

Analysis of Ultrastability in Small Dynamical Recurrent Neural Networks

Eduardo J. Izquierdo¹, Miguel Aguilera², Randall D. Beer¹

¹ Indiana University Bloomington, IN, U.S.

² Universidad de Zaragoza, Spain
edizque@indiana.edu

Abstract

This paper reconsiders Ashby's framework of adaptation within the context of dynamical neural networks. Agents are evolved to behave as an ultrastable dynamical system, without imposing a priori the nature of the behavior-changing mechanisms, or the strategy to explore the space of possible dynamics in the system. We analyze the resulting networks using dynamical systems theory for some of the simplest conditions. The picture that emerges from our analysis generalizes the idea of ultrastable mechanisms.

Introduction

Organisms change their behavior in an attempt to remain adapted to their interaction with the environment, in order to maintain their self-constitution. W. Ross Ashby developed a theoretical framework to understand operationally this tendency towards ultrastability in organisms (Ashby, 1960). His approach considered the role of what can be interpreted as proxies to the essential variables of the organism - variables that correlate with its physiological well-being and that are directly available to the organism (e.g., body temperature, sugar level, oxygen intake). A behavior is ultrastable if it conserves the proxies to the essential variables of the organism within some physiological limits under changing conditions. This paper explores the nature of the dynamics of ultrastable systems in artificially evolved neural networks.

Ashby's framework of ultrastability is still central to much of the work on adaptive behavior, but many of the details of the implementation in neural networks have not been studied in enough depth (Harvey, 2008). The mechanism of adaptation Ashby studied was hand-designed specifically to find stable agent-environment interactions during perturbations. There are two aspects of his original framework that we seek to re-examine. First, the process of adaptation depends on a strict separation between the parameter-changing mechanisms and the behavior-producing mechanisms. In doing so, it relies heavily on the stochastic, switch-like nature of the behavior-changing mechanisms. Second, the framework of adaptation treats the projection of the critical boundary on the space of internal dynamics of the agent as a function only of the constitution of the agent.

In this paper we ask: what are the agent-environment conditions that allow artificial evolution to find neural networks with ultrastable properties? And, what are some of the ways in which ultrastability specified at the behavioral level can be implemented dynamically at the neural network level? We are interested in how a dynamical system can explore the richness of its state space, navigating through the different regions of its phase-portrait. The goal of this paper is to analyze neural networks evolved to be ultrastable in an attempt to extend and generalize Ashby's framework of adaptation.

Ashby's Ultrastable Mechanism

Ashby based his framework around the basic assumption that organisms change their behavior by learning, so that the later behavior is better adapted to their environment than the earlier. He distinguished between two kinds of nervous system activity: hardwired reflex and learned behavior. In his framework (Figure 1A), the agent (\mathcal{A}) is defined by two subsystems: \mathcal{R} , responsible for the ongoing reflexive behavior; and \mathcal{S} , responsible for regulating the behavior of \mathcal{R} . There are also a set of variables, \mathcal{P} , that serve as proxies to the organism's essential variables (\mathcal{V}). The environment (\mathcal{E}) is defined as a system whose variables affect the organism through coupling and which are in turn affected by it.

For the organism to survive, its essential variables (\mathcal{V}) must be kept within viable limits (gray area, Figure 1B). The 'constitutive viable region' maps to an 'internal viable region,' $\mathcal{M}(\mathcal{V})$ (gray areas, Figure 1C). When the agent's internal dynamics are inside this region, the essential variables are kept within viable limits. When the internal dynamics are outside this region, the agent-environment dynamics have an effect on the proxies to the essential variables of the system. When the proxies are outside the appropriate ranges for the organism, they introduce parametric changes in \mathcal{S} . The configuration of \mathcal{S} influences the agent-environment interaction through \mathcal{R} . Ashby proposed a step-function as a mechanism for \mathcal{S} . In Ashby's view, adaptation occurs as the system randomly flips through the catalog of \mathcal{R} dynamics, until it stumbles across one that is stable for the current environment.

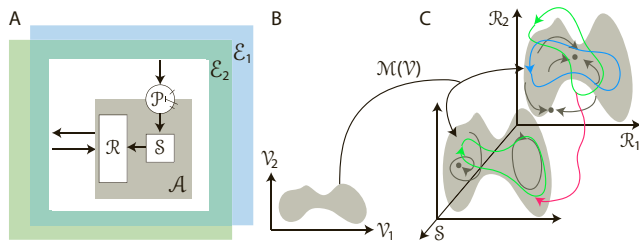


Figure 1: Ultrastable system (adapted from Ashby, 1960). [A] Architecture: Environments (\mathcal{E}_i); agent (\mathcal{A}); reactive system (\mathcal{R}); system in charge of learning (\mathcal{S}); and proxy for the essential variables (\mathcal{P}). [B] Region of viability (gray area) over the agent’s essential variables (\mathcal{V}_i), defined by the agent’s constitution. [C] Agent’s state space (\mathcal{R} and \mathcal{S}). Gray attractors and limit cycles depict autonomous dynamics of \mathcal{R} . Blue and green trajectories depict dynamics of the system when coupled to environments \mathcal{E}_1 and \mathcal{E}_2 , respectively. Gray areas depict the internal viable regions, according to $\mathcal{M}(\mathcal{V})$. The magenta trajectory represents the state of the system during adaptation.

Behaviorally Ultrastable Dynamical System

This paper extends the framework of ultrastability by reconsidering it from a behavioral level. There are two aspects of Ashby’s framework that we re-examine.

First, in Ashby’s mechanism and the more recent simulation models that have stemmed from his work (Di Paolo, 2000; Iizuka and Di Paolo, 2008; Herrmann et al., 2004), there is a strict division between the parts of the system that continuously interact with the environment, and the parts that change the internal organization of the system itself. As with previous work on modeling learning without synaptic plasticity (Yamauchi and Beer, 1994; Izquierdo et al., 2008), we consider a broader definition of an ultrastable system by removing the *a priori* distinction between the reactive and the learning components within the agent (Figure 2A). This allows us to explore some of the ways in which ultrastability specified at the behavioral level can be implemented at the neural network level.

Second, in Ashby’s framework, the internal viable region (gray areas, Figure 1C) is a function $\mathcal{M}(\mathcal{V})$ solely of the constitutive viable region (gray area, Figure 1B). When coupled, the agent-environment produce a dynamic which is specific to that environment (blue trajectory, Figure 1C). A change of environment modifies the coupled dynamics (green trajectory in the back, Figure 1C), but not the internal viable region. Therefore, a change in the agent-environment interaction can drive the internal dynamics of the agent outside its internal viable region, triggering changes in \mathcal{S} (magenta trajectory, Figure 1C), until a new coupled dynamic is found (green trajectory in the front, Figure 1C). This perspective is problematic because it assumes the internal dynamics of the agent must be within the same internal vi-

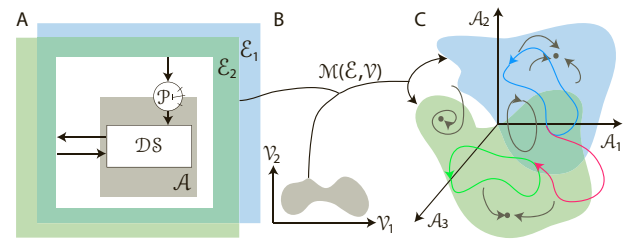


Figure 2: Extended ultrastable system. [A] Architecture: Environments (\mathcal{E}_i); agent (\mathcal{A}); internal dynamics (\mathcal{DS}); and proxy for the essential variables (\mathcal{P}). [B] Region of viability (gray area) over the agent’s essential variables (\mathcal{V}_i). [C] Agent’s state space (\mathcal{A}). Gray attractors and limit cycles depict the autonomous dynamics of \mathcal{A} . Blue and green trajectories depict the dynamics of the system when coupled to environments \mathcal{E}_1 and \mathcal{E}_2 , respectively. Colored volumes represent the internal viable regions, $\mathcal{M}(\mathcal{E}, \mathcal{V})$, colored according to the environment. Magenta trajectory represents the state of the system during adaptation.

able region for the two different behaviors. We consider a broader condition (Figure 2), where the internal viable region is a function of both the constitutive viable region and the environment, $\mathcal{M}(\mathcal{E}, \mathcal{V})$. Like in Ashby’s case, we assume the constitutive viable region is defined over the essential variables by the organization of the organism (Figure 2B). However, when the organism is situated, the internal dynamics required to maintain the essential variables within their viability constraint changes as a function of the environment (blue and green regions, Figure 2C).

The picture of ultrastability in a dynamical system that emerges is different from Ashby’s and subsequent models (Figure 2C). Assume a certain dynamics of the autonomous agent is suitable to maintain its essential variables within their viability constraints for a certain environment, \mathcal{E}_1 , such that the coupled agent-environment interaction (blue trajectory) remains within the internal viable region (blue volume). Assume also a change to the environment, \mathcal{E}_2 , such that the agent dynamics required to survive is different from the previous one (green volume). If the new coupled dynamics fall outside the new internal viable region, the agent’s viability is threatened, setting off the proxies to the essential variables. The proxies drive the organization of the system towards an exploratory regime (magenta trace), until the behavior regains viability. The system regains viability by finding an internal dynamic that when coupled to the environment remains within the internal viable region.

In this view, a behaviorally ultrastable system must have three properties: (1) a rich reservoir of autonomous dynamics; (2) an exploratory regime capable of navigating the repertoire of autonomous dynamics; and (3) the ability to integrate those two regimes by modulating some aspect of the system via the proxies to the essential variables.

Methods

We propose to study ultrastability by studying a simplified version of the extended view described above. We take for granted the self-constitution of the agent and the existence of a viability region. We also abstract away the agent-environment interaction. We assume the existence of N different environments such that, when combined with the agent's viability, they produce a series of internal viable regions. The organism is modeled as a network of N interacting neurons with C constrained ones and one proxy to the essential variables that can affect the whole network. The constrained neurons, in interaction with the environment, determine the state of the proxy to the essential variable. We artificially evolve these simple neural networks to maintain their proxies to the essential variables within viable ranges.

Neural network We used continuous-time recurrent neural networks (CTRNNs) as a model of the organism's internal dynamics. Each component in the network is governed by the following state equation (Beer, 1995):

$$\tau_i \dot{y}_i = -y_i + \sum_{j=1}^N w_{ji} \sigma(g_i(y_j + \theta_j)) + I_i \quad (1)$$

where y is the activation of each node, τ is its time constant, w_{ji} is the strength of the connection from the j th to the i th node, g is a gain term, θ is a bias term, $\sigma(x) = 1/(1 + e^x)$ is the standard logistic activation function, I_i represents the external input to the neuron, and N represents the number of nodes in the network. The model does not include any form of synaptic plasticity mechanisms.

Agent-environment interaction For an adaptive organism, different environments are likely to require different internal dynamics. We abstract the agent-environment interaction by imposing arbitrary 'internal viable regions' over the neuron output space. Each of those regions is considered adaptive to a particular environment. That is, the dynamics available within this region of the phase-portrait are capable of maintaining the essential variables of the organism within their viability constraints. For simplicity, each environment is associated with one of the corners of the output space of the constrained neurons. The agent is considered adapted to the environment if its state in neural output is a distance less than 0.49 away from its corner.

Proxy to the essential variable According to the idealized rules above, at any point in time the agent can be adapted ($a = 0$) or unadapted ($a = 1$) to the environment. When the agent is adapted, the proxy to the essential variable rests at 0.0. When the agent is unadapted, the proxy to the essential variable increases over time, until it reaches one, according to the following state equation: $\tau \dot{E} = -E + a$. The proxy to the essential variable can affect the neurons

via the external input via a set of weights: $I_i = gw_i E$. In one of the experiments, we allowed for the proxy to the essential variable to modify the gain parameter instead: $g_i = g_i + sw_i E$, which is otherwise set to 1.

Evolutionary algorithm The parameters of the model were evolved using a real-valued genetic algorithm (Bäck, 1996). The optimization algorithm was run for populations of 100 individuals. We evolved the following parameters (ranges are shown in brackets): τ_i [1, 10]; w_{ji} , θ_j , and sw_i [-10, 10]; gw_i [-0.5, 0.5]. Parameters were encoded in a vector of real-values between [-1, 1] and linearly mapped to their corresponding ranges. Each time the algorithm was run, individuals were initialized by random selection from the range of each parameter. Populations were seeded with center-crossing networks (Mathayomchan and Beer, 2002). At the end of a run, the parameters of the best performing individual were stored for later analysis.

Fitness evaluation Network activity is initialized to 0.0 once at the beginning of a trial. During a fitness trial, the environment changes to a new one every 200 units of time. The fitness of a trial is measured as the integral of the adapted variable over time.

$$f = \frac{\int_{t=0}^{200} w(t)(1 - a(t))}{c \int_{t=0}^{200} w(t)} \quad (2)$$

where t is time, $a(t)$ is the binary adapted variable. The contribution of the adapted variable is weighted over time for each environment with a power function of time, $w(t) = t^2$, such that unadaptedness immediately after a change of environment weighs less than unadaptedness later on. Finally, in order to minimize the use of the critical boundary within the dynamics of stability, fitness is penalized for the number of times the agent becomes unadapted within a trial, c . The final fitness of a circuit is the average over all trials.

Incremental evolution The network is faced with two problems of a different nature. The first, to evolve attractors in each of the adapted regions, such that the network can stay within the adapted boundaries. The second, to transition between these regions. To allow for incremental evolution we divided the task into two stages. First, a minimal transition stage, where the network is asked to move from one environment to another one, always in the same sequence. Populations that are successful in this stage, move on to a second stage, where the network is asked to transition from any one environment to any other one, in random sequence.

Alternative fitness evaluations The fitness function defined above is based on the overt behavior, not the internal mechanisms of the circuit. In a final set of experiments, we use two fitness functions defined more mechanistically.

First, a function that maximizes the number of attractors in the phase-portrait. The network is started at different parts of the output space (3^C starting points that cover the full range of the constrained neurons) and integrated for 100 units of time. The fitness is calculated according to: $f_a = (k/2^C)^5$, where k are the number of attractors found in the circuit. The measure is normalized by the maximum number of attractors possible in the constrained circuit (2^C). The proxy to the essential variable is fixed to 0 for this fitness evaluation. Second, a function that maximizes the area covered by a single dynamic in the phase-portrait. The network is started at one randomly chosen point in the output space and integrated for $50 * 2^C$ units of time. The fitness is calculated by counting the number of corners of the constrained neural output space the circuit covers according to: $f_b = (\sum_{i=1}^{2^C} m_i/2^C)^5$, where $m = 1$, if the trace travelled near enough to the corner, $d^* < 0.25$, where d^* is the minimum distance to the corner in the full trajectory; otherwise $m = 1.0 - ((d^* - 0.25)/0.75)$. The proxy to the essential variable is fixed to 1 for this fitness evaluation. For this experiments, some of the ranges of the parameters were made larger: w_{ji} , θ_j , and sw_i [-50, 50], and w_{ii} [4, 50]. When combined, the total fitness function was the product: $f = f_a * f_b$.

Results

2D Ultrastability Task

Two-neuron networks The first set of experiments examined the ability of two-node CTRNNs to solve the simplest version of the ultrastability task: both neurons constrained (2C2N). In this task there are four possible environments, each with a respective internal viable region within the neural output space. The best two-node circuits attained a fitness of around 75%. That is, a typical successful network can only adapt to 3 of the four possible environments and transition between them in order (Figure 3). Evolutionary searches led to such networks 93 out of 100 times.

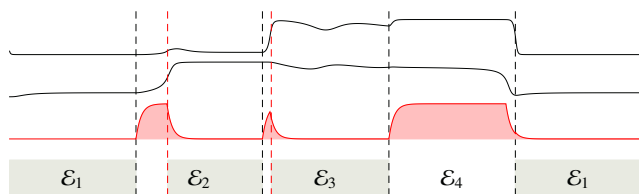


Figure 3: Neural traces during environment transitions for the 2C2N condition. Top two black traces represent neural output over time. Red trace represents the proxy to the essential variable. Black dashed vertical lines represent changes of environment. The environment is labeled at the bottom. Each environment is further divided by a red dashed vertical line into unadapted (no shade) and adapted (gray shade) stages.

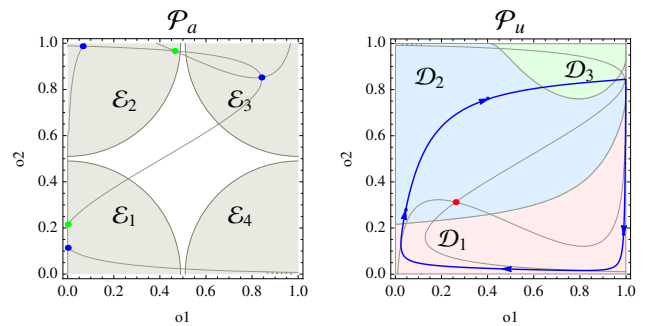


Figure 4: Phase portraits for the 2C2N condition. Adapted (\mathcal{P}_a): stable equilibrium points (blue disks), saddle points (green disks), nullclines (solid gray curves), adapted regions for each environment (gray regions, labeled). Unadapted (\mathcal{P}_u): unstable equilibrium point (red disk), limit cycle (blue trace). Basins of attraction of the adapted phase portrait are shown as colored volumes. The labels of the basins of attraction have a correspondence with the labels of the environments they are adapted to.

To understand the dynamics of the evolved circuit we analyzed the phase portraits of the most successful network when decoupled from the environment in the adapted and unadapted conditions independently by fixing the proxy to the essential variable to 0 or 1, respectively. The dynamics of a network are determined by the relative configuration of the nullclines. The network evolved three attractors (blue disks) in the phase portrait of the adapted condition (\mathcal{P}_a , Figure 4). The attractors cover three of the adapted regions (gray areas). In the unadapted condition, the network exhibits a limit cycle (\mathcal{P}_u , Figure 4). The basins of attraction, delimited by the saddle manifolds, of the attractors in the adapted phase portrait are visualized as light colored areas in the unadapted phase portrait. As can be seen, the limit cycle can transit between all evolved basins of attraction in the adapted phase. As we will see ahead, it this combination of: (a) the shapes of the basins of attraction in the adapted phase portrait, and (b) the trajectory of the exploratory behavior in the unadapted phase portrait, that are crucial for the success of the adaptation.

To understand the limitations of the 2-neuron network, we need to understand how the proxy to the essential variable affects the dynamics of the network. A successful circuit needs two distinct types of dynamics: a rich repertoire of attractors, and a maximally exploratory dynamics. The maximum number of attractors in a circuit occurs when each nullcline intersects maximally. In the two node circuit, a limit cycle that can explore all the internal viable regions requires that the nullclines ‘fit’ inside of each other. There are other possible limit cycles in 2-neuron networks, but they don’t cover as much of the output space (Beer, 1995). The change in input to the neuron by the proxy to the essential variable

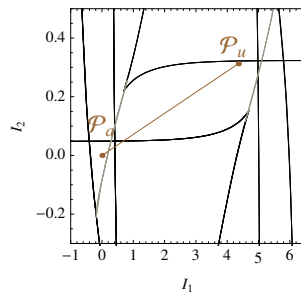


Figure 5: Parameter space for the 2C2N condition. Adapted phase (\mathcal{P}_a). Unadapted phase (\mathcal{P}_u). Regions of parameter that are crossed as the proxy to the essential variable, E , is gradually changed between 0 and 1 (brown line). Saddle node bifurcation (black curves). Hopf bifurcation (gray curves).

effectively shifts the nullcline up/down or left/right. A useful way to visualize this movement is in parameter space (Figure 5). The state of the proxy to the essential variable, by affecting the external input of the neurons, effectively moves the system through its parameter space. Different points in parameter space entail different dynamics. Crossing certain boundaries entails the system undergoing a bifurcation. Constrained to move in one direction in a line, the most successful circuits for this task evolve to navigate between a limit cycle and, at best, three stable attractor dynamics. From the limit cycle configuration, there is no way to shift both nullclines in one direction in a line to make them intersect maximally.

To understand the adaptive behavior of the coupled system, we need to understand the relationship between the shapes of the basins of attraction in the adapted phase portrait, and the trajectory of the exploratory behavior in the unadapted phase portrait. During a trial, the agent re-adapts to new environments by moving back and forth between the adapted (\mathcal{P}_a) and unadapted (\mathcal{P}_u) configurations. The timing of when the portraits change in relation to the basin boundary of the attractors is key to successful transitions. To illustrate this idea, we visualize the first transition of the trial shown in Figure 3, with the state of the network (brown trace) imposed over the changing phase-portrait (Figure 6). At the start of a run, the system starts off within the basin of attraction of the adapted environment (\mathcal{D}_1 , Figure 4). The first time the environment changes, the proxy to the essential variable changes, the portrait of the system transitions towards the limit cycle configuration, and the state of the system begins to follow it ($\mathcal{E}_2, \mathcal{P}_u$, Figure 6). When the system hits the boundary of the adapted region of the second environment, the proxy to the essential variable begins to return to normal, changing the portrait configuration back to the original ($\mathcal{E}_2, \mathcal{P}_a$, Figure 6). Because the state of the system is now within the basin of attraction of the top-left

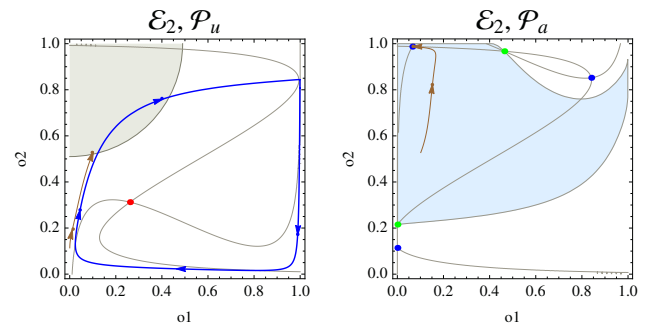


Figure 6: Neural activity (brown trace) in output space imposed over the phase portraits for the 2C2N condition. The two panels show the transition into \mathcal{E}_2 (from \mathcal{E}_1) from the example run in Figure 3. During the transition, the dynamics are first unadapted (\mathcal{P}_u), then adapted (\mathcal{P}_a).

attractor point, as soon as the limit cycle disappears, the system falls into it - effectively changing the dynamics of the system within the same configuration. A similar process occurs for each of the transitions.

The most successful 2-node networks displayed a similar dynamical configuration: (a) three equilibrium points in the adapted phase-portrait; (b) a limit cycle in the unadapted phase-portrait that allow them to transit between the basins of attraction of the adapted phase-portrait.

Adding unconstrained neurons One approach to obtain complete solutions to the 2-dimensional ultrastability problem is to increase the size of the network. This second set of experiments examined the ability of three-node CTRNNs to solve the same 2-dimensional version of the ultrastability task (2C3N). Same as before, there are four possible environments, each with a respective internal viable region within the neural output space of the two constrained neurons. The best three-node circuits attained a fitness close to 100%. Evolutionary searches led to successful networks (i.e., fitness greater than 98%) 38 out of 100 runs. A typical successful network was able to adapt to all four possible environments and transition between them in order (Figure 7). These successful networks can also transition successfully between any possible two regions. The networks use the output of the third unconstrained neuron (blue trace) to help them in both components of the task: (a) to have more attractors, and (b) to transition between them.

Similar to the two-neuron case, to understand the behavior of the network we can analyze the phase portraits of the dynamical system when decoupled from the environment in the adapted and unadapted conditions independently by fixing the proxy to the essential variable to 0 or 1, respectively (Figure 8). Analysis of a representative network shows four attractors in the phase portrait of the adapted condition (blue disks, \mathcal{P}_a). The attractors cover each of the four adapted regions (gray volumes). In the unadapted exploratory condi-

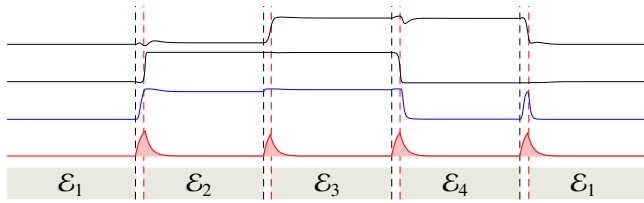


Figure 7: Neural traces for the 2C3N condition. Neural output of the constrained neurons over time (black traces). Neural output of the unconstrained neuron (blue trace). Proxy to the essential variable (red trace). Changes of environment (dashed vertical lines). Environment labeled at the bottom. Each environment is further divided by a red dashed vertical line into unadapted (no shade) and adapted (gray shade) stages.

tion, the network exhibits a limit cycle (blue trace, \mathcal{P}_u). The limit cycle begins to form as soon as the proxy to the essential variable becomes greater than 0.0, which simultaneously turns on the third neuron (bifurcation diagram not shown). Crucially, for the purposes of the ultrastable behavior, the limit cycle transits through all of the basins of attraction of the adapted phase portrait (colored regions). During the exploratory phase, as soon as the state of the system becomes adapted with respect to the current environment, the proxy to the essential variable goes back to normal, the exploratory portrait gives way back to the multiple attractor portrait, and the state of the system is contained within the basin of attraction of the currently adaptive configuration.

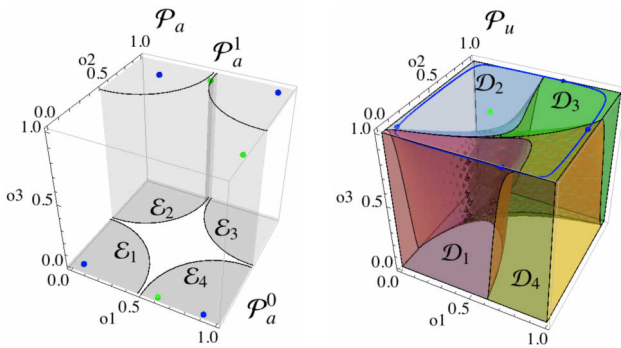


Figure 8: Phase portraits for the 2C3N condition. Adapted (\mathcal{P}_a): equilibrium points (blue disks), saddle points (green disks), adapted regions for each environment (gray regions, labeled). The phase portraits can be divided into two depending on the output of the third neuron: off (\mathcal{P}_a^0) or on (\mathcal{P}_a^1). Unadapted (\mathcal{P}_u): limit cycle (blue trace). Basins of attraction of the adapted phase portrait are shown as colored volumes. The labels of the basins of attraction have a correspondence with the labels of the environments they are adapted to.

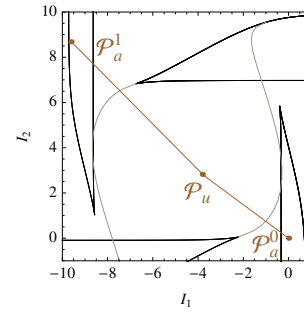


Figure 9: Parameter space for the 2C3N condition. Movement (brown trace) from the unadapted dynamics (\mathcal{P}_u) to the adapted dynamics ($\mathcal{P}_a^0, \mathcal{P}_a^1$). The additional neuron allows for the system to affect the system in two different directions. Saddle node bifurcation (black curves). Hopf bifurcation (gray curves).

This network overcomes the limitation of the 2-neuron network using the added dimension from the additional unconstrained neuron. To see how, we can re-interpret this system as a two-neuron network with two parameters: the proxy to the essential variable and the third, unconstrained neuron. We can visualize the different dynamics of the network in this two dimensional parameter space (Figure 9). When the proxy to the essential variable is at rest, and the unconstrained neuron is off, the system is in a two-attractor configuration (\mathcal{P}_a^0). Like in the two-dimensional case, when the proxy to the essential variable moves away from the base, the nullclines of both neurons shift in one direction in a line, moving the organization of the system towards a limit cycle configuration (\mathcal{P}_u). From the same original configuration, leaving the proxy to the essential variable at rest, but activating the third neuron, moves the system in a different direction (\mathcal{P}_a^1). Therefore, adding unconstrained components allows the nullclines of the constrained neurons to shift in multiple ways. Ultimately, from the limit cycle configuration, depending on the state of the third neuron, the system can transition to different regions of parameter space (brown lines) until it finds an internal viable region for the current environment.

Modifying neuron gains There are variations to the model that can overcome the limitations of the 2-neuron network. These variations are possible when the proxy to the essential variable can affect parameters other than the external input. This is an interesting condition to consider because recent work in neuroscience has shown how neuromodulators can modify the intrinsic properties of neurons (Marder and Thirumalai, 2002). For example, when the proxy to the essential variable is allowed to affect the gain of the circuit (g_i in Eq. 1), rather than the input, then it is possible to evolve a 2-neuron network that can adapt to all four environments, and transition between each of them. The

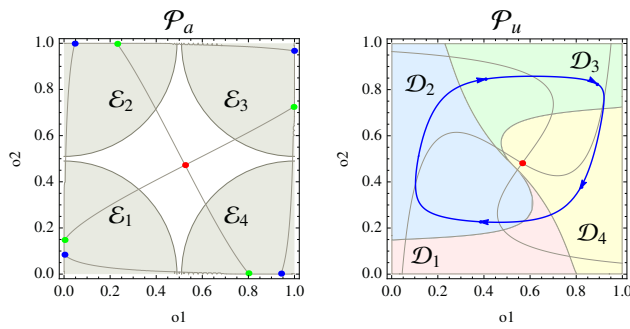


Figure 10: Phase portraits for the 2C2N where the proxy to the essential variable can affect the gains of the neurons. Adapted (\mathcal{P}_a) and unadapted (\mathcal{P}_u). The limit cycle in the unadapted phase-portrait transits all the basins of attraction of the adapted phase-portrait.

system easily changes between four attractors and a limit cycle (Figure 10). Change in the gain causes the nullclines to ‘twist.’ With this kind of modification available, it is possible for a system to change between the limit cycle configuration, where the nullclines are ‘fit’ inside each other, and the maximal-attractor configuration, where the nullclines intersect maximally, by ‘twisting’ the nullclines until they intersect the two other branches of the other nullclines.

Ultrastability in Higher Dimensions

Evolving networks within the current restrictions of the task (i.e., internal viable regions in the corners of the neural output space) and model (i.e., proxy to the essential variable modifying only the external inputs) proved to be rather difficult for higher dimensions of the task. There are three tasks a behaviorally ultrastable system has to accomplish. First, it must maximize the number of attractors. Second, it must have access to a cyclic attractor that maximizes the coverage of its state space. Third, it must integrate the two dynamics so as to solve the ultrastable task. We designed a series of experiments to identify which of those tasks becomes harder in higher dimensions.

First, we tested each of the tasks individually, by directly evolving networks to either maximize the number of attractors or to maximize the area covered by network’s activity in output space. It was relatively easy to evolve larger networks to have the maximum number of attractors in their portraits. For example, after 5000 generations, 9 out of 10 circuits evolved successfully ($> 99\%$ fitness) on the 5C5N task (i.e., 5-neuron network with 32 attractors). Similarly, it was easy to evolve larger networks to have dynamics that maximized the coverage of their output space. For example, after 5000 generations, 5 out of 10 circuits evolved successfully ($> 99\%$ fitness) on the 5C5N task (i.e., a 5-neuron network with a single dynamic that allowed it to travel through all 32 corners of the neural output space). As far as we can

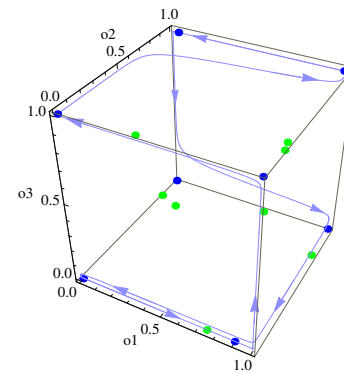


Figure 11: Phase portraits for the 3C5N condition. Projections in 3-dimensional space of the 5-dimensional phase portraits. Both adapted and unadapted portraits are shown in the same projection. The adapted portrait (\mathcal{P}_a) contains 8 attractors in the corners (blue disks) and 8 saddle points (green disks). The unadapted portrait (\mathcal{P}_u) contains one limit cycle (light blue trace) that transits close enough to all corners.

tell, the individual tasks scale well to higher dimensions.

We then tested evolving agents using the combined fitness function: one component to maximize the number of attractors while the proxy to the essential variable is set to 0 and another component to maximize the area covered by network’s activity in output space while the proxy to the essential variable is set to 1. We performed 50 evolutionary runs for 3C3N, 3C4N, and 3C5N tasks. We found no solutions for 3-, and 4-neuron networks. We found 4 out of 50 successful ($>99\%$) 5-neuron networks. These results suggest the difficulty is in the integration of the two distinct behaviors.

As expected, the successful networks had 8 attractors in the adapted condition (blue disks, Figure 11), and an intricate cycle in the unadapted condition that passed relatively close to all the attractors (light blue trace, Figure 11). Due to the higher dimensionality, analyzing the basins of attraction of this network and their relationship to the limit cycle was not attempted. These networks were only partially successful because although they integrate the two tasks, when tested on the ultrastable task, they do not always transition successfully between all possible environments.

A number of key challenges remain to be studied. It will require further analysis to understand the difficulties encountered in higher dimensions. A few things to consider are: (1) adding unconstrained neurons; (2) further analyzing the dynamics of the partial solutions; and (3) using alternative arrangements of the internal viable regions.

Discussion

This paper analyzes the dynamical possibilities for mechanisms of adaptation in living organisms based on Ashby’s framework of ultrastability. To explore alternative mecha-

nisms to the ones studied by Ashby and subsequent works, we developed evolutionary conditions that selected for behaviorally ultrastable systems. We analyzed the resulting neural networks using dynamical systems theory for the simplest conditions. The picture that emerged from our analysis opens up the idea of ultrastable mechanisms to a broader set than the original. It also allows for a more concrete instantiation of the framework within the context of neural networks.

The analysis in this paper was purposefully limited to the smallest networks in the lowest dimensional conditions. It's not hard to generalize our re-interpretation of ultrastable systems to higher dimensions. A system with many neurons can have a vast set of internal viable regions. The regions can form intricately connected networks in the high-dimensional output space. The proxies to the essential variables shift the rich repertoire of dynamics into the exploratory configuration, allowing the network to transition between different internal viable regions. The dynamics of exploration can be richer than just limit cycles, including for example chaotic dynamics. Re-adaptation corresponds to shifts in the phase portrait configuration: out of the exploratory regime, back into the rich dynamical repertoire. Crucially, in any one system, it is possible that only a fraction of the internal viable regions are explored during its lifetime, depending on the environmental changes experienced. In this view, the co-development of a rich dynamical repertoire together with a sufficiently encompassing exploratory dynamic occurs over evolutionary timescales, endowing agents with richer ultrastable systems to increase their probabilities of re-adaptation to novel environments.

Although the simplified 'internal viable regions' used in this study makes for tractable initial analysis, there are a number of conceptual variations not considered in this paper that are easily conceivable within the same framework. First, the shape and location of the region that is adaptive for any one environment can be any, not just the corners. Second, an internal viable region can contain more than one attractor, the boundaries of the region delimited by their combined basins of attraction. Third, internal viable regions for different environments can overlap. A simple re-interpretation of some of the networks analyzed in this paper could illustrate any of these variations. Interestingly, one idea that follows immediately from having richer internal viable regions is that which attractor within the region an agent finds itself in will depend on its history of interaction with the environment. Ultimately, the idea for this framework is to use internal viable regions defined by the actual interaction between an agent with certain essential requirements for its constitution and the environment.

The framework of behavioral ultrastability proposed in this paper could be developed further to study habit formation. In this paper, agents are evolved first to transition between environments in a specific order, and once successful, to transition between any possible environment. In both con-

ditions, the history of agent-environment interactions does not contain additional information about the likeliness of the next possible environment. An interesting space of solutions to study would be agents evolved to cope with a subset of all possible sequences of transitions. Changes from one sequence of transitions to a different sequence could involve a second level of adaptation, where the exploratory dynamics changes as a function of the sequence experienced. This would allow for the study of mechanisms with slower time constants that keep track of the sequence of transitions to modulate the exploratory dynamics.

Acknowledgements

The revised paper benefited from discussions with Thomas Buhrmann, Matthew Egbert, and Inman Harvey. This work was supported by NSF grant No. IIC-1216739 and in part by TIN2011-24660. EUCogIII financed M.A.'s visit to I.U.

References

- Ashby, W. (1960). *Design for a brain: The origin of adaptive behavior*. Chapman & Hall, 2nd edition.
- Bäck, T. (1996). *Evolutionary algorithms in theory and practice*. Oxford University Press.
- Beer, R. (1995). On the dynamics of small continuous-time recurrent neural networks. *Adaptive Behavior*, 3:471–511.
- Di Paolo, E. A. (2000). Homeostatic adaptation to inversion in the visual field and other sensorimotor disruptions. In Meyer, J.-A., Berthoz, A., Floreano, D., Roitblat, H., and Wilson, S., editors, *From Animals to Animats 6*, pages 440–449. MIT Press.
- Harvey, I. (2008). Homeostasis via chaos: implementing the uniselecter as a dynamical system. In Bullock, S., Noble, J., Watson, R. A., and Bedau, M. A., editors, *Proc. of the Eleventh Int. Conf. on Artificial Life*, page 774. MIT Press.
- Herrmann, J., Hilicki, M., and Der, R. (2004). On Ashby's homeostat: A formal model of adaptive regulation. In Schaal, S., Ijspeert, A., A., B., Vijayakumar, S., Hallam, S., and Meyer, J.-A., editors, *From Animals to Animats 8*, pages 324–333. MIT Press.
- Iizuka, H. and Di Paolo, E. A. (2008). Extended homeostatic adaptation: Improving the link between internal and behavioural stability. In Asada, M., Hallam, J., Meyer, J.-A., and Tani, J., editors, *From Animats to Animals 10*, pages 1–11. Springer.
- Izquierdo, E., Harvey, I., and Beer, R. (2008). Associative learning on a continuum in evolved dynamical neural networks. *Adaptive Behavior*, 16:361–384.
- Marder, E. and Thirumalai, V. (2002). Cellular, synaptic and network effects of neuromodulation. *Neural Networks*, 15:479–493.
- Mathayomchan, B. and Beer, R. (2002). Center-crossing recurrent neural networks for the evolution of rhythmic behavior. *Neural Computation*, 14:2043–2051.
- Yamauchi, B. and Beer, R. (1994). Sequential behavior and learning in evolved dynamical neural networks. *Adaptive Behavior*, 2:219–246.