Can Signal Delay be Functional? Including Delay in Evolved Robot Controllers

Matthew Egbert*

University of Auckland
  School of Computer Science
  Te Ao Mārama–Centre for
    Fundamental Inquiry
  mde@matthewegbert.com

Andrew Keane

University of Auckland
  Department of Mathematics

Claire Postlethwaite*

University of Auckland
  Department of Mathematics
  c.postlethwaite@auckland.ac.nz

Nelson Wong

University of Auckland
  Department of Mathematics

Abstract   Engineers, control theorists, and neuroscientists often view the delay imposed by finite signal propagation velocities as a problem that needs to be compensated for or avoided. In this article, we consider the alternative possibility that in some cases, signal delay can be used *functionally*, that is, as an essential component of a cognitive system. To investigate this idea, we evolve a minimal robot controller to solve a basic stimulus-distinction task. The controller is constrained so that the solution must utilize a delayed recurrent signal. Different from previous evolutionary robotics studies, our controller is modeled using delay differential equations, which (unlike the ordinary differential equations of conventional continuous-time recurrent neural networks) can accurately capture delays in signal propagation. We analyze the evolved controller and its interaction with its environment using classical dynamical systems techniques. The analysis shows what kinds of invariant sets underlie the various successful and unsuccessful performances of the robot, and what kinds of bifurcations produce these invariant sets. In the second phase of our analysis, we turn our attention to the parameter θ, which describes the amount of signal delay included in the model. We show how the delay destabilizes certain attractors that would exist if there were no delay and creates other stable attractors, resulting in an agent that performs well at the target task.

1   Introduction

After NASA's *New Horizons* flew past Pluto, it turned its antenna toward Earth and began transmitting the data that it had gathered. The distance between the probe and Earth was so vast that even though the radio signals traveled at the speed of light, there was a five-hour delay between the start of the transmission and the receipt of the first bit of data. Electrical signals moving through copper wires travel almost as fast as those radio waves, moving at roughly 95% of the speed of light. When the wire is short, as is the case in a robot's control circuitry, the delay between the emission of a signal and its receipt at the other end of the wire is so brief that it can often be approximated as nonexistent—an instantaneous transfer of information. The situation in your nervous system is different: The signals that travel along the neurons in your body and brain travel at a comparative snail's

---

* Corresponding authors.

pace, with some types of signals taking three seconds to travel a mere meter, a difference of nine orders of magnitude.

Slow signal propagation implies working with information that is old and sending instructions that get to their destination too late, and robots can become remarkably difficult to control when signal delay is significant. Even relatively short delays in signal propagation can be problematic when the signals are involved in a feedback loop, and feedback loops are commonplace in both biological organisms and robotics.[1] For these reasons, roboticists tend to see delay as a problem to be overcome. They embrace the biologically unrealistic conduction velocity of copper, and build robots out of rigid materials that go quickly and precisely to where they're told and stay where they've been put. By doing these things, these engineers can stay in control, working with accurate information and simplified models of their systems that allow them to apply calculus and other mathematical formalisms, not just to predict or describe or control their creations, but even to prove that their systems will perform as desired.

What, if anything, is lost in this approach? Certain modern efforts in robotics, psychology, and philosophy of mind have returned to biology for inspiration, wondering how it is that floppy and squishy biological bodies can be controlled with nervous tissues that are extremely complex, heterogeneous, asynchronous, and seemingly disordered. These researchers are interested not in how brains are similar to computers, but instead in how they are different, and how by coming to understand these differences we might develop a better understanding of natural and artificial forms of agency, cognition, and intelligence.

One emphasis in this perspective is upon the dynamical (i.e., time-related) differences between brains and computers [4, 28]. Nervous tissues have their own autonomous and temporal dynamics. For example, small neural circuits known as central pattern generators produce rhythmic or chaotic activity, and the brain's various rhythms, with their own intrinsic natural frequencies, are famous (if not well understood). Instead of viewing cognition as a kind of computation, the dynamical perspective sees it more generally as the result of an interaction between the autonomous dynamics of the nervous tissues (and other parts of the body) and the world.

The classic, computationalist "brains are like computers" perspective can be made to fit within this approach, but the emphasis is rather different. Computation generally is divided into three distinct and sequential phases: First data is acquired, then it is processed, and then an output is produced. Early efforts in AI such as Shakey the Robot [19] worked this way, and it remains common in modern AI efforts such as image classification, where the data describing an image is entered, then processed so as to produce an output that categorizes the image. In the computationalist's perspective, the important part of the system is how an AI can best build a model from the data and use that model to make intelligent cognition-like decisions. The dynamical perspective is different. It blends data input, processing, and output into one ongoing interaction, less easy to untangle, between coupled dynamical systems. The important part in this perspective is the ongoing interaction between the brain, body, and world [2]. Instead of trying to figure out how a computer program could generate the best possible internal representation of its environment, the question becomes (more broadly) how to create a dynamical system that results in intelligent behavior when coupled to a body in an environment [5]. In this latter approach, the use of manifest internal representations becomes optional, and the set of possible brain architectures grows, as it is no longer committed to notions of computation or internal model creation.

The field of evolutionary robotics has established itself as a popular method for investigating this dynamical perspective. The method consists essentially of using an evolutionary algorithm to tune the parameters of a dynamical system so that when it is coupled to the sensors and motors of a robot, the system performs a target behavior. Once a solution is identified by the genetic algorithm, the dynamics of the whole brain + body + environment system are analyzed to understand how the desired behavior is accomplished. One of the reasons evolutionary robotics is appealing is that it

---

1 For example, sensors that indicate the position of a robot's arm may be used to determine how next to move that arm, which influences the state of the sensor, and so on.

allows us to generate examples of an interesting cognitive behavior that is not as complicated as those found in nature and that thus might be more easily studied and understood. The hope is that an understanding of these artificial systems will inform our understanding of their natural counterparts. A second appealing aspect of this approach is that it does not assume that the brain (i.e., the neural network) is the locus of the system's intelligence, but instead shows how intelligent behavior can emerge from, and in a sense be fundamentally built out of, interaction between brains, bodies, and the environments in which they are situated. A third appealing aspect of evolutionary robotics is that studies often involve simulated bodies and environments, where the entire system is formulated as a set of differential equations, allowing researchers to employ dynamical systems analysis on the entire system (and/or parts of it), providing a formal language (attractors, separatrices, bifurcations, etc.) that can be used to describe *how* the intelligent behavior takes place.

Some of the best examples of work in this area have examined cognitive abilities such as categorical perception [3], associative learning [14], and behaviors performed by the nematode *Caenorhabditis elegans* [15]. In these and most other evolutionary robotics investigations, the neural networks used are continuous-time recurrent neural networks (CTRNNs) [1]. These are described by a set of coupled ordinary differential equations (ODEs), where each equation describes the dynamics of a single *node* in the network (conceptualized as a neuron or group of neurons) and takes the following form:

$$\tau \frac{dy_i}{dt} = -y_i + \sum_j w_{ji} \sigma\left(y_j - \beta_j\right) + I, \tag{1}$$

where y_i is the excitation of node i; β and τ are bias and time constants; σ is the logistic sigmoidal function; w_{ji} is the weight of the connection from node j to node i; and I is an input signal into the node, such as the state of a light sensor on the robot.

CTRNNs are universal dynamical approximators [2], but this does not mean that all dynamical systems are easily or simply approximated by CTRNNs. Of particular interest here, CTRNNs do not easily capture the delayed influence that we discussed at the start of this article, where the change in a variable is directly influenced by the state of a variable *at some time in the past*. To be clear, this is not to say that CTRNNs are ahistorical or that the effects of a perturbation are all immediate. Conventional CTRNNs are recurrent stateful systems, and signals can propagate through these networks in historical ways (i.e., where the current state is the result of its history). But this is qualitatively different than delayed influence, and this difference is demonstrated by the fact that delayed systems cannot in general be described by ODEs, requiring a different mathematical formulation known as delayed differential equations (DDEs).

To elaborate, ODEs take the form $\dot{\mathbf{y}}(t) = f(t, \mathbf{y}(t))$, where the state of the system at time t is the finite vector $\mathbf{y}(t)$ and f is the evolution function. The simplest DDE systems extend ODEs to include one or more historical terms, thus:

$$\dot{\mathbf{y}}(t) = f(t, \mathbf{y}(t), \mathbf{y}(t - \tau_1), \mathbf{y}(t - \tau_2), \ldots), \tag{2}$$

where each τ_j is a fixed parameter. Other forms of DDEs are possible, where for instance a delayed *derivative* of a variable appears on the right-hand side, or where the amount of delay (the τ_j in the equation above) is itself a dynamic variable, or depends on the current state of the system (i.e., $\tau_j = \tau_j(\mathbf{y})$). The inclusion of delay means that unlike an ODE, where the state is completely captured by the finite vector $\mathbf{y}$, the state of a DDE system includes the continuous history of the variables as far back as the longest delay ($\tau = \max_j(\tau_j)$) in the system. The phase space of the DDE is thus no longer finite, but is the infinite-dimensional space $C([-\tau, 0]; \mathbb{R}^n) \times \mathbb{R}$. Here, $C([-\tau, 0]; \mathbb{R}^n)$ is the infinite-dimensional space of continuous functions over the interval $[-\tau, 0]$, and $\tau \in \mathbb{R}$ represents time. This infinite dimensionality often entails more complex dynamics and makes DDE systems more difficult to investigate than ODEs. As a case in point: To specify an initial-value problem, one must specify an initial history (a function segment) over the time interval $[-\tau, 0]$. An often-used choice of

initial history in numerical solutions of DDEs (and the method employed below) is a constant value, but this is only a one-dimensional subspace of the infinite-dimensional possible space of initial histories.

Despite the increase in complexity, provided certain constraints are met (e.g., the delay is not very short), these delayed systems are often numerically tractable. In fact, many physical or biological phenomena have been modeled using delayed differential equations. Examples include climate models, where delay times model, for instance, the time taken for an oceanic wave to travel across the Pacific Ocean [11], and lasers, where the delay times model the transit time for the light to cross an optical cavity [18]. These models can often be simpler to analyze than large systems of ODEs or PDEs, which would be required to otherwise incorporate the modeled delayed effect.

A number of tools exist for Python and Matlab designed for analyzing DDE systems. In the work presented below, the Python package pydelay [10] was used for the numerical integration during evolution, plotting, and analysis, and DDE-Biftool [8, 21], a Matlab package for continuation of solutions to DDEs, was used as part of the bifurcation analysis.

The dynamical approach argues that we should take seriously the time-related properties of nervous and other bodily tissues when trying to understand natural cognition, but a primary methodology used to investigate these ideas (CTRNN-based evolutionary robotics) precludes a dynamical phenomenon known to operate in nervous tissues: delayed influence due to the finite conduction velocity of nervous tissues.

Why have the effects of time delays been omitted so far in evolutionary robotics? The most straightforward explanation is that ordinary differential equations are easier to analyze than delay differential equations. But there may also be a conceptual hangover from the classical computationalist and engineering perspectives, where delay is generally seen as a problem to be overcome rather than as a feature that might contribute to a functional cognitive system. This might cause researchers to think, "Let's first figure out the easier case, where delay is omitted and from there move on to the more difficult and more complicated case where delay is included." "Furthermore," they might argue, "delay may not play an important or interesting role in the generation of intelligent, adaptive functional behaviors." This perspective both props up and is itself supported by the common assumption that neurons are as fast as they can possibly be given the other constraints that they must satisfy. But in the following subsection (a lightly edited version of the review in [7]) we present evidence from neuroscience research that argues against this dismissal of delay; evidence that supports the notion that delay plays an important role in the generation of functional intelligent behavior. Some of this evidence is circumstantial, but in one case that we describe, the finite conduction velocity of certain neurons in octopuses clearly plays an essential role in the production of one of its coordinated activities. This evidence sets the scene for the remainder of the article, which presents our evolutionary-robotics-based investigation of how delay can underlie good performance in a simple cognitive task.

1.1 Delay in Natural and Artificial Systems

Delay is well recognized in neuroscience, where the finite conduction velocity of neurons is known to cause delayed effects. In this context, as in engineering, delay is often characterized as a problem to be compensated for or avoided. For instance, Swadlow and Waxman [24] and Chklovskii et al. [6] suggest that evolution has acted to maximize conduction velocity to the point that any further increase would come at too great a cost in terms of increased neuronal volume, additional metabolic costs, reliability, and so on.

When nervous tissues are involved in reflexive responses to damage caused by cuts, heat, or the like, it makes good sense that they would evolve to conduct as fast as they possibly can. But it is interesting to note that neuronal conduction velocities vary over three orders of magnitude from 0.3 to 120 meters per second [24]. This wide range of conduction velocities is largely the result of two ways that neurons vary: mylenation and axon diameter—thicker, mylenated axons conduct more quickly than those that are thinner or less mylenated [29]. In the peripheral nervous system, neurons associated with pain and temperature conduct more slowly than those associated with muscle activation, and within the central nervous system there exist a wide variety of conduction velocities with
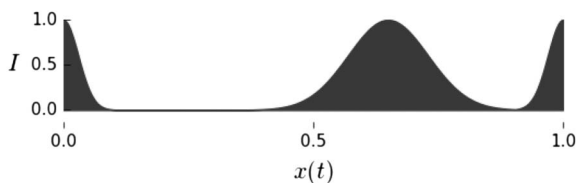
"some axonal systems [that] are nearly exclusively fast-conducting (corticospinal, corticotectal, and sensory thalamocortical axons), some [that] are exclusively slowly conducting, with impulses taking many 10s of ms to reach their terminals (cortically projecting neurons of the locus coeruleus and substantia nigra), and some [that] consist of a broad spectrum of axons (corpus callosum, some corticocortical populations, corticothalamic neurons of layer 6)" [24].

The presence of different conduction velocities does not eliminate the possibility that neurons have evolved to be as fast as possible. Constraints upon different neural systems may vary, in some cases enabling the growth of faster, but more costly neurons. Such constraints, however, do not explain another common feature of neural conduction velocities: They change over time. This change takes place over both long and short time scales. To give examples: Callosal axons can change progressively (1–2%/day) over periods of months [23], while bursts in neuron activity can increase the conduction velocity for seconds afterwards [24].

In many cases, it remains unclear what the function is of this variation and how it is accomplished. What is clear is that such variation in conduction velocity will undoubtedly influence the relative times at which signals arrive at particular neurons, which we would expect to seriously influence neural dynamics. As a case in point, Hebbian learning and other synaptic plasticity rules describe how synapses change as a function of the timing of the firing of downstream and upstream neurons [12, 13, 25]. These mechanisms rely upon signals arriving more or less synchronously, which suggests very precise timing, as the voltage change produced when a synapse fires lasts only 2–4 ms. In some cases the conduction velocity of neurons appears to be tuned so as to synchronize the arrival of stimuli along different-length axons. One such case occurs in peripheral cephalopod nerve fibers. The conduction velocities of these fibers is structured so as to synchronize the arrival time of stimuli that occur at different distances from the organism's central nervous systems [20, 24] (essentially, the shorter fibers conduct at a lower velocity than the longer fibers). Other examples of conduction velocity being modulated so as to synchronize the arrival have also been found—a number of examples can be found in [9].

Here we see the first clear examples of delay playing a functional role. The conduction velocity of these neurons is not just something to be maximized, but should instead (or in addition) be tuned so as to enable synchronous activation of the muscles along an octopus's arm.

The synchronization of signals along axons of different lengths is a straightforward example of delay playing a functional role. It is almost an *anti-temporal* mechanism—a mechanism that erases temporal differences between signals. This kind of mechanism is easy for us to understand—are there also less easily understood functions that delay might perform? Evolutionary robotics, and more broadly the field of artificial life, involves the creation of artificial systems that can be used to aid in the investigation of complex biological phenomena. In the next section, we develop an artificial system that uses delay in signal propagation in part of its sensorimotor feedback loop to solve a simple discrimination task. By understanding how the artificial system takes advantage of signal delay, we hope to expand our understanding of the ways that delay can be used as part of sensorimotor feedback systems or cognitive architectures. Such knowledge would hopefully prove useful in the study of delay in natural systems.

2 Model

To evaluate how delay can contribute to solving a task, we consider a minimal model (first presented in [7]) that consists of a robot situated in a one-dimensional periodic environment of length 1. The robot's "brain" is modeled as a single neuron with one recurrent, delayed, and weighted synaptic connection. The velocity of the robot is determined by the following ODE:

$$\frac{dx}{dt} = \frac{2}{1 + \exp(-4y)} - 1.0, \qquad (3)$$

Figure 1. The state of the sensor (I) is a function of the current position of the robot, $x(t)$. In each trial the location of the peaks is different. This figure shows one example environment.

where x is the robot's position and y is the excitation of its neuron, which is governed by the following DDE:

$$\tau \frac{dy(t)}{dt} = -\gamma y(t)^3 + \omega y(t-\theta) + \psi I + \beta. \tag{4}$$

In this equation, τ is a time-scale parameter and $-\gamma y(t)^3$ formalizes a tendency of the neuron to return to a base level of excitation (the cubic is included so that this tendency grows increasingly strong as the system moves farther away from the base level of excitation, which prevents the system from exploding). The second term describes the influence of the delayed recurrent connection, where ω expresses the weight of the connection and θ describes the delay associated with it. The third term, ψI, indicates a scaled sensory input (described below), and β is a constant bias term.

The input to the neuron (I) is a function of the robot's current position:

$$I = \exp\left(\frac{-(x(t)|p_n)^2}{0.0018}\right) + \exp\left(\frac{-(x(t)|p_w)^2}{0.0128}\right), \tag{5}$$

where $A|B = \min\{|A-B|, |1.0 - |A-B||\}$ represents the distance between points A and B (in accordance with the minimum image convention followed for periodic boundary conditions). The input (I) has been plotted as a function of the robot's current position, $x(t)$, in Figure 1, with the position of the narrow peak ($p_n = 0$) and the position of the wide peak ($p_w = 0.75$). Informally, one can think of there being two hills in the agent's environment, of equal height, but with one narrower and steeper than the other. In this description, I corresponds to the altitude of the robot.

A simple genetic algorithm was used to tune the parameters described by the Greek letters in the equations above, so that after approximately $50s$ (s being the time unit), the robot is located near the narrow peak. To evaluate the fitness of a parameterization, 120 initial-value problems of duration $d \in [45, 55]$ were solved numerically, where the initial starting position of the robot (x_0) and the position of the wide peak (p_w) were taken from the Cartesian product $\{(x_0, p_w) | x_0 \in \{0, 0.05, ..., 1\}$ and $p_w \in \{0.25, 0.35, ..., 0.75\}\}$. In all evaluations, the narrow-peak position is $p_n = 0$. The excitation preceding the trial was set to a constant value, $y_{t\leq 0} = 0$. Each of these 120 initial-value problems were given a score, $S = 0.5 + 0.5(\bar{x}_{t>d-10}|p_n) - 0.5(\bar{x}_{t>d-10}|p_w)$, which is higher when the average distance of the robot during the last $10s$ of the trial ($\bar{x}_{t>d-10}$) is close to the narrow peak and far away from the wide peak. The fitness is then given by $f = \prod_S(\frac{S}{4} + \frac{3}{4})$, the product of all of the 120 scores after they have been scaled to lie within $[0.75, 1]$.

3   Analysis

We performed 32 evolutionary runs, each lasting approximately 3500 tournaments. The parameters of the best-performing individual can be found in Table 1, and the evolutionary change in population fitness in the run that produced this agent is presented in Figure 2. The remainder of this section presents our analysis of this system.

Table 1. Evolved parameters. Parameters specified to three digits of precision. Simulation was conducted with an absolute tolerance of 10^{-6} and a relative tolerance of 10^{-3}.

τ:	0.563	γ:	9.595
ψ:	1.794	β:	−0.272
ω:	−1.297	θ:	1.140

An overall summary of performance can be seen in Figure 3, which indicates the maximum distance between the robot and the narrow peak in the last $10s$ of a $100s$ trial for different initial robot starting positions (horizontal axis) and different placements of the wide peak (vertical axis). The controller performs well in the majority of initial conditions, especially those initial conditions that were evaluated during the evolution, where $p_w \in [0.25, 0.75]$.

The plots in Figure 4 show trajectories taken in various successful and unsuccessful trials, providing a first glimpse of some of the qualitative variety demonstrated in this system. The only things differing in these trials are the placement of the distractor wide peak and the initial position of the robot. These conditions are indicated above each set of plots. The plots on the left show how the robot's position changes over time, and those on the right indicate the state of the robot's neuron excitation (y) plotted against its weighted delayed value, $\omega y(t - \theta)$ (the second term in the RHS of Equation 4). Note that the trajectories in the right column appear to self-intersect only because the phase space is infinite-dimensional and the figure presents a two-dimensional projection of that space.

The diverse dynamics seen in Figure 4 can be explained in terms of the attractors that exist for different values of the parameter p_w. We find the attractors by conducting a bifurcation analysis, using the continuation software DDE-Biftool, to numerically continue (i.e., track) branches of equilibria and periodic orbits while varying p_w. In contrast to simulations, this reveals both stable and unstable solutions. The software can calculate the stability properties of the solutions and identify different types of bifurcations. Understanding how the different solution types are organized in
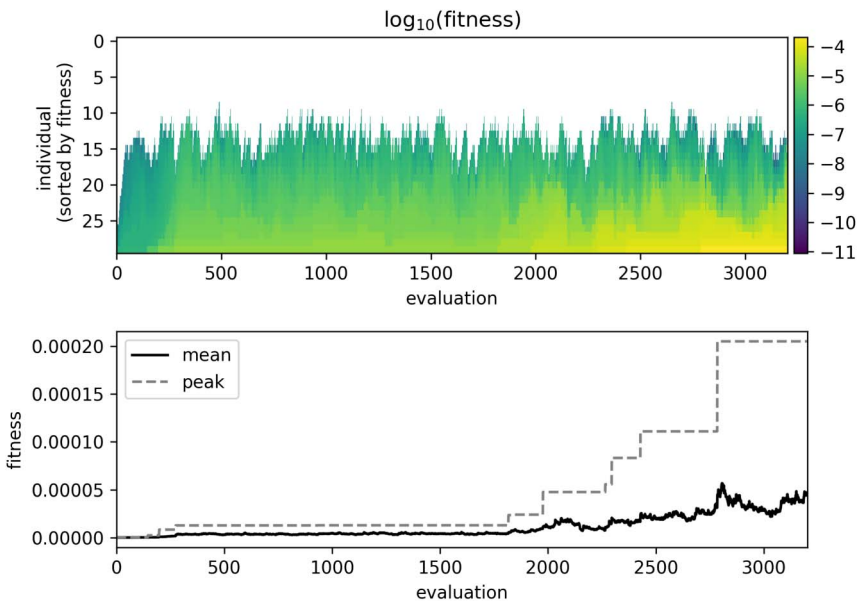
Figure 2. Progression of fitness evaluations during evolution. The top plot shows the distribution of fitness across the population as it evolved. In this figure, each column indicates the fitness of the population (sorted by fitness).
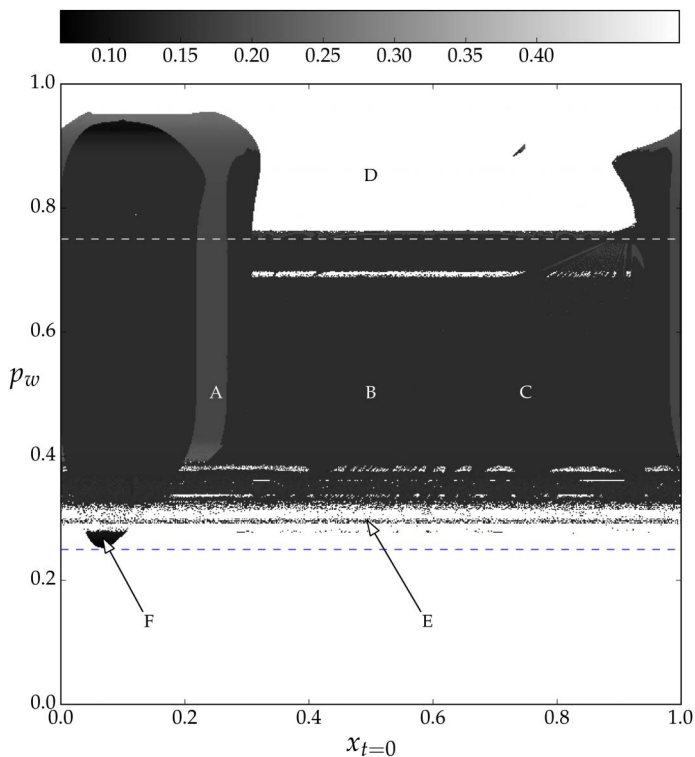
Figure 3. Initial condition survey. Values in this image indicate the maximum distance of the agent from the narrow peak in the last 10s of 100s simulation 100, $\max(x_{90<t<100}|p_n)$ for the parameter (p_w) and initial condition (x_0) values conditions specified on the axes. The dashed lines indicate the limits of the parameter values tested during evolution ($p_w \in [0.25, 0.75]$).

terms of parameters and how they are related through bifurcations gives one an overall picture of the dynamical capabilities of the system.

We now very briefly recall the types of bifurcations we encounter. A formal and detailed description of these bifurcations can be found in any standard nonlinear dynamics textbook, for example, [22]. *Hopf bifurcation:* An equilibrium changes stability, resulting in the birth of a periodic orbit with zero amplitude. After the bifurcation, the amplitude of the periodic orbit grows. *Saddle-node bifurcation:* Two invariant sets (most typically, equilibria or periodic orbits) collide and mutually destroy each other. *Period-doubling bifurcation:* A periodic orbit changes stability, resulting in the birth of another periodic orbit with twice the period of the original. *Torus bifurcation:* A periodic orbit changes stability, resulting in the birth of a two-dimensional torus. *Homoclinic bifurcation:* A periodic orbit is born from the meeting of trajectories along one stable and one unstable direction of a saddle equilibrium. This bifurcation results in the sudden appearance of a nonzero-amplitude, but long-period, periodic orbit.

3.1 Bifurcation Analysis of Best-Performing Individual

We now describe the bifurcation diagram of the system in the parameter p_w (the position of the wide distractor peak) and then explain how this analysis can be interpreted to understand the behavior of our robot. Figure 5 shows equilibria (thin curves) and periodic orbits (thick curves). In this figure, solid lines indicate stable solutions and dotted lines indicate unstable solutions. The horizontal axis represents the x value of the equilibria or the maximum x value of the periodic orbits (recalling that x describes the position of the robot). The Hopf, homoclinic, and saddle-node bifurcations of equilibria are labeled H, C, and SN, respectively.
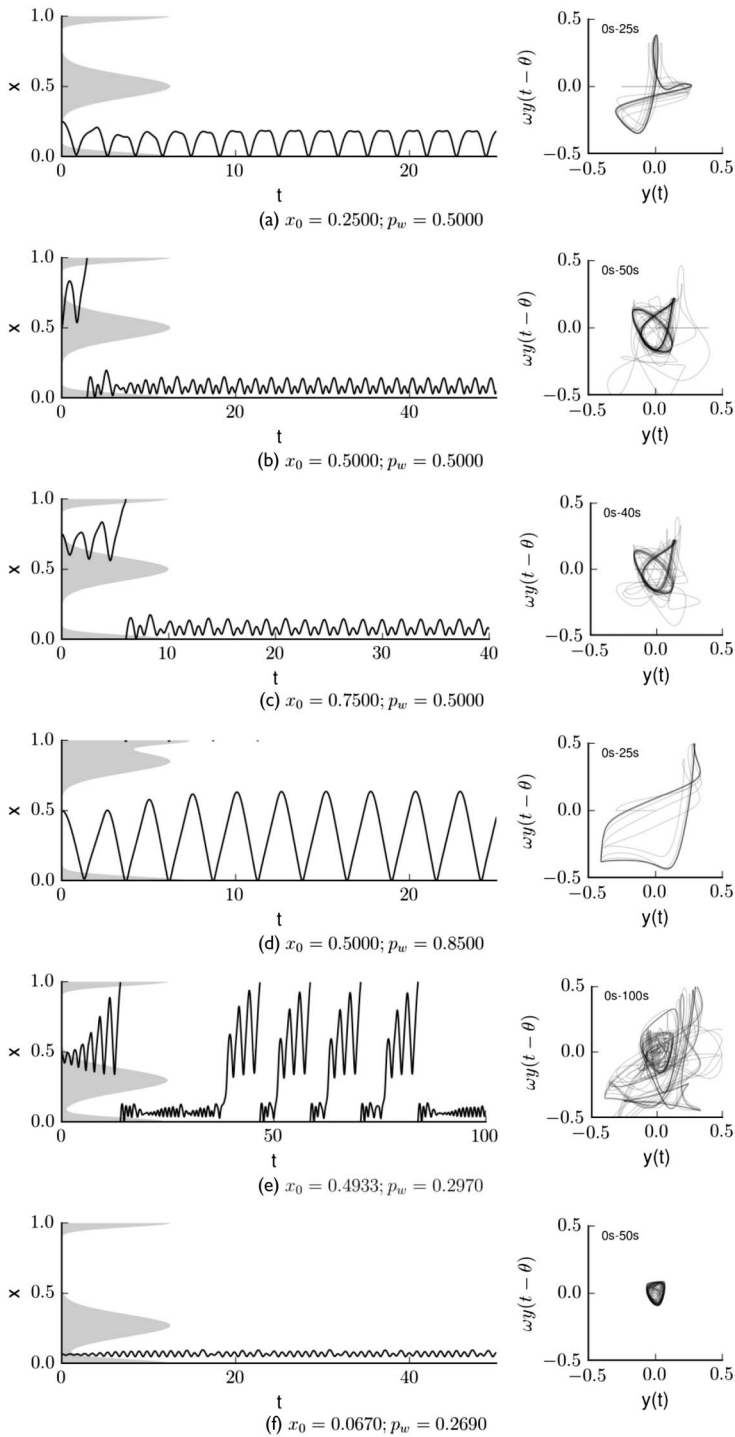
Figure 4. Trajectories for indicated initial conditions (x_0) and environmental parameter values (p_w). Plots on the left indicate the position of the robot as a function of time. The positions of the stimulus peaks are also indicated by the gray areas on the left of the plots. The graphs on the right show the excitation of the robot's neuron ($y(t)$) plotted against its earlier value ($y(t-\tau)$) and scaled by the evolved parameter ω. These phase portraits are 2D projections of the system's infinite-dimensional phase space.
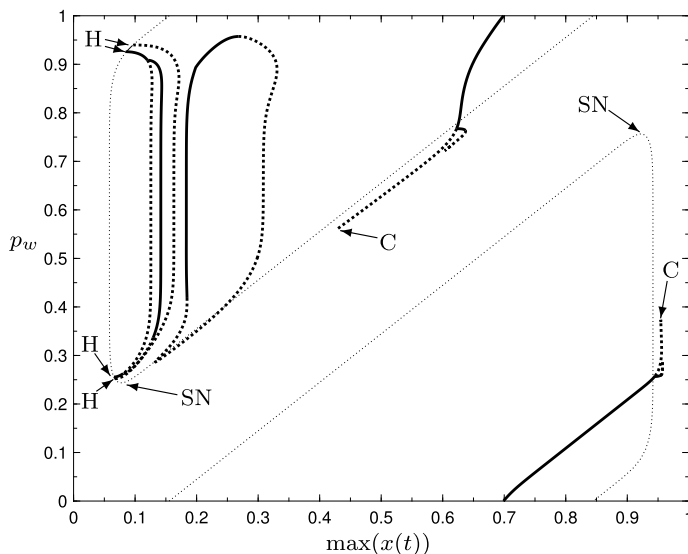
Figure 5. Bifurcation diagram in parameter p_w. Solid and dotted curves represent stable and unstable solutions, respectively. Thin and thick curves represent equilibria and periodic orbits, respectively. Hopf, homoclinic, and saddle-node bifurcations of equilibria are labeled H, C, and SN. Note that we plot the bifurcation parameter on the vertical axis, so as to facilitate comparison with Figure 3.

The bifurcation diagram reveals that for most values of p_w, there exist four equilibria (between the saddle nodes at $p_w \approx 0.75$ and $p_w \approx 0.25$). For almost all values of p_w, these equilibria are unstable. It is only for two very small parameter ranges near $p_w \approx 0.25$ and $p_w \approx 0.95$ that we can find stable equilibria, book-ended by Hopf bifurcations—only the second of these two branches is large enough to be seen in Figure 5. In the context of the robot's dynamics, these equilibria represent positions along the slopes of the peaks shown in the sensor function in Figure 1, where if the robot were stationary at one of these positions, and the recent excitation of its neuron were zero ($y_{t \in [t-\theta, t]} = 0$), the robot would remain at that position indefinitely (or until perturbed). The positions of the equilibria must have $y = 0$, and hence occur at a fixed input height, specifically $I = \beta / \psi$ (by solving Equations 3 and 4 for $d/dt = 0$). For both small and large values of p_w, the two peaks merge into a single peak. When this happens, there are only two locations in the environment with the particular sensor excitation associated with this unstable equilibrium, and this corresponds to the the disappearance of two equilibria in saddle-node bifurcations when $p_w | p_n \lesssim 0.25$.

The fact that the equilibria are almost always unstable implies that the dynamics of the robot will almost always be oscillatory. This agrees with the observations in Figure 4, where the various behaviors observed are indeed oscillatory. Again, we can develop an understanding of the origin of these oscillatory dynamical solutions by examining the bifurcations for different placements of the distractor wide peak (p_w). The most clearly relevant periodic solutions are those that produce high-fitness behaviors and that have relatively large basins of attraction. Other solutions seem, at least at first glance, to be less relevant, either because they are unstable, or because they only arise for a very small set of initial conditions. However, tracking unstable solutions can provide insight, as they may be associated with the disappearance of stable solutions (e.g., in a saddle-node bifurcation) or with the reduction of another attractor's basin of attraction, or they may serve as an organizing center for other important dynamics.

Let us briefly describe the solutions and bifurcations we observe before connecting these abstractions to the robot's behavior. Figure 6 provides enlargements of the bifurcation diagram depicted in Figure 5. It shows the locations of homoclinic (C), period-doubling (P), torus (T), and saddle-node (SN) bifurcations of periodic orbits. Panel A illustrates that a branch of periodic orbits emerges from two Hopf bifurcations and is unstable. These periodic orbits are unstable, and so the Hopf bifurcations
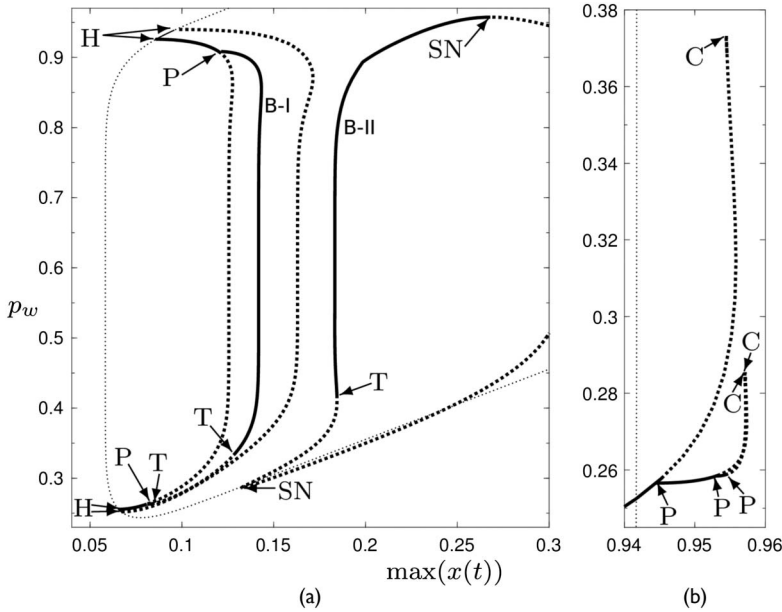
Figure 6. Enlargements of the bifurcation diagram shown in Figure 5. Line types are as in Figure 5. Homoclinic (C), period-doubling (P), torus (T), and saddle-node (SN) bifurcations of periodic orbits are indicated.

are subcritical. A second branch of stable periodic solutions emerges from two other supercritical Hopf bifurcations. However, these periodic orbits soon lose their stability at period-doubling bifurcations, producing another branch of periodic orbits. Part of this new branch of periodic orbits loses stability between two torus bifurcations. We also observe a branch of periodic orbits with larger maxima in x for a range of p_w values. This branch of periodic orbits is bounded by two saddle-node bifurcations (which are different from the saddle-node bifurcations that underlie the unstable equilibria mentioned above). The periodic orbits along the stable segment of this branch change in stability at a torus bifurcation at about $p_w \approx 0.4$. Panel B shows another branch of stable solutions that loses stability in a cascade of period-doubling bifurcations. The resulting unstable periodic orbits then each terminate at homoclinic bifurcations.

What role do these bifurcations play in determining the success of the robot's mission to find the narrow peak? To start our approach to this question, we can compare the $\max(x|p_n)$ values depicted in the fitness survey of Figure 3 and in the bifurcation diagrams just described. First we observe the large area of uniform high-fitness performance that covers most of the center of Figure 3. Throughout this area, the maximum distance of the robot from the target narrow peak in the final $10s$ of the simulation is $\max(x_0|p_n) \approx 0.142$. Looking at the bifurcation diagram, we see a branch of stable periodic orbits with the same $\max(x)$. We have labeled this branch B-I in Figure 6. Figure 7 superimposes the bifurcation diagram on top of the fitness-survey data from Figure 3, showing that B-I does indeed exist for the relevant values of p_w, and thus further supporting our association of this stable periodic orbit with the majority of successful behaviors. Returning now to Figure 4, we see that the two initial conditions that lie within this region (trajectories (b) and (c)) do indeed fall into what appears to be the same attractor.

A second region of high fitness is also found in Figure 7. This area (which includes trajectory (a)) is slightly lower in fitness than that just described, and involves an oscillation of different form than those of the higher-fitness region (compare the right column for trajectories (a), (b), and (c) in Figure 4). This area has a large section where $\max(x_0|p_n) \approx 0.183$, which nicely matches the branch of stable periodic solutions labeled B-II in Figure 6. A torus bifurcation just above $p_w = 0.4$ is associated with this region becoming less and less fit as p_w decreases. Note that continuation analysis does not pick up quasi-periodic or chaotic solutions.
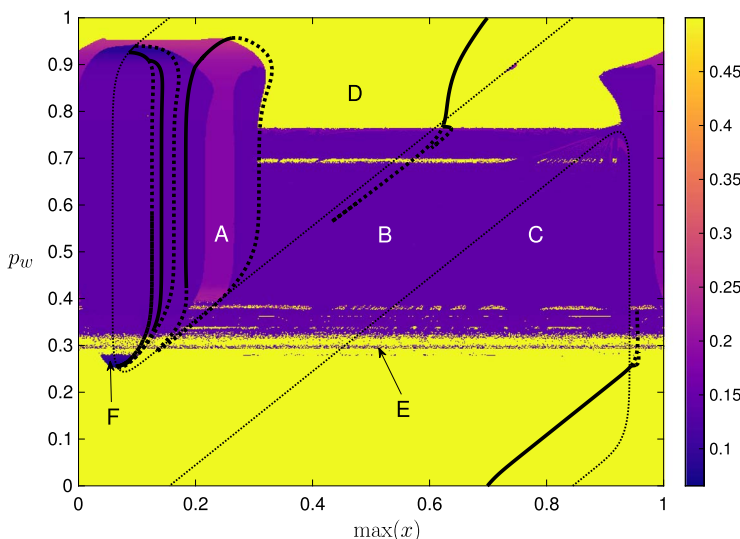
Figure 7. Bifurcation diagram in p_w superimposed on the initial conditions survey for $\theta = 1.140$. The color scheme represents an evaluation of the robot's performance with purple (yellow) representing more (less) successful behaviors. Line types are as in Figure 5. The letters A–F correspond to the initial conditions for the trajectories (a)–(f) shown in Figure 4.

Turning now to the examples of poor performance, trajectory D presents a large-amplitude oscillation that belongs to one of two branches of periodic solutions that originate from homoclinic bifurcations, labeled C (to be precise, trajectory D approaches the upper branch), in Figure 5. The ranges of p_w values where these branches are stable agree well with the yellow initial conditions in Figure 7. Trajectory E is an example of a chaotic attractor. The robot initially appears to approach a small-amplitude torus that is created in the torus bifurcation shown in Figure 6(a) at $(\max(x), p_w) \approx (0.08, 0.26)$. However, the trajectory in Figure 4(e) then undergoes irregular excursions throughout phase space, resulting in large-amplitude oscillations in x. The final example in Figure 4(f) demonstrates a successful trial, where the robot begins close to the Hopf bifurcation. Eventually, the robot settles into aperiodic motion, in agreement with the attractor being born at the torus bifurcation near $p_w \approx 0.26$. In fact, this attractor appears to also be chaotic, because changes to the initial condition of x on the order of 10^{-9} lead to a set of different trajectories. Figures 4(e) and (f), therefore, demonstrate that chaotic attractors can be both successful and unsuccessful for the robot.

The scattered pattern of yellow and purple points in Figure 7 in the region $0.28 \lesssim p_w \lesssim 0.4$ and near 0.7 is evidence of the sensitivity to initial conditions associated with chaotic behavior. These regions of the parameter p_w correspond well to the period-doubling cascades seen on the branches of periodic orbits for large $\max(x)$, shown in detail in Figure 6(b), which represents a well-known route to chaos [22]. Furthermore, one could expect to see chaotic behavior resulting in between the torus bifurcations shown in Figure 6(a). It is likely that different resonances that result from dynamics on tori interact to create chaotic attractors; for example, see Keane et al. [17].

To summarize the analysis this far, two branches of stable periodic orbits are responsible for the successful behaviors demonstrated by the robot in a wide range of initial and environmental conditions. The highest performance behavior is associated with branch B-I, and a slightly less effective behavior with a smaller basin of attraction (at least in terms of the conditions explored in Figure 3) is associated with a second branch of stable periodic orbits that we have labeled B-II. It is notable that if this system did not include a delayed term, we would find fewer and less diverse bifurcations. We see here a typical case of delayed systems involving a degree of dynamical complexity that is not always present in systems of ODEs. This complexity is playing a role in producing the high-fitness behavior, but beyond this vague assertion, the way that delay is contributing remains unclear. To
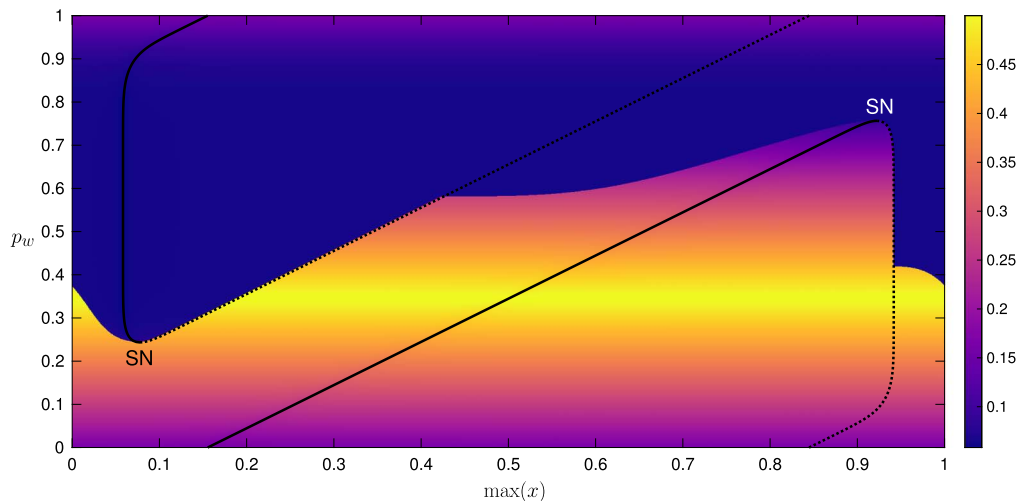
Figure 8. Bifurcation diagram in p_w superimposed on the initial-condition survey for $\theta = 0$. The color scheme is as in Figure 7. Line types are as in Figure 5.

address this and try to better understand the role that the delay is playing, our next phase of analysis evaluates the parameter θ, which describes the amount of delay in the neuron's recurrent connection.

3.2   Analysis of the Effect of the Delay Time, θ

We now consider the effect of varying the time delay parameter, θ. We consider all other parameters to be fixed at the values given in Table 1. As an initial study, we consider the dynamics of the system when there is no delay, that is, we set $\theta = 0$. We then consider increasing the delay from zero to the value found in the evolution, and slightly beyond, to analyze how this changes the dynamics.

Figure 8 shows the bifurcation diagram in p_w with $\theta = 0$, superimposed on an initial-condition performance survey for $\theta = 0$. We can see that the system admits either two or four equilibria, depending on the position p_w of the wide peak. Indeed, the locations of the equilibria are exactly the same as seen in Figure 5, precisely because the locations of the equilibria are independent of θ. Again, the equilibria appear and disappear in saddle-node bifurcations, but unlike in the delayed system, the emerging solutions always include one stable equilibrium. This stable equilibrium corresponds with a high-fitness score for values of p_w between about 0.25 and 0.9, when it lies at approximately $x = 0.06$. The dark purple shading in Figure 8 shows the initial conditions that converge to this *good* equilibrium. The remaining initial conditions converge to the *bad* equilibrium, which is generally far from the narrow peak. Comparing Figures 8 and 3 and recalling that the genetic algorithm focused on improving the robot's performance within the range $p_w \in [0.25, 0.75]$, it is clear that (as we might expect) the system with zero delay does not perform as well.

We now consider the effects of increasing θ from zero. We now have two parameters to consider (p_w and θ), and we first fix p_w while varying θ. Figure 9 provides a first glimpse of the effect of varying θ. The top panel in this figure presents two different initial conditions (green and red) with $\theta = 0$. The green trajectory converges to the good stable solution, and the red trajectory converges to the bad one. Crudely summarized, the robot appears to stop at the first peak that it encounters. The next two panels show trajectories for the same initial conditions and value of p_w, but with different values of θ. For $\theta = 0.7$ we can see that the good solution is unchanged, but the bad solution has now become oscillatory. As θ is increased further to the value selected by evolution, the red trajectory actually becomes a high-performance solution that moves to and remains close
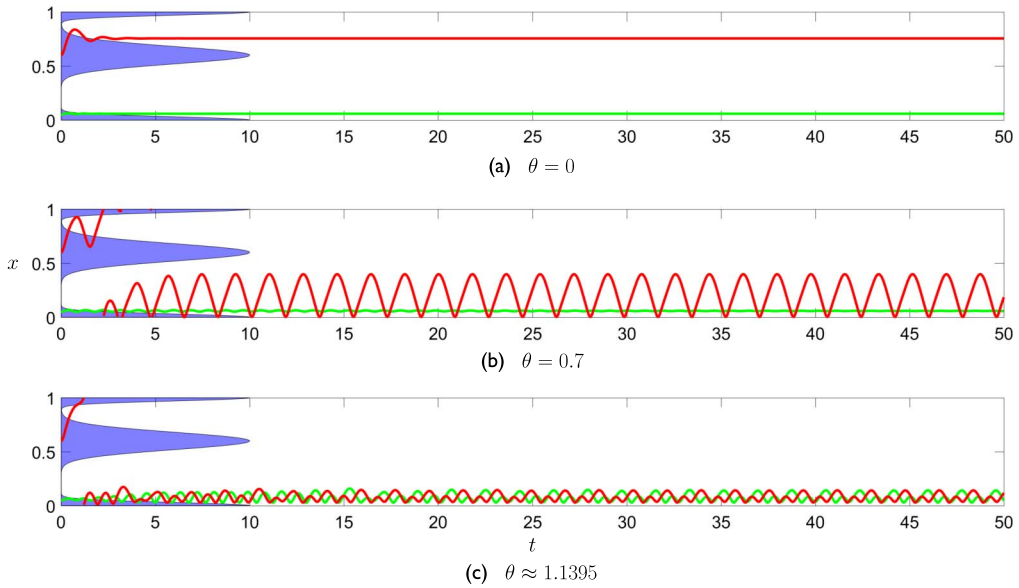
Figure 9. Time series for fixed p_w = 0.6 and (a) $\theta = 0$, (b) 0.7, and (c) 1.140. The plot shows the position of the robot (horizontal axis) as a function of time (vertical axis). The green and red trajectories have constant-valued initial conditions $(x_0, y_{t \in [-\theta, 0]}) = (0.05, 0)$ and $(0.6, 0)$, respectively. The robot's environment on an $(l(x), x)$ plane is plotted in blue in the background.

to the narrow peak. In fact the two solutions in this case have converged to the same attractor. As a tradeoff, the green solution has also started to oscillate, and thus is now less fit than the green solution for $\theta = 0$.

This behavior can be seen in more detail in Figure 10, which shows a bifurcation diagram for fixed p_w, varying θ. The equilibrium solutions are now represented by vertical lines, because their positions in x do not change as the delay θ is varied (recall the discussion above, where we observed that these equilibria are independent of θ). We observe that two of the four equilibria are always unstable. The other two change their stability through Hopf bifurcations at various values of θ. The
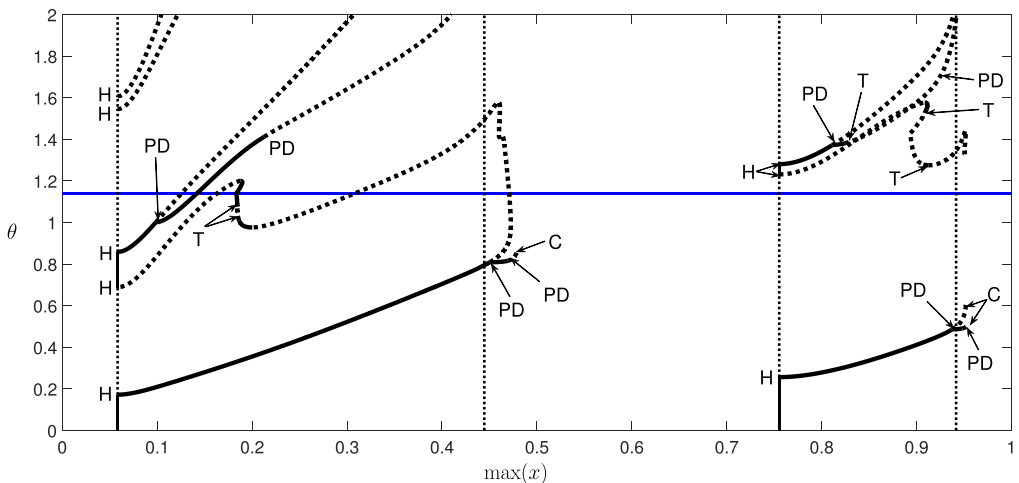

Figure 10. Bifurcation diagram for varying θ and fixed p_w = 0.6. Line types are as in Figure 5. The horizontal blue line corresponds to the θ value of 1.140.
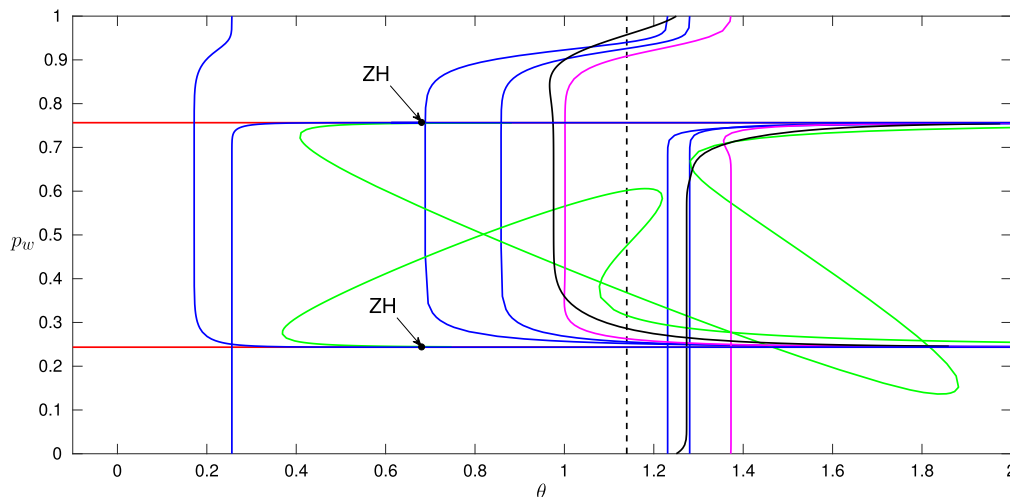
Figure 11. Bifurcation set in the (θ, p_w) plane, including curves of (red) saddle node of equilibria, (blue) Hopf, (green) homoclinic, (black) saddle node of periodic orbits, and (magenta) period-doubling bifurcations. Codimension-2 zero-Hopf bifurcations are labeled ZH. The dashed vertical line indicates the θ value selected by evolution, $\theta = 1.140$.

Hopf bifurcations produce branches of periodic orbits that generally increase in amplitude as θ increases before undergoing torus, period-doubling, and homoclinic bifurcations.

A high-fitness parameter value would likely have a small-amplitude stable solution and no stable large-amplitude solutions. In agreement with the example time series shown in Figure 9, the evolution algorithm appears to have found a value of θ where for $p_w = 0.6$ only small-amplitude stable solutions exist. We see that all initial conditions in x will converge to a small-amplitude attractor. For this particular value of p_w we can see that near $\theta = 0.9$ in Figure 10, there appear to be no large-amplitude solutions and only solutions that are even smaller than those at the evolved value of θ. This might appear to be a more optimal choice of θ, but in fact, near $\theta = 0.9$ there exist complex and chaotic solutions that regularly cross the periodic boundary in x and are therefore bad solutions. These are not shown in Figure 10 (because of the difficulty in assigning a max(x) for these solutions as they wrap around the periodic environment), but result from the period-doubling cascade near $\theta \approx 0.8$, max(x) ≈ 0.5. Moreover, we remind the reader that this is only a bifurcation diagram for fixed $p_w = 0.6$ and that the evolved solution was required to produce successful behavior for a wide range of p_w values.

Finally, we can show the effects of varying both θ and p_w by showing a *bifurcation set* in Figure 11. A bifurcation set shows curves of bifurcations in a two-parameter space, so it can indicate how and where different types of solutions arise in parameter space. However, the actual solutions, or any measures of them, are not shown, so it is harder to assess the "fitness" of the solutions than in a bifurcation diagram.

The bifurcation set in Figure 11 includes curves of (red) saddle-node bifurcations of equilibria, (blue) Hopf bifurcations, (green) homoclinic bifurcations, (black) saddle-node bifurcations of periodic orbits, and (magenta) period-doubling bifurcations. The curves of saddle-node bifurcations of equilibria are horizontal lines in this projection because their location in p_w is independent of θ. This is because their location is a function of the positions of the equilibria, which again (recall) is independent of θ. These curves meet Hopf bifurcation curves at zero-Hopf bifurcations, labeled ZH. As θ is increased from 0, we again see a succession of Hopf bifurcations (blue curves), creating periodic solutions. These periodic solutions change stability as period-doubling (magenta) or torus bifurcations (not shown) are crossed. The dashed vertical line indicates the θ value 1.140. The bifurcation set presented here is not exhaustive; for example, period-doubling cascades are not

shown. Nonetheless, it gives a clear impression of the complexity of the delayed system. The bifurcation set also reveals that the behavior of the robot is robust to small perturbations of θ. One could imagine shifting the vertical dashed line in Figure 11 slightly to the right or left and observing that the same bifurcations occur (albeit at slightly different values of p_w), and hence a similar set of solutions would occur, again at slightly different values of p_w, and with slightly different x-values.

The bifurcations in Figure 6 are those curves in Figure 11 that intersect the vertical line at $\theta = 1.14$. Going down that vertical line from the top, first you get the saddle-node bifurcation (black) that is at about $\max(x) = 0.27$ in Figure 6, and is part of branch B-II. Then there are the two Hopf bifurcations (blue) from the (leftmost) equilibrium. Below these is a period-doubling bifurcation (pink), which creates branch B-I. The branch that destabilizes the bad equilibrium trajectories (exemplified in the top panel of Figure 9) is the second-from-the-left blue Hopf curve (at $\theta \approx 0.25$). The destabilization of the periodic solutions (second panel) looks like it happens in a period-doubling bifurcation that is not drawn in Figure 10.

4   Discussion

We have presented an extended dynamical analysis of a robot controller whose parameters were optimized by an evolutionary algorithm to perform a simple embodied discrimination task. The purpose of this study was to investigate how delayed signals might play a functional role within a sensorimotor loop that accomplishes a discrimination task. To that end, the evolved controller was constrained so that the only way that it could succeed would be by taking advantage of its delayed recurrent connection, as this was the only aspect of the controller with sufficient potential dynamical complexity to solve the task at hand. (In a preliminary study, we found that when θ was constrained to be zero, evolution was unable to find a parameterization that performed well on this problem, supporting our claim that the delay is a necessary component of the high-performance behavior. The results of this preliminary study are not presented in this article.) Within these constraints, the task and controller were designed to be as simple as possible. The controller was modeled as a single neuron with a recurrent connection, and the task to solve involved a robot embedded in a 1D environment that has to distinguish between a narrow target stimulus peak and a wide distractor peak. Despite this minimalism, the evolved system demonstrated diverse high- and low-performance behaviors.

The top-performing solution found in all of the 32 evolutionary runs demonstrated relatively complex dynamics compared to ODE-based systems that lack delay. To understand these dynamics and their relation to delay, we conducted a dynamical analysis. The first stage of this analysis exposed the diverse bifurcations and solutions that underlie the demonstrated behaviors, and allowed us to identify which branches of solutions were associated with the behaviors that contributed most to fitness. The majority of high-performing behaviors of the evolved agent were oscillatory, with only a relatively small set of parameters and initial conditions producing high-performing equilibrium-based behaviors. Oscillations are common when delay is present, but engineers often prefer to avoid oscillations and use architectures that approach one or more equilibria. It would be interesting to see if further constraints imposed upon the system could produce high-performance equilibria rather than oscillations. It would also be interesting to consider basic biological sensorimotor behaviors in this light—do they tend to involve (perhaps low-amplitude) oscillations, or are they better described as equilibria?

We also observed a range of conditions that produced chaotic dynamics. Some of the trajectories in this region were high-fitness, and others were low-fitness—trajectories (e) and (d) in Figure 4 provide examples of each. It is not uncommon for chaotic dynamics to appear in systems with delay, and it is interesting to consider this, both in the context of engineering (chaos is generally something robotics engineers try to avoid) and in the context of biology, where finite neural conduction velocity means that delay is unavoidable.

The second stage of our dynamical analysis focused upon θ, the amount of delay in the recurrent connection. When $\theta = 0$ (i.e., when there was no delay in the recurrent connection), the system's
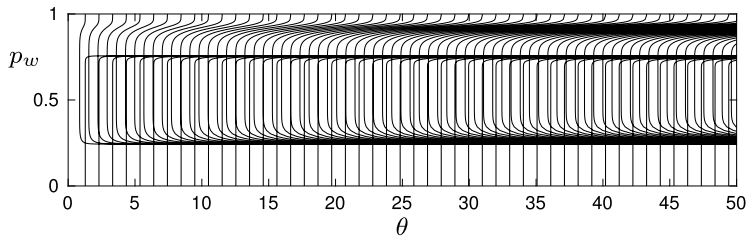
Figure 12. Hopf bifurcation curves shown in Figure 11 for a larger range of θ. These curves demonstrate the repeating nature of periodic solutions in the delay system.

dynamics were relatively simple and incapable of solving the discrimination task; the robot essentially moved with a positive velocity until it encountered a peak, where it would remain (an equilibrium solution) regardless of whether that peak was the target narrow stimulus or the distractor wide stimulus. For larger values of θ, the equilibrium associated with the distractor peak becomes unstable. So too does the equilibrium associated with the target peak, but as θ increases, bifurcations produce stable periodic orbits that cause the robot to oscillate relatively close to the target peak. The basins of attraction of these periodic solutions include the majority of initial conditions tested during evolution (the starting point of the robot and the placement of the distractor peak), and so, in most tested cases, the robot avoids the distractor peak and moves toward the target peak as encouraged by the fitness function. The evolved value of θ appears to be a kind of compromise where the robot does not approach as close to the target peak as the equilibrium that exists when $\theta = 0$, but settles where the robot is more capable of distinguishing between the two peaks, avoiding the distractor and moving toward the target.

Evolution tuned all of the controller's parameters simultaneously, so evaluating how the system changes when we vary θ on its own is perhaps a slightly artificial perspective, but one that nevertheless provides some insight into what evolution (or other adaptive processes such as development and learning) might be able to take advantage of (i.e., work with) when modulating signal delay in nervous systems. Specifically, as delay increases, the number of solutions and bifurcations in the system increases. This is a well-established property of delayed systems. Yanchuk and Perlikowski [31] show that as the delay time increases, so too does the number of coexisting solutions. This is due to the fact that any finite-period solution that exists for a certain delay time will reappear for infinitely many delay times. The same effect can give rise to repeating patterns of bifurcations as delay increases, as demonstrated in systems of delay-coupled Hodgkin-Huxley neurons [16] and of pulsing lasers with delayed self-feedback [26]. As discussed in [30], the delay time can be changed to directly influence the dynamical complexity of a delay system. Increasing delay means increasing the dimensionality of unstable manifolds and chaotic attractors, leading to complex dynamics. To give an example that relates this idea to the model presented above, Figure 12 shows curves of Hopf bifurcations (for the system examined in this article) in the (p_w, θ) plane for $\theta \leq 50$. Every time a Hopf curve is crossed as the delay is increased, an additional family of periodic orbits is introduced into phase space (though keep in mind that many of the added solutions will be unstable).

One of the overarching goals of this research was to broaden our understanding of how delay can serve a functional role in control architectures. In the introduction we described one example of such functionality: The synchronization of signals traveling from the brain to limbs in octopuses. The destabilization of equilibria by delay and the production of new stable and unstable invariant sets described above provide examples of other ways that delay can be used by evolved controllers to produce desired dynamics. In addition, delay has wide-ranging diverse effects upon the observed dynamics. This is rather different from the parameters of conventional CTRNN-based evolutionary robotics, where any one parameter can *in a particular context* cause one or a few of these kinds of qualitative change, but in general is not capable of producing such a wide variety of changes. When optimizing a system, it can be useful to have a handle that can not just effect small change, but also

qualitative change. Delay appears to be such a channel, and, as discussed in the introduction, neuroscience evidence strongly suggests that the delay of neurons varies over a wide range of time scales and is modulated by poorly understood, but presumably adaptive, mechanisms. Could the delays in nervous systems and their modulation play just as important a role in influencing the nervous system's (and broader sensorimotor/behavioral) dynamics as the topology (i.e., the connectome) of the brain itself?

Scientific investigation and modeling always involve a tradeoff between detailed realism and comprehensibility via simplification or abstraction. The dynamical hypothesis in the cognitive sciences has made important advances in recognizing the dynamical natures of natural cognitive systems (and the differences between these and computational devices). These advances have been accomplished without including delay, despite its ubiquitous presence throughout nervous tissues. Is it time to include delay in evolutionary robotics investigations? We have shown here that at least it is possible to do so—the methods exist for investigating delay in situated, embodied, and dynamical robots. It certainly seems that there is much more that can be learned through studies such as that just presented. Reiterating content presented in the introduction, we know of types of functional delay that synchronize signals or cancel out the effect of diverse neuron lengths. It seems highly unlikely that this is the only way that signal propagation delay is used functionally in natural systems. Evolutionary robotics methods seem like an excellent way to expand our library of ways that functional delay might be used advantageously.

Minimalistic evolutionary robotics studies such as this one, where we try to create as simple as possible a model, can sometimes end up highlighting the immense complexity of biological nervous systems. The human nervous system is already stupefying in scale, with approximately 100 billion neurons and approximately 7000 times as many synapses. Recognizing the significant delays imposed by the diverse conduction velocities and varied-length neurons further complicates this picture. At some level it is amazing that such a complex system does anything coherent at all. Recognizing the asynchronous, spatially and temporally extended dynamics of the brain also raises interesting phenomenological questions regarding the perceived temporal unity of experience. As mentioned above, engineers avoid delayed influence whenever possible, as it makes systems difficult to understand, predict, and control. Evolution has different constraints than those of engineers in that it does not need to be understood by anyone, it just needs to work [27]. Real tissues and biological sensors and motors have non-negligible delay. How are organisms organized so as to compensate for, or take advantage of, the complexity inherent in delayed systems? We have presented above our first steps toward approaching this question.

References

1. Beer, R. D. (1995). On the dynamics of small continuous-time recurrent neural networks. *Adaptive Behavior*, *3*(4), 469–509.

2. Beer, R. D. (1997). The dynamics of adaptive behavior: A research program. *Robotics and Autonomous Systems*, *20*, 257–289.

3. Beer, R. D. (2003). The dynamics of active categorical perception in an evolved model agent. *Adaptive Behavior*, *11*(4), 209–243.

4. Beer, R. D. (2014). Dynamical systems and embedded cognition. In K. Frankish & W. Ramsey (Eds.), *The Cambridge handbook of artificial intelligence* (pp. 128–148). Cambridge, UK: Cambridge University Press.

5. Brooks, R. A. (1990). Elephants don't play chess. *Robotics and Autonomous Systems*, *6*(1–2), 3–15.

6. Chklovskii, D. B., Schikorski, T., & Stevens, C. F. (2002). Wiring optimization in cortical circuits. *Neuron*, *34*(3), 341–347.

7. Egbert, M., Keane, A., & Postlethwaite, C. (2017). Lag in situated, embodied and dynamical adaptive systems. In H. Fellerman, J. Bacardit, A. Goni-Moreno, & R. M. Füchslin (Eds.), *Proceedings of the 14th European Conference on Artificial Life, ECAL 2017* (pp. 130–137). Cambridge, MA: MIT Press.

8. Engelborghs, K., Luzyanina, T., & Samaey, G. (2000). DDE-BIFTOOL: A Matlab package for bifurcation analysis of delay differential equations. *TW Report*, 305.

9. Fields, R. D. (2008). White matter in learning, cognition and psychiatric disorders. *Trends in Neurosciences*, *31*(7), 361–370.

10. Flunkert, V., & Schöll, E. (2009). *pydelay: A python tool for solving delay differential equations. arXiv:0911.1633.*

11. Ghil, M., Zaliapin, I., & Thompson, S. (2008). A delay differential model of ENSO variability: Parametric instability and the distribution of extremes. *Nonlinear Processes in Geophysics*, *15*(3), 417–433.

12. Hebb, D. O. (1949). *The organization of behavior: A neuropsychological theory*. New York: Wiley. Google Books ID: yReB9eGP2pwC.

13. Izhikevich, E. M. (2006). Polychronization: Computation with spikes. *Neural Computation*, *18*(2), 245–282.

14. Izquierdo, E., Harvey, I., & Beer, R. D. (2008). Associative learning on a continuum in evolved dynamical neural networks. *Adaptive Behavior*, *16*(6), 361–384.

15. Izquierdo, E. J., & Lockery, S. R. (2010). Evolution and analysis of minimal neural circuits for klinotaxis in Caenorhabditis elegans. *The Journal of Neuroscience*, *30*(39), 12908–12917.

16. Kantner, M., & Yanchuk, S. (2013). Bifurcation analysis of delay-induced patterns in a ring of Hodgkin–Huxley neurons. *Philosophical Transactions of the Royal Society A*, *371*(1999), 20120470.

17. Keane, A., Krauskopf, B., & Postlethwaite, C. M. (2016). Investigating irregular behavior in a model for the El Niño southern oscillation with positive and negative delayed feedback. *SIAM Journal on Applied Dynamical Systems*, *15*(3), 1656–1689.

18. Lang, R., & Kobayashi, K. (1980). External optical feedback effects on semiconductor injection laser properties. *IEEE Journal of Quantum Electronics*, *16*(3), 347–355.

19. Nilsson, N. J. (1984). Shakey the Robot (Technical Report). Menlo Park, CA: SRI International.

20. Pumphrey, R. J., & Young, J. Z. (1938). The rates of conduction of nerve fibres of various diameters in cephalopods. *Journal of Experimental Biology*, *15*(4), 453–466.

21. Sieber, J., Engelborghs, K., Luzyanina, T., Samaey, G., & Roose, D. (2014). DDE-BIFTOOL manual: Bifurcation analysis of delay differential equations. *arXiv preprint arXiv:1406.7144*.

22. Strogatz, S. H. (2014). *Nonlinear dynamics and chaos: With applications to physics, biology, chemistry, and engineering*. Boulder, CO: Westview Press.

23. Swadlow, H. A. (1982). Impulse conduction in the mammalian brain: Physiological properties of individual axons monitored for several months. *Science*, *218*(4575), 911–913.

24. Swadlow, H. A., & Waxman, S. G. (2012). Axonal conduction delays. *Scholarpedia*, *7*(6), 1451 revision #125736.

25. Taylor, M. M. (1973). The problem of stimulus structure in the behavioural theory of perception. *South African Journal of Psychology*, *3*, 23–45.

26. Terrien, S., Krauskopf, B., Broderick, N. G., Jaurigue, L., & Lüdge, K. (2018). Q-switched pulsing lasers subject to delayed feedback: A model comparison. *Physical Review A*, *98*(4), 043819.

27. Thompson, A., Layzell, P., & Zebulum, R. (1999). Explorations in design space: Unconventional electronics design through artificial evolution. *IEEE Transactions on Evolutionary Computation*, *3*(3), 167–196.

28. van Gelder, T. (1998). The dynamical hypothesis in cognitive science. *Behavioral and Brain Sciences*, *21*(05), 615–628.

29. Waxman, S. G. (1980). Determinants of conduction velocity in myelinated nerve fibers. *Muscle & Nerve*, *3*(2), 141–150.

30. Wolfrum, M., Yanchuk, S., Hövel, P., & Schöll, E. (2010). Complex dynamics in delay-differential equations with large delay. *The European Physical Journal Special Topics*, *191*(1), 91–103.

31. Yanchuk, S., & Perlikowski, P. (2009). Delay and periodicity. *Physical Review E*, *79*(4), 046221.