have gotten into a less stable state, which is quickly corrected as can be seen in the density of lines in the bottom graph. We can also see that even though in some trials, the system has reached its maximum balancing time, resistances are still updated during balancing. This means that even though it knows how to balance the pendulum for a given starting condition, it will still keep optimizing toward the ideal  $0.1$ – $1 \text{ s}$  range as seen in figure F1.



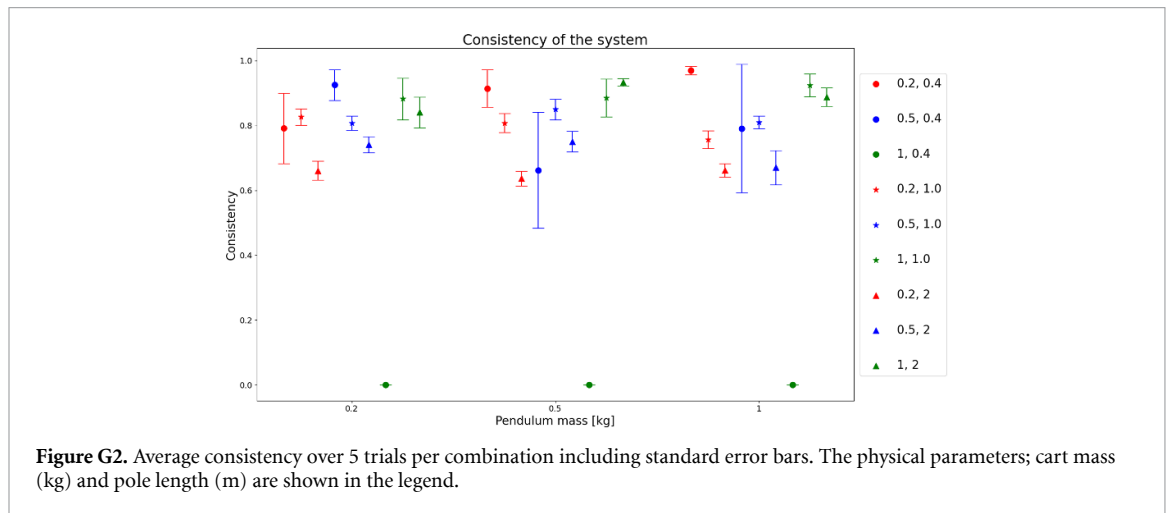
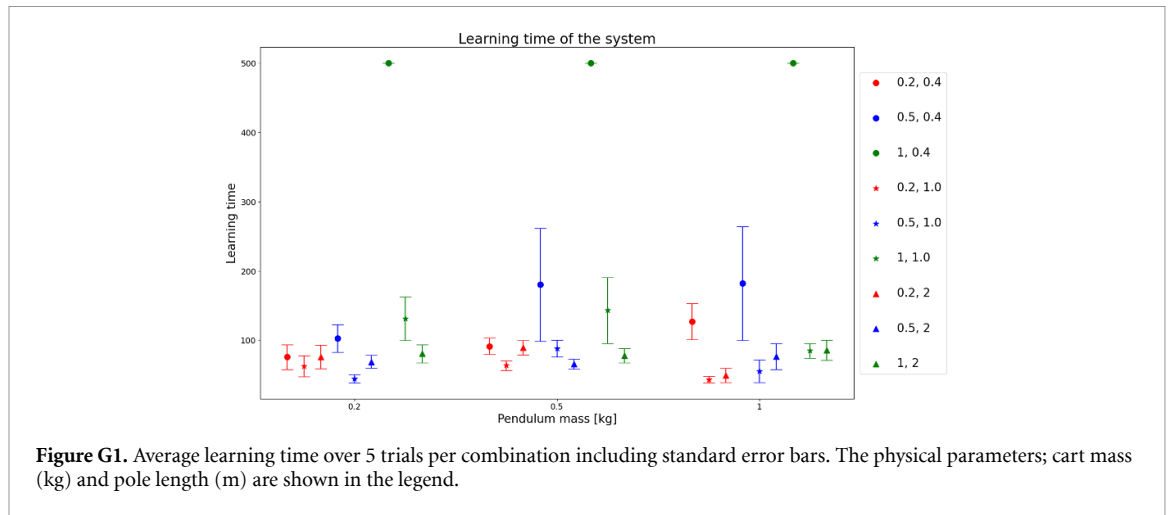
**Figure F1.** Control curve representing update states based on the relative timing of events for the circuits that control the pendulum.



**Figure F2.** Balancing the pendulum using an ETDP update rule during a single training trial (run) of 500 timesteps. The top figure shows the balancing time of the pendulum. The bottom figure shows whether the resistances have changed during balancing (1) or not (0).



**Figure F3.** Single run performance, capped on one hour. Top graph: balancing time in seconds. Bottom graph: resistance update events (1 = change, 0 = no change). The performance keeps improving over time, occasionally dropping in case of difficult initial pendulum states at the boundaries of the operating domain of the circuits.

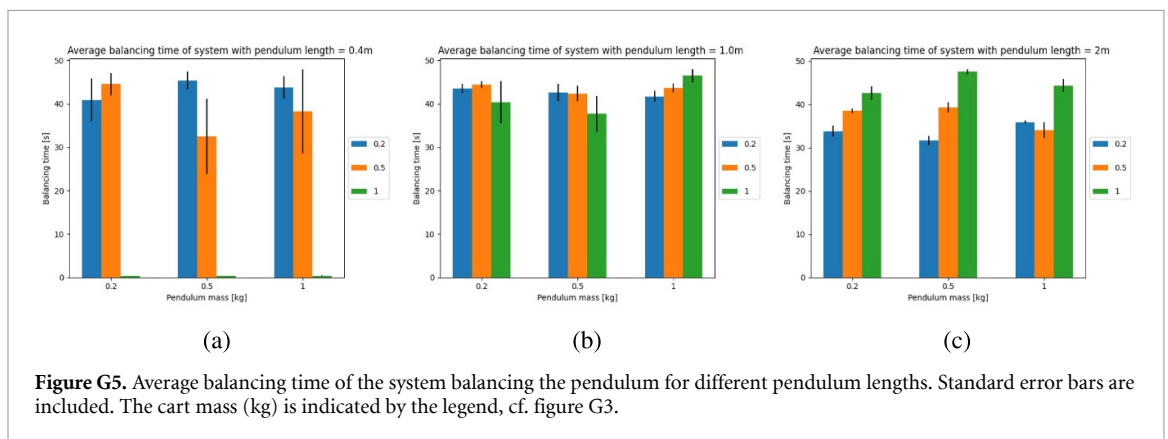
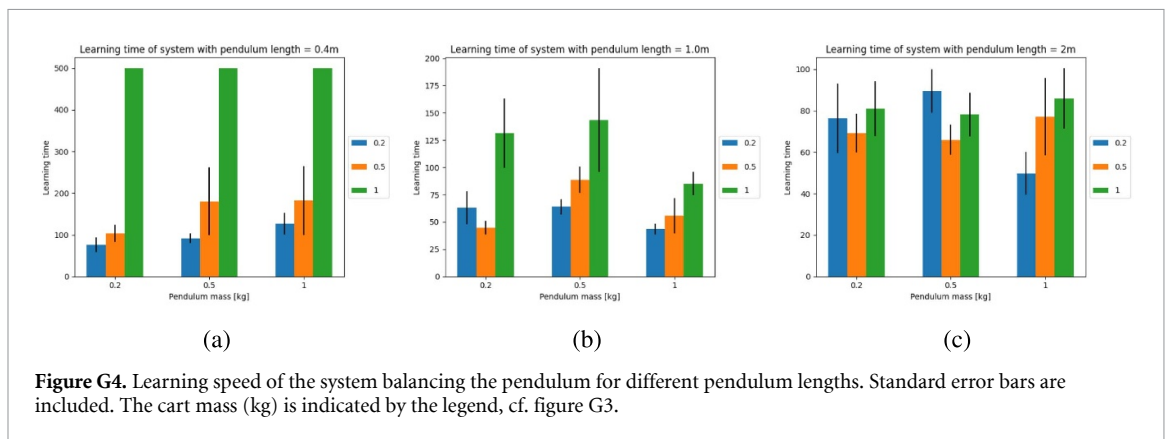
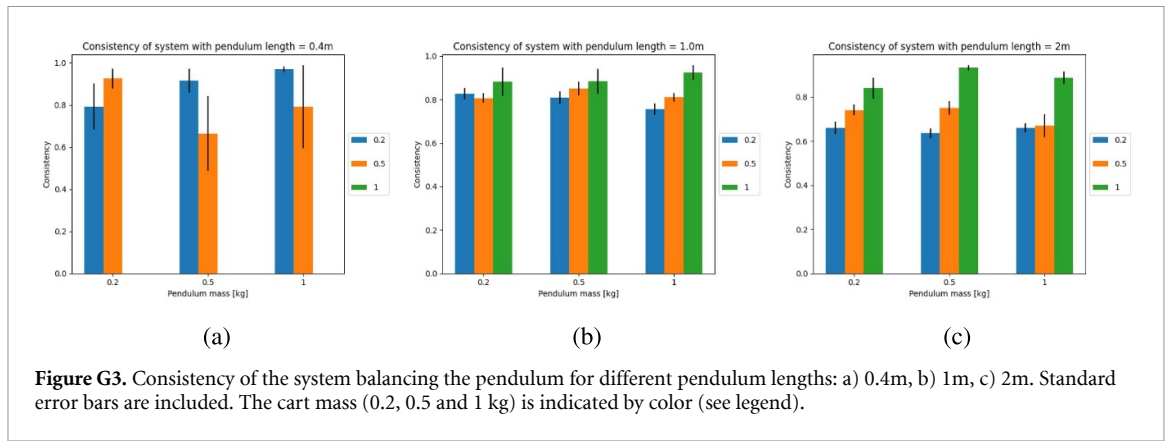


## Appendix G. Pendulum working range

From the learning time graph in figure G1, we can see that it takes a relatively long time to learn to balance the pendulum when the cart mass is the highest (1 kg) and the pendulum is shortest (0.4 m). The rest of the settings seem to be learned fairly fast. By looking at the consistency plot in figure G2 we can see that again the highest cart mass and shortest pendulum setting performs worse. Besides this, the setting with the shortest pendulum length and second to highest cart mass also does not seem to be consistent.

Combining these two observations it seems that in the extreme case the system barely learns. Additionally, when the system is less hard to stabilize (i.e. with a cm of 0.5 kg and a pl of 0.4 m, the system does learn fast but is not able to generalize for all starting states.

From the consistency plot in figure G2 we can see that for most physical parameter-value combinations the system seems to be consistent: When it has learned to balance the pendulum for 60 s once, it generally is able to keep balancing the pendulum for various starting conditions.



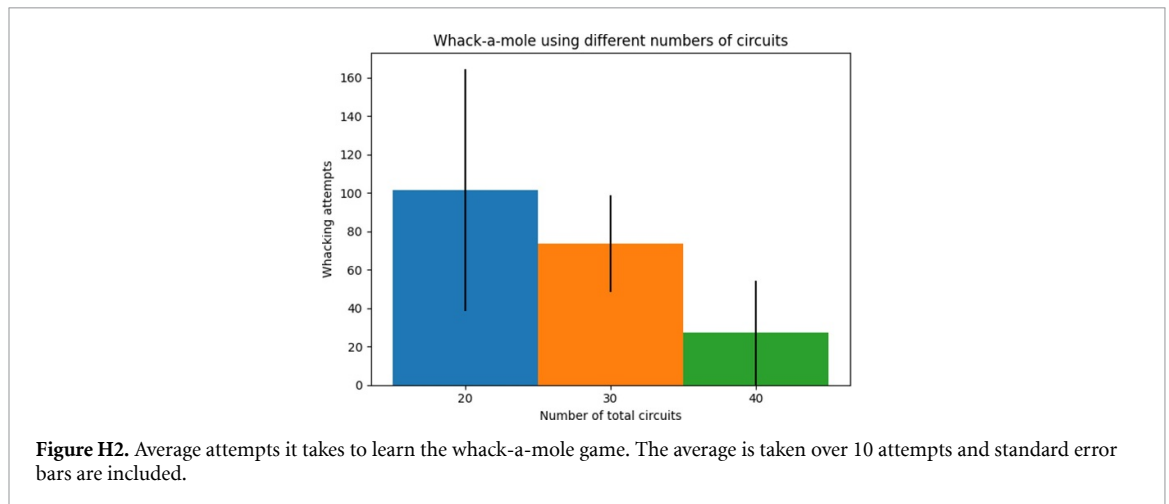
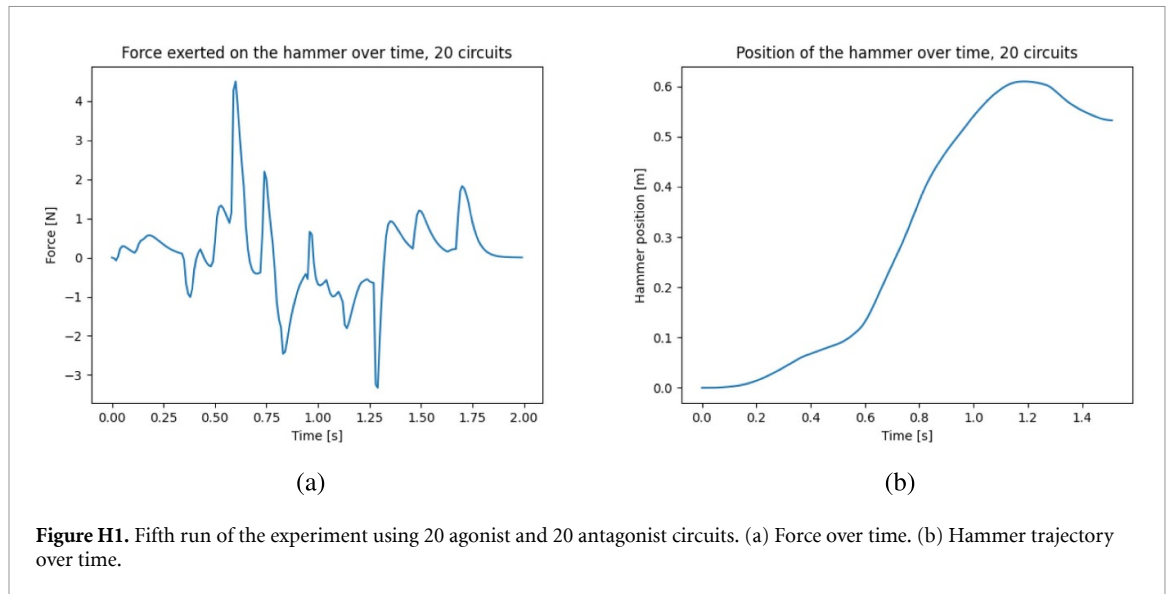
## Appendix H. Whack-a-mole results

### H.1. Whack-a-mole

In the whack-a-mole game the mole, with a length of 0.05 m is put 0.5 m away from the starting point of the hammer. The mole thus covers 0.5 m–0.55 m relative to the starting point of the hammer. The hammer is a ‘point in space’ with a mass of 0.3 kg. The time limit to reach the mole is 1 s and the position limits are –10 cm and 75 cm. When exceeding the limits, the resistances update. We also ran a simulation of multiple runs using a different number of circuits. The

force on the hammer and the trajectory of the hammer of one run, is shown in figure H1.

We can see the hammer speeding up when the force is high, the hammer then overshoots, slows down and is pulled back. This allows the hammer to arrive between 0.5 and 0.55 m with a very low velocity. The comparison between the average of runs with different numbers of circuits can be seen in figure H2. In this figure we can see that although there is a large variability in the attempts, the general trend seems to be that the more circuits are used to learn to play the game, the faster the system



learns to play in terms of the number of attempts. The difference in mean however is not statistically significant when performing the one-way ANOVA test  $F(2, 27) = 0.183, p = 0.872$

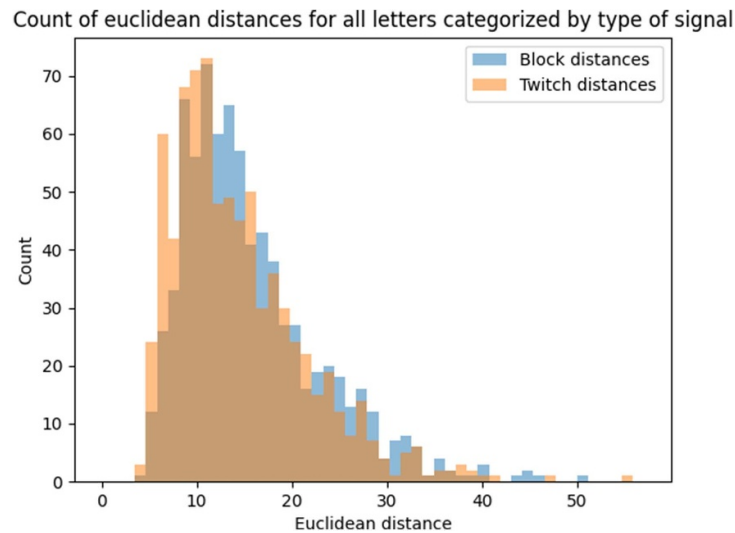
## Appendix I. Handwriting results

In the handwriting task, for each of the 780 fits<sup>6</sup> per signal type, the average Euclidean distance is

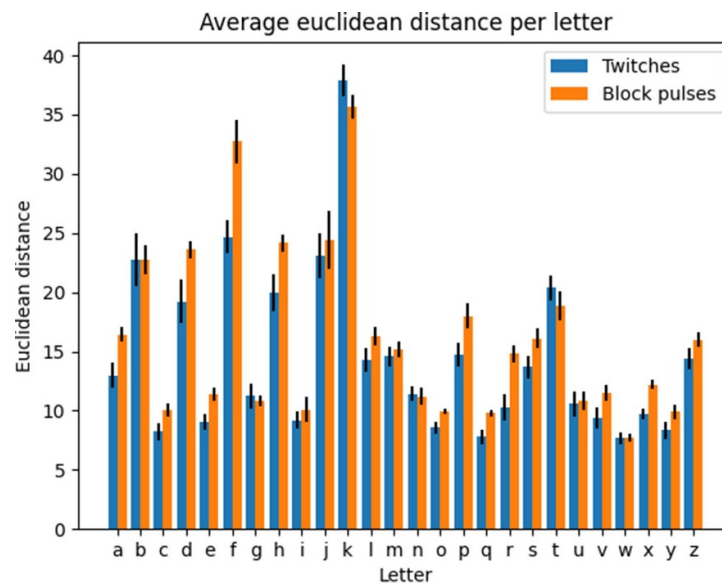
<sup>6</sup> 26 letters, 3 differently written letters per letter, 10 fitting attempts per letter.

calculated as an error measure. Errors of all fits can be seen in figure I1. Here it seems that the distribution of the twitch fits is more left-leaning than the distribution of the block fits. By means of a two-way ANOVA test and the plots in figures I1 and I2, it can be confirmed that the twitches significantly fit the letters better  $F(1, 1508) = 21.3, p = 4.3 \times 10^{-6}$ .

(Figure I2). Here the average and standard error per letter are calculated. We can see that fitting using twitches always has a better or equal performance compared to fitting with block pulses except for the letters *i*, *k* and *t*.



**Figure 11.** Euclidean distances of all fits categorized by fits using block pulses and fits using twitches.

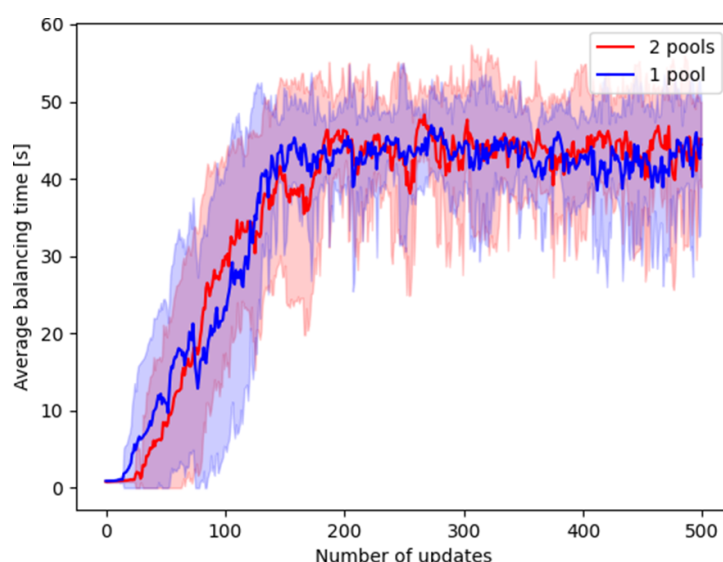


**Figure 12.** Euclidean distance and standard error per letter. The average Euclidean distance shown is the average for a letter over three writers, and 10 model fit attempts per letter, yielding 30 fits. Error bars represent the standard error. Letters differ in their degree of difficulty.

### Appendix J. Evaluation of using a heterogeneous pool with slow and FTs on the inverted-pendulum task

As a first step to future research we evaluated the effect of applying the ‘size principle’ (Henneman 1957) in the composition of the motor-unit pool. Please note that in the case of electronic control there are no constraints that are dictated by the mechanical actuator (e.g. contracting fibers). A distinction was made between FT MUs, using capacitors of 200 nF, and slow-twitch (ST) MUs, using capacitors of 1  $\mu$ F. The

FT MUs had an amplitude range of 0.72–10.2 V. The ST MUs had an amplitude range of 0.1–1.9 V. This check involved a replication of the inverted pendulum task (section 2.3), using 90 possible pendulum starting states and 16 replications. The obtained balancing duration for both a single homogeneous MU pool and a heterogeneous ST/FT pool are measured. Figure J1 shows the result of the simulations for ‘1 pool’ and ‘2 pools’. At the end of the training (500 updates) there are no clear differences. More research is required to explore the benefits of specific twitch amplitude and duration distributions.



**Figure J1.** Comparison of training on the inverted-pendulum task with a single MU pool (1 pool) and a heterogeneous MU ensemble with ST and FTs (2 pool).

## ORCID iDs

L Schomaker  <https://orcid.org/0000-0003-2351-930X>

J Timmermans  <https://orcid.org/0000-0001-8833-5064>

T Banerjee  <https://orcid.org/0000-0001-6848-0467>

## References

- Abadía I, Naveros F, Ros E, Carrillo R R and Luque N R 2021 A cerebellar-based solution to the nondeterministic time delay problem in robotic control *Sci. Robot.* **6** eabf2756
- Abbott L F 1999 Lapique's introduction of the integrate-and-fire model neuron (1907) *Brain Res. Bull.* **50** 303–4
- Arora N 1993 MOS capacitor *MOSFET Models for VLSI Circuit Simulation: Theory and Practice* (Vienna: Springer) pp 121–66
- Barto A G, Sutton R S and Anderson C W 1983 Neuronlike adaptive elements that can solve difficult learning control problems *IEEE Trans. Syst. Man Cybern.* **SMC-13** 834–46
- Basmajian J V 1963 Control and training of individual motor units *Science* **141** 440–1
- Becerikli Y and Celik B K 2007 Fuzzy control of inverted pendulum and concept of stability using java application *Math. Comput. Modelling* **46** 24–37
- Betts G, Desaix P, Johnson E, Johnson J E, Korol O, Kruse D, Poe B, Wise J, Womble M D and Young K A (OpenStax College, Open Textbook Library and OpenStax (Nonprofit organization)) 2013 *Anatomy & Physiology* (Houston, TX: OpenStax College Rice University, BCcampus BC Open Textbook Project)
- Bhatti M R A, Bilotti E, Zhang H, Varghese S, Verpaalen R C P, Schenning A P H J, Bastiaansen C W M and Peijs T 2020 Ultra-high actuation stress polymer actuators as light-driven artificial muscles *ACS Appl. Mater. Interfaces* **12** 33210–8
- Blitzer L 1965 Inverted pendulum *Am. J. Phys.* **33** 1076–8
- Brock D L 1991 Review of artificial muscle based on contractile polymers, AI Memo 1330 *Technical Report* (MIT) (available at: [www.ai.mit.edu/projects/muscle/papers/memo1330/memo1330.html](http://www.ai.mit.edu/projects/muscle/papers/memo1330/memo1330.html))
- Brockman G, Cheung V, Pettersson L, Schneider J, Schulman J, Tang J and Zaremba W 2016 OpenAI gym *CoRR* (arXiv:1606.01540)
- Buchanan K and Mellor J 2010 The activity requirements for spike timing-dependent plasticity in the hippocampus *Front. Synaptic Neurosci.* **2** 11
- Buchthal F and Schmalbruch H 1980 Motor unit of mammalian muscle *Physiol. Rev.* **60** 90–142
- Burke R, Rudomin P and Zajac F 1976 The effect of activation history on tension production by individual muscle units *Brain Res.* **109** 515–29
- Carrillo-Reid L and Yuste R 2020 What is a neuronal ensemble? (<https://doi.org/10.1093/acrefore/9780190264086.013.298>)
- Chua L 1971 Memristor-the missing circuit element *IEEE Trans. Circuit Theory* **18** 507–19
- Cohen A H 1992 The role of heterarchical control in the evolution of central pattern generators *Brain Behav. Evol.* **40** 112–24
- Cox F I, Kuhn W, Murray J and Tynor S 1992 XSPICE: code-level modeling in XSPICE *Proc. IEEE Int. Symp. on Circuits and Systems (ISCAS 92)* vol 2 pp 871–4
- Crochetiere W J 1967 On the use of electrically stimulated muscle as a controlled actuator of a limb *PhD Thesis* Department of health education and welfare, Washington, DC (available at: <https://apps.dtic.mil/sti/pdfs/ADA422620.pdf>)
- Crowe D A, Averbeck B B and Chafee M V 2010 Rapid sequences of population activity patterns dynamically encode task-critical spatial information in parietal cortex *J. Neurosci.* **30** 11640–53
- Davies M et al 2018 Loihi: a neuromorphic manycore processor with on-chip learning *IEEE Micro* **38** 82–99
- De Luca C J 1979 Physiology and mathematics of myoelectric signals *IEEE Trans. Biomed. Eng.* **BME-26** 313–25
- DeWolf T 2021 Spiking neural networks take control *Sci. Robot.* **6** eabk3268
- DeWolf T, Stewart T C, Slotine J-J and Eliasmith C 2016 A spiking neural model of adaptive arm control *Proc. R. Soc. B* **283** 20162134
- Dideriksen J L, Negro F, Enoka R M and Farina D 2012 Motor unit recruitment strategies and muscle properties determine the influence of synaptic noise on force steadiness *J. Neurophysiol.* **107** 3357–69
- Dirac P 1947 § 15 the  $\delta$  function *The Principles of Quantum Mechanics* 3rd edn (Oxford: Oxford University Press) p 58

- Donaldson P E K 1960 Error decorrelation: a technique for matching a class of functions *Proc. 3rd Int. Conf. on Medical Electronics* pp 173–8
- Ehrlich M and Tsur E E 2021 Neuromorphic adaptive body leveling in a bioinspired hexapod walking robot 2021 *IEEE Biomedical Circuits and Systems Conf. (BioCAS)* (IEEE) (<https://doi.org/10.1109/BioCAS49922.2021.9644943>)
- Ehrlich M, Zaidel Y, Weiss P L, Yekel A M, Gefen N, Supic L and Tsur E E 2022 Adaptive control of a wheelchair mounted robotic arm with neuromorphically integrated velocity readings and online-learning *Front. Neurosci.* **16** 1007736
- Elder G C, Bradbury K and Roberts R 1982 Variability of fiber type distributions within human muscles *J. Appl. Physiol.* **53** 1473–80
- Feinstein B, Lindegård B, Nyman E and Wohlfart G 1955 Morphologic studies of motor units in normal human muscles *Cells Tissues Organs* **23** 127–42
- Formento E, Bottos P and Carmenta J M 2021 Skilled independent control of individual motor units via a non-invasive neuromuscular-machine interface *J. Neural Eng.* **18** 066019
- Gauvain J-L and Lee C-H 1994 Maximum a posteriori estimation for multivariate Gaussian mixture observations of Markov chains *IEEE Trans. Speech Audio Process.* **2** 291–8
- Gaylord R H 1955 Fluid actuated motor system and stroking device (Filed by Clevite Corp.) *US Patent* US2844126A
- Georgopoulos A P 1991 Higher order motor control *Annu. Rev. Neurosci.* **14** 361–77
- Gerritsen K 2020 Motor unit properties affect the excitation-force relation of a muscle: a simulation study *Master's Thesis* TU Delft Mechanical, Maritime and Materials Engineering, Delft University of Technology, The Netherlands
- Gutiérrez S, Krumke S O, Megow N and Vredevelde T 2006 How to whack moles *Theor. Comput. Sci.* **361** 329–41
- Hartenberg R S and Denavit J 1964 *Kinematic Synthesis of Linkages* (New York: McGraw-Hill)
- Henneman E 1957 Relation between size of neurons and their susceptibility to discharge *Science* **126** 1345–7
- Hill K O, Fujii Y, Johnson D C and Kawasaki B S 1978 Photosensitivity in optical fiber waveguides: application to reflection filter fabrication *Appl. Phys. Lett.* **32** 647–9
- Hornik K, Stinchcombe M and White H 1989 Multilayer feedforward networks are universal approximators *Neural Netw.* **2** 359–66
- Houk J and Henneman E 1967 Responses of Golgi tendon organs to active contractions of the soleus muscle of the cat *J. Neurophysiol.* **30** 466–81
- Hu L, Yang J, Wang J, Cheng P, Chua L O and Zhuge F 2021 All-optically controlled memristor for optoelectronic neuromorphic computing *Adv. Funct. Mater.* **31** 2005582
- Hulliger M 1984 The mammalian muscle spindle and its central control *Reviews of Physiology, Biochemistry and Pharmacology* (Berlin: Springer) pp 1–110
- Idlett S, Halder M, Zhang T, Quevedo J, Brill N, Gu W, Moffitt M and Hochman S 2019 Assessment of axonal recruitment using model-guided preclinical spinal cord stimulation in the *ex vivo* adult mouse spinal cord *J. Neurophysiol.* **122** 1406–20
- Ielmini D and Waser R (eds) 2016 *Resistive Switching* (New York: Wiley)
- Kang T S and Banerjee A 2018 Learning feedforward and recurrent deterministic spiking neuron network feedback controllers (arXiv:1708.02603)
- Kanosue K, Yoshida M, Akazawa K and Fujii K 1979 The number of active motor units and their firing rates in voluntary contraction of human brachialis muscle *J. Physiol. Sci.* **29** 427–43
- Khubieh A, Ratté S, Lankarany M and Prescott S A 2015 Regulation of cortical dynamic range by background synaptic noise and feedforward inhibition *Cereb. Cortex* **26** 3357–69
- Kuramaya S and Ito F 2020 Antagonistic drive mechanism to increase impulsive force inspired by exoskeleton spring system of mantis shrimp *Proc. 23rd Int. Conf. on Climbing and Walking Robots and the Support Technologies for Mobile Machines* (CLAWAR Association Ltd) (<https://doi.org/10.13180/clawar.2020.24-26.08.63>)
- Lan N, Zhang J, Zhang Z, Chou C, Rymer W Zev, Niu C M and Fang P 2023 Biorealistic hand prosthesis with compliance control and noninvasive somatotopic sensory feedback *Prog. Biomed. Eng.* **5** 023001
- Leng X, Hu X, Zhao W, An B, Zhou X I and Liu Z 2021 Recent advances in twisted-fiber artificial muscles *Adv. Intell. Syst.* **3** 2000185
- Li Z, Liu J, Huang Z, Peng Y, Pu H and Ding L 2017 Adaptive impedance control of human–robot cooperation using reinforcement learning *IEEE Trans. Ind. Electron.* **64** 8013–22
- Liang P, Yang C, Wang N, Li Z, Li R and Burdet E 2014 Implementation and test of human-operated and human-like adaptive impedance controls on baxter robot *Advances in Autonomous Robotics Systems* (Cham: Springer) pp 109–19
- Liddell E G T and Sherrington C S 1925 Recruitment and some other features of reflex inhibition *Proc. R. Soc. B* **97** 488–518
- Lillicrap T P, Hunt J J, Pritzel A, Heess N, Erez T, Tassa Y, Silver D and Wierstra D 2019 Continuous control with deep reinforcement learning (arXiv:1509.02971)
- Liu X, Ji H, Liu B and Yang Q 2022 All-solid-state carbon-nanotube-fiber-based finger-muscle and robotic gripper *Int. J. Smart Nano Mater.* **13** 64–78
- Macefield V G and Knellwolf T P 2018 Functional properties of human muscle spindles *J. Neurophysiol.* **120** 452–67
- Mainen Z F and Sejnowski T J 1995 Reliability of spike timing in neocortical neurons *Science* **268** 1503–6
- Milner-Brown H S, Stein R B and Yemm R 1973 The contractile properties of human motor units during voluntary isometric contractions *J. Physiol.* **228** 285–306
- Mirfakhrai T, Madden J D and Baughman R H 2007 Polymer artificial muscles *Mater. Today* **10** 30–38
- Moghadam A A A, Kouzani A Z, Zamani R, Magniez K and Kaynak A 2015 Nonlinear large deformation dynamic analysis of electroactive polymer actuators *Smart Struct. Syst.* **15** 1601–23
- Nagel L et al 2011 The 40th anniversary of SPICE: an IEEE milestone [guest editorial] *IEEE Solid-State Circuits Mag.* **3** 4–54
- Phan P T, Hoang T T, Thai M T, Low H, Lovell N H and Do T N 2021 Twisting and braiding fluid-driven soft artificial muscle fibers for robotic applications *Soft Robot.* **9** 820–36
- Prochazka A 1996 *Feedback Control of Movement* (Berlin: Springer) pp 1560–4
- Rothmund P, Kellaris N, Mitchell S K, Acome E and Keplinger C 2021 Hasel artificial muscles for a new generation of lifelike robots—recent progress and future opportunities *Adv. Mater.* **33** 2003375
- Sahlin K 1986 Muscle fatigue and lactic acid accumulation *Acta Physiol. Scand. Suppl.* **556** 83–91
- Schlink B R, Nordin A D, Brooks C N and Ferris D P 2021 Fatigue induces altered spatial myoelectric activation patterns in the medial gastrocnemius during locomotion *J. Neurophysiol.* **125** 2013–23
- Schomaker L R 1992 A neural oscillator-network model of temporal pattern generation *Hum. Mov. Sci.* **11** 181–92
- Schomaker L R 1994 The UNIPEN/NICI/HP data collection of summer/autumn 1994 *Technical Report* (NICI, University of Nijmegen and Int. Unipen Foundation) (<https://doi.org/10.5281/zenodo.7894283>)
- Schomaker L R, Thomassen A J and Teulings H-L 1989 A computational model of cursive handwriting *Computer Recognition and Human Production of Handwriting* (Singapore: World Scientific) pp 153–77
- Sejnowski T 2018 *The Deep Learning Revolution* (Cambridge, MA: MIT Press)
- Simony E, Bagdasarian K, Herfst L, Brecht M, Ahissar E and Golomb D 2010 Temporal and spatial characteristics of vibrissa responses to motor commands *J. Neurosci.* **30** 8935–52

- Smith I C, Adam H and Herzog W 2020 A brief contraction has complex effects on summation of twitch pairs in human adductor pollicis *Exp. Physiol.* **105** 676–89
- Strasnick E, Agrawala M and Follmer S 2021 Coupling simulation and hardware for interactive circuit debugging *Proc. 2021 CHI Conf. on Human Factors in Computing Systems (CHI '21)* (New York: Association for Computing Machinery)
- Strukov D B, Snider G S, Stewart D R and Williams R S 2008 The missing memristor found *Nature* **453** 80–83
- Sun B, Ranjan S, Zhou G, Guo T, Xia Y, Wei L, Zhou Y and Wu Y 2021 Multistate resistive switching behaviors for neuromorphic computing in memristor *Mater. Today Adv.* **9** 100125
- Tax A A M and van der Gon J J D 1991 A model for neural control of gradation of muscle force *Biol. Cybern.* **65** 227–34
- Unal R, Behrens S, Carloni R, Hekman E, Stramigioli S and Koopman B 2018 Conceptual design of a fully passive transfemoral prosthesis to facilitate energy-efficient gait *IEEE Trans. Neural Syst. Rehabil. Eng.* **26** 2360–6
- Van Boxtel A and Schomaker L R 1983 Motor unit firing rate during static contraction indicated by the surface EMG power spectrum *IEEE Trans. Biomed. Eng.* **BME-30** 601–9
- Vuurpijl L and Schomaker L 1997 Finding structure in diversity: a hierarchical clustering method for the categorization of allographs in handwriting *Proc. 4th Int. Conf. on Document Analysis and Recognition (IEEE)* (<https://doi.org/10.1109/ICDAR.1997.619876>)
- Wang Q and He D 2017 Time-decay memristive behavior and diffusive dynamics in one forget process operated by a 3D vertical Pt/Ta<sub>2</sub>O<sub>5-x</sub>/W device *Sci. Rep.* **7** 822
- Wang Z et al 2019 Reinforcement learning with analogue memristor arrays *Nat. Electron.* **2** 115–24
- Widrow B and Hoff M E 1960 Adaptive switching circuits 1960 *IRE WESCON Convention Record* pp 96–104 (reprinted in *Neurocomputing* MIT Press, 1988)
- Wikipedia 2022 Whac-a-mole children's game (available at: <https://en.wikipedia.org/wiki/Whac-A-Mole>) (18 January 2022)
- Wolf B, Morton J, MacPherson W and van Netten S 2018 Bio-inspired all-optical artificial neuromast for 2D flow sensing *Bioinspir. Biomim.* **13** 026013
- Zhang Z, Zhang J, Luo Q, Chou C, Xie A, Niu C M, Hao M, Lan N and Lan N 2022 A biorealistic computational model unfolds human-like compliant properties for control of hand prosthesis *IEEE Open J. Eng. Med. Biol.* **3** 150–61