Tackling Learning Intractability Through Topological Organization and Regulation of Cortical Networks

Jekanthan Thangavelautham, *Member, IEEE,* Gabriele M.T. D'Eleuterio, *Member, IEEE,*

***Abstract*—A key challenge in evolving control systems for robots using neural networks is training tractability. Evolving monolithic fixed topology neural networks is shown to be intractable with limited supervision in high dimensional search spaces. Common strategies to overcome this limitation are to provide more supervision by encouraging particular solution strategies, manually decomposing the task and segmenting the search space and network. These strategies require a supervisor with domain knowledge and may not be feasible for difficult tasks where novel concepts are required. The alternate strategy is to use self-organized task decomposition to solve difficult tasks with limited supervision. The Artificial Neural Tissue (ANT) approach presented here uses self-organized task decomposition to solve tasks. ANT inspired by neurobiology combines standard neural networks with a novel wireless signaling scheme modeling chemical diffusion of neurotransmitters. These chemicals are used to dynamically activate and inhibit wired network of neurons using a coarse-coding framework. Using only a global fitness function that does not encourage a predefined solution, modular networks of neurons are shown to self-organize and perform task decomposition. This approach solves the sign-following task found to be intractable with conventional fixed and variable topology networks. In this paper, key attributes of the ANT architecture that perform self-organized task decomposition are shown. The architecture is robust and scalable to number of neurons, synaptic connections and initialization parameters.**

***Index Terms*—evolutionary algorithms, task decomposition, coarse coding, robotics**

I. Introduction

Biological nervous systems are highly adaptive, decentralized, extensible (takes into account future growth, needs), can generalize and are fault-tolerant. Artificial neural networks are simplified models of the nervous system that are used in robotics and have the potential to match the capabilities of biological control systems. They have three useful properties, that includes ability to *generalize* (i.e., find patterns, classify data), have synaptic *plasticity* (ability to alter synaptic connections permanently) and are capable of *universal* approximation [1] (represent any continuous function with a desired accuracy with sufficient number of hidden neurons).

However, artificial neural networks face several challenges that limit their potential in robotics and need to be addressed. One of the main challenges is how best to design neural network architectures to be extensible and tractable for solving difficult tasks, where there is limited knowledge or no reliable intuition of how to solve the task. Insufficient supervision during training results in intractability. For evolutionary training algorithms, the approach becomes intractable in high dimensional search spaces when it is harder to rank and select for incrementally better solutions for crossover and mutation resulting in premature search stagnation, known as the *bootstrap problem* [2].

Conventional strategies to overcome the bootstrap problem are by providing more supervision by manual decomposition of a task [3], [4] or by exploiting user knowledge to segment the task space and assigning the segments to different controller modules. Typically, pre-specified fixed topology neural network architectures are used. Fixed topology neural networks require that a designer specify a topology. This limits the adaptability and extensibility of the controller because the controller is designed for a specific task in mind. If additional tasks were given, then the topology may become unsuitable. Furthermore, the wrong choice of network topology can result in poor training performance or worse, intractability due to *spatial-crosstalk* [5] where parallel neurons provide conflicting or disruptive signals. How should a neural network architecture be organized to enable task decomposition?

In this work, this problem is addressed using a neural network architecture that draws heavily from nature called the Artificial Neural Tissue framework [6], [7]. The artificial neural tissue (ANT) superimposes on a typical feed-forward neural-network structure a novel coarse-coding mechanism inspired by the work of Albus and Hinton [8], [9]. Coarse coding allows for modules of neurons to be added, removed and rewired through training resulting in self-organized task decomposition. The coarse coding mechanism has a biological analogy as neurons can communicate not only electrically by exchanging signals along axons but also through chemical diffusion [10], [11] that activates and inhibits networks of neurons.

This coarse coding chemical signaling mechanism regulates (selectively activates and inhibits) wired connections between neurons reducing spatial crosstalk. This process of regulating wired connections between neurons results in compartmentalization and emergence of distinct modules. These modular networks of neurons are shown to self-organize to perform task decomposition. It is shown that the formation of these modules starts through competition followed by cooperation between modules. Through competition, the modules specialize and

Jekanthan Thangavelautham is with the Department of Mechanical Engineering, Massachusetts Institute of Technology, Cambridge, MA, 02139 USA e-mail:jekan@mit.edu (http://robots.mit.edu/jekan).

Gabriele M.T. D'Eleuterio is with the Institute for Aerospace Studies, University of Toronto, University of Toronto, Toronto, ON M3H 5T6, Canada e-mail:gabriele.deleuterio@utoronto.ca.

segment the search space. Through cooperation, newly discovered functionality is protected from damaging mutations. With limited supervision, this coarse-coding mechanism is shown to effectively solve the sign-following task found to be intractable with conventional fixed and variable topology networks. The mechanism is shown robust and scalable to number of neurons, synaptic connections and initialization parameters.

The remainder of this article is organized as follows. Section II presents some background and related work. Section III presents the Artificial Neural Tissue approach. Section IV and Section V present a representative task, namely the sign-following task used to demonstrate ANT's capabilities. This is followed by analysis in Section V and discussion in Section VI.

II. Related Work

Conventional training strategies for difficult tasks with high dimensional input spaces use predesigned modular neural networks. These networks consist of modules called 'expert networks' that are trained on subsets of the input space selected by an experimenter to perform task decomposition [5]. Typically this requires less training time and shows improved performance than a single monolithic network [5]. Arbitration is performed between the expert networks using a competitive Product of Experts model [12] or cooperative Mixture of Experts model [5]. With these methods, the arbitrating (gating) function and the expert networks use different error functions for backpropagation training.

The network topology and number of modules are typically predetermined by the experimenter based on domain knowledge of a task at hand. The mixture of experts models have been adapted to using evolutionary algorithms for the design of the networks and negative correlation learning for tuning the weights [13]. These methods are in contrast to the ANT architecture, where the design processes is the product of a single evolutionary training run.

Self-Organizing Maps (SOMs) are neural network architectures used to extract one or two dimensional representations of high dimensional input spaces [14]. The topology of the input data tends to be well preserved, enabling understanding of its structures [14], [15], [17]. Typically SOMs are fixed neural network architectures that require domain knowledge of an input space to design them. More recently, SOMs with dynamic topologies have been developed [16]. SOMs have also been developed for robotic control without requiring detailed models of a physical system [17]. For a typical robotic task, several SOMs are organized in a user defined hierarchical structure to perform task decomposition [17]. ANT avoids this problem with the effective use of a global fitness function for a given task. The fitness function mediates training and facilitates evolution of the neural network architecture to decompose and solve the task through trial and error.

Another class of neural network architectures known as Liquid State Machines (LSMs) [18], Echo State Networks [19] and Backpropagation Decorrelation [20] rely on reservoir (liquid) of randomly connected neurons. In Liquid State Machines a state vector linearly combines a reservoir of neurons to produce an output. Training is performed on the state vector not the reservoir. However this requires the reservoir topology be effective in separating the task space. Typically the reservoir is randomly generated using carefully chosen parameters specific to a problem domain and has been an area of active research [21], [22], [20]. For a general case, the generation of reservoirs may require searching hundreds or thousands of reservoirs to find a suitable one. Training algorithms have also been used to generate and find suitable reservoirs [22]. These algorithms are comparable to the state-of-the-art, in such fields as face recognition [23]. LSMs have also been suggested for motor-control similar to the cerebellum [24]. With these networks, the computational cost of the architecture is high because all the neurons in the reservoir need to be evaluated, although it is often unclear of what are contributions of each neuron in the reservoir.

ANT like LSMs benefit from having a reservoir of neurons in the form of a tissue. However, not all the neurons are active all the time in the tissue. This selective activation and inhibition is performed through a coarse coding process. This enables ANT to be computationally efficient and flexible compared to LSMs. Coarse coded selection can vary the number of active neurons during training. The training process could be explorative during early stages, looking for new strategies and taking risks, while acting as a filter and safeguarding innovate traits during later stages as shown in Section V.C.

Artificially evolved neural network architectures such as NEAT (NeuroEvolution of Augmenting Topologies) show the potential advantage of evolving both the neuron weights and topology concurrently [25]. It is argued that growing the network incrementally through ('complexification') helps minimize the dimensional search space and thus improve evolutionary performance [25]. Hierarchical Enforced SubPopulations (H-ESP) is another variable topology neural network architecture, where populations of neurons and the networks that contain them are evolved in parallel [26]. ANT is more flexible than NEAT and H-ESP. It can be initialized with a large number of neurons without the need for incremental 'complexification' as we see in Section V.C. The process evolves the networks and neurons concurrently. This is accomplished using coarse-coding that can effectively suppress unnecessary/disruptive neurons (that NEAT lacks) while activating neurons specialized for specific input signals.

Another approach to evolving solutions to complex tasks is to exploit encoding schemes that partition the search space. This includes a multicellular developmental system that uses gene regulatory networks to control neuron differentiation, division and death by Eggenberger [27]. In these systems, regulatory networks acts on the developmental program in constructing the phenotype. In contrast, within ANT, regulatory networks are active throughout an individual's lifetime, both during development and after.

Astor and Adami's [28] NORGEV tissue architecture is another multicellular development system that generates neurons arranged in a two-dimensional structure. Cell replication and connections are formed through a gene-regulated development process using a Genetic Programming type command set. Planar architectures such as NORGEV lack the laminar (columnar layering) of neurons found in cortical structures. Within ANT,

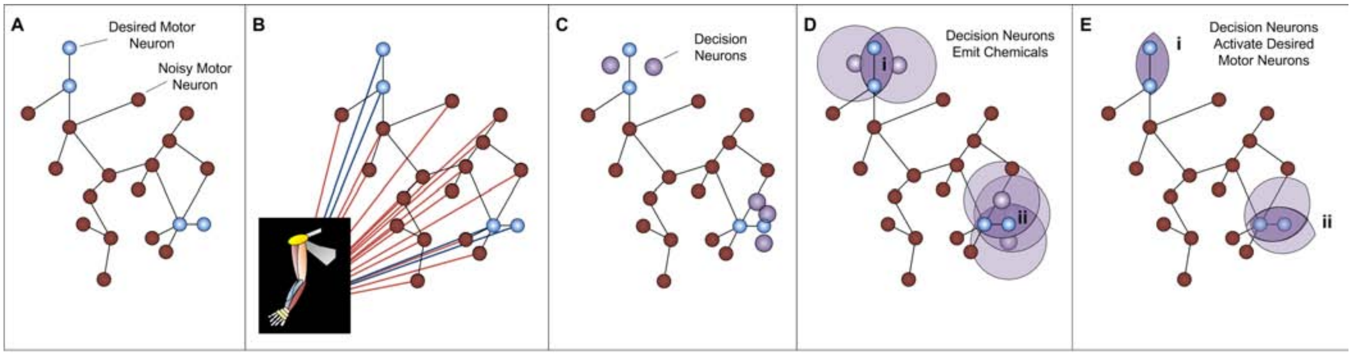

A
Desired Motor Neuron
Noisy Motor Neuron
B
C
Decision Neurons
D
Decision Neurons Emit Chemicals
i
ii
E
Decision Neurons Activate Desired Motor Neurons
i
ii

Fig. 1. In a randomly generated tissue, most motor neurons would produce noisy/incoherent output that would 'drown out' signals from a few desired motor neurons due to spatial crosstalk [5]. Neurochemicals emitted by decision neurons selectively activate networks of desired motor neurons (i) and (ii) by coarse-coding overlapping diffusion as shown. This inhibits noisy motor neurons and eliminates spatial crosstalk.

the neurons are organized into a three dimensional tissue, well suited for chemical diffusion of signals and better match existing observations from neurophysiology of columnar layering of neurons [29].

It has been argued that effectively encoding phenotypes (such as neural networks) for evolutionary training can decrease the search space by exploiting regularities through gene duplication [30]. Indirect developmental encoding schemes for neural networks such as Artificial Embrogeny (AE) systems [31] produces phenotypes through recursive rewriting of the genotype code. These systems use an artificial chemistry as a grammar or model cellular metabolism and replication. Other recursive rewriting schemes include Cellular Encoding [32], matrix-rewriting [33], L-Systems (see [34], [35]) to generate neural networks. Some Artificial Embrogeny systems combine multicellular development such as that by Dellaert and Beer [36] that can sense neighboring neurons and use Morphers (fixed topology modular feed-forward neural network without hidden layers) to signal cell replication and development with embryonal stages (DES) [37]. ANT relies on a developmental scheme to generate its tissue and in theory can exploit regularities in the task space through gene duplication. This can make task decomposition effective, by having not to repeatedly learn simillar concepts used in different subtasks.

III. Artificial Neural Tissue

The ANT architecture presented in this paper consists of a developmental program encoded in the 'genome,' that constructs a three-dimensional neural tissue and regulatory functions. Regulatory functions exists at two levels, gene/molecular level and at the cellular level. The tissue consists of two types of neural units, *decision neurons* and *motor-control neurons*, or simply motor neurons. Assume a randomly generated set of motor neurons in a tissue connected electrically (Figure 1a). Chances are most of these neurons will produce incoherent/noisy output, while a few may produce desired functions. If the signal from all of these neurons are summed, then these "noisy" neurons would drown out the output signal (Figure 1b) due to spatial crosstalk [5]. Within ANT, decision neurons emit chemicals that diffuse omnidirectionally (shown shaded) (Figure 1d). By coarse-coding multiple overlapping diffusion fields, the desired motor neurons can be selected and noisy neurons inhibited, referred to as neural regulation. With multiple overlapping diffusion field (Figure 1d ii), there is redundancy and when one decision neuron is damaged the desired motor neurons are still selected. Networks of selected neurons, (i) and (ii) when controlling the same output result in competition. Cooperation results with different output. Let us discuss the computational mechanism of the tissue in detail first and then outline the process by which the tissue is created.

A. Motor Neurons

We imagine the motor neurons within the tissue to be arranged in a regular rectangular lattice in which the neuron $N_{\boldsymbol{\lambda}}$ occupies the position $\boldsymbol{\lambda} = (l, m, n) \in \mathbb{I}^3$ (Figure 2). Depending on the activation functions used, the state $s_{\boldsymbol{\lambda}} \in \mathbb{S}$ of the neuron is either binary, i.e., $\mathbb{S}_b = \{0, 1\}$ or can be real, $\mathbb{S}_p = [0, 1]$ or $\mathbb{S}_r = [-1, 1]$.

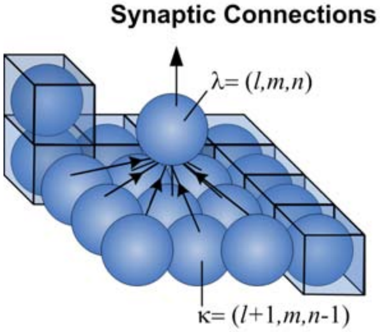


Fig. 2. Synaptic connections between ANT motor neurons from layer $l+1$ to l.

Each neuron $N_{\boldsymbol{\lambda}}$ nominally receives inputs from neurons $N_{\boldsymbol{\kappa}}$ where $\boldsymbol{\kappa} \in \Uparrow(\boldsymbol{\lambda})$, the nominal input set. Here we shall assume that these nominal inputs are the 3×3 neurons centered one layer below $N_{\boldsymbol{\lambda}}$; in other terms, $\Uparrow(\boldsymbol{\lambda}) = \{(i, j, k) \,|\, i = l-1, l, l+1;\ j = m-1, m, m+1;\ k = n-1\}$. (As will be explained presently, however, we shall not assume that all the motor neurons are active all the time.)

The sensor data are represented by the activation of the sensor input neurons $N_{\boldsymbol{\alpha}_i}, i = 1 \dots m$, summarized as $A = \{s_{\boldsymbol{\alpha}_1}, s_{\boldsymbol{\alpha}_2} \dots s_{\boldsymbol{\alpha}_m}\}$. Similarly, the output of the network is represented by the activation of the output neurons $N_{\boldsymbol{\omega}_j}, j = 1 \dots n$, summarized as $\Omega = \{s_{\boldsymbol{\omega}_1^1}, s_{\boldsymbol{\omega}_2^1}, s_{\boldsymbol{\omega}_3^2} \dots s_{\boldsymbol{\omega}_n^b}\}$, where $k = 1 \dots b$ specifies the output behavior. Each output neuron commands one behavior of the agent. (In the case of a robot, a typical behavior may be to move forward a given distance.

This may require the coordinated action of several actuators. Alternatively, the behavior may be more primitive such as augmenting the current of a given actuator.) If $s_{\boldsymbol{\omega}_j^k} = 1$, output neuron $\boldsymbol{\omega}_j$ votes to activate behavior k; if $s_{\boldsymbol{\omega}_j^k} = 0$, it does not. Since multiple neurons can have access to a behavior pathway, an arbitration scheme is imposed to ensure the controller is deterministic where $p(k) = \sum_{j=1}^{n} \gamma(s_{\boldsymbol{\omega}_j^i}, k) s_{\boldsymbol{\omega}_j^i} / n_k$ and $n_k = \sum_{j=1}^{n} \gamma(s_{\boldsymbol{\omega}_j^i}, k)$ is the number of output neurons connected to output behavior k where $\gamma(s_{\boldsymbol{\omega}_j^i}, k)$ is evaluated as follows:

$$\gamma(s_{\boldsymbol{\omega}_j^i}, k) = \begin{cases} 1, & \text{if } \ i = k \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

and resulting in behavior k being activated if $p(k) \geq 0.5$.

As implied by the set notation of Ω, the outputs are not ordered. In this embodiment, the order of activation is selected randomly. We are primarily interested here in the statistical characteristics of relatively large populations but such an approach would likely not be desirable in a practical robotic application. However, this can be remedied by simply assigning a sequence *a priori* to the activations. We moreover note that the output neurons can be redundant; that is, more than one neuron can command the same behavior, in which case for a given time step one behavior may be 'emphasized' by being voted multiple times.

B. The Decision Neuron

The coarse coding nature of the artificial neural tissue is performed by the decision neurons. Coarse coding is a distributed representation that uses multiple overlapping coarse fields to encode a finer field [8], [9]. For example, coarse receptive fields can be overlapping circles and the area represented by the finer field more accurately encodes for a position than the individual coarse receptive fields.

Decision neurons occupy nodes in the lattice as established by the evolutionary process (Figure 3). The effect of these neurons is to excite into operation or inhibit the motor control neurons (shown as spheres). Once a motor control neuron is excited into operation, the computation outlined in (3) is performed. Motivated as we are to seek biological support for ANT, we may look to the phenomenon of chemical communication among neurons. In addition to communicating electrically along axons, some neurons release chemicals that are read by other neurons, in essence serving as a 'wireless' communication system to complement the 'wired' one.

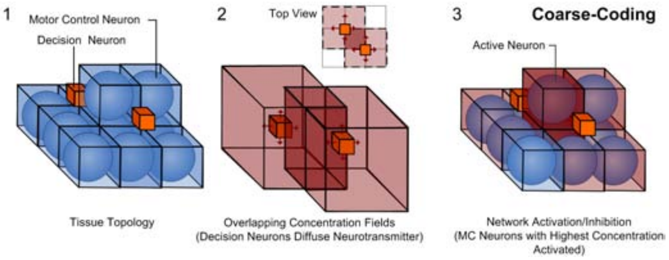

1
Motor Control Neuron
Decision Neuron
Tissue Topology
2
Top View
Overlapping Concentration Fields
(Decision Neurons Diffuse Neurotransmitter)
3
Coarse-Coding
Active Neuron
Network Activation/Inhibition
(MC Neurons with Highest Concentration
Activated)

Fig. 3. Coarse coding regulation being performed by two decision neurons (shown as squares) that diffuse a chemical, in turn activating a motor neuron column located at the center (right).

Each decision neuron can be in one of two states, one in which it diffuses a neurotransmitter chemical or remains dormant. The state of a decision neuron $T_{\boldsymbol{\mu}}$, $\boldsymbol{\mu}$ is binary and determined by one of the activation functions (see Section III.C). Assuming the decision neurons use the modular activation function described in Section III.C, the inputs to $T_{\boldsymbol{\mu}}$ are all the input sensor neurons $N_{\boldsymbol{\alpha}}$; *i.e.*, $s_{\boldsymbol{\mu}} = \psi_{\boldsymbol{\mu}}(s_{\boldsymbol{\alpha}_1} \ldots s_{\boldsymbol{\alpha}_m})$ where $\sigma_{\boldsymbol{\mu}} = \sum_{\boldsymbol{\alpha}} v_{\boldsymbol{\alpha}}^{\boldsymbol{\mu}} s_{\boldsymbol{\alpha}} / \sum_{\boldsymbol{\alpha}} s_{\boldsymbol{\alpha}}$ and $v_{\boldsymbol{\alpha}}^{\boldsymbol{\mu}}$ are the weights. The decision neuron is dormant if $s_{\boldsymbol{\mu}} = 0$ and releases a virtual neurotransmitter chemical of uniform concentration $c_{\boldsymbol{\mu}}$ over a prescribed field of influence if $s_{\boldsymbol{\mu}} = 1$.

Motor control neurons within the highest chemical concentration field are excited into operation. Only those neurons that are so activated will establish the functioning network for the given set of input sensor data. Owing to the coarse coding effect, the sums used in the weighted input of (2) are over only the set $\overline{\Uparrow}(\boldsymbol{\lambda}) \subseteq \Uparrow(\boldsymbol{\lambda})$ of active inputs to $N_{\boldsymbol{\lambda}}$. Likewise the output of ANT is in general $\overline{\Omega} \subseteq \Omega$. The decision neuron's field of influence is taken to be a rectangular box extending $\pm d_{\boldsymbol{\mu}}^r$, where $r = 1, 2, 3, ...$, from $\boldsymbol{\mu}$ in the three perpendicular directions. These three dimensions along with $\boldsymbol{\mu}$ and $c_{\boldsymbol{\mu}}$, the concentration level of the virtual chemical emitted by $T_{\boldsymbol{\mu}}$, are encoded in the genome.

Decision neurons emit chemicals that are used to selectively activate and inhibit motor control neurons. We label this component of ANT as the *neural regulatory system*. This is akin to genes being able to activate or inhibit other genes in a gene regulatory system.

C. Activation Function

The modular activation function allows selection among four possible threshold functions of the weighted input σ. The use of two threshold parameters, allows for a single neuron to compute the XOR function, in addition to the AND and OR functions. For this version of the modular activation function,

$$\begin{aligned}
\psi_{\text{down}}(\sigma) &= \begin{cases} 0, & \text{if } \sigma \geq \theta_1 \\ 1, & \text{otherwise} \end{cases} \\
\psi_{\text{up}}(\sigma) &= \begin{cases} 0, & \text{if } \sigma \leq \theta_2 \\ 1, & \text{otherwise} \end{cases} \\
\psi_{\text{ditch}}(\sigma) &= \begin{cases} 0, & \min(\theta_1, \theta_2) \leq \sigma < \max(\theta_1, \theta_2) \\ 1, & \text{otherwise} \end{cases} \\
\psi_{\text{mound}}(\sigma) &= \begin{cases} 0, & \sigma \leq \min(\theta_1, \theta_2) \text{ or } \sigma > \max(\theta_1, \theta_2) \\ 1, & \text{otherwise} \end{cases}
\end{aligned} \tag{2}$$

and the weighted input $\sigma_{\boldsymbol{\lambda}}$ for neuron $N_{\boldsymbol{\lambda}}$ is nominally taken as

$$\sigma_{\boldsymbol{\lambda}} = \frac{\sum_{\boldsymbol{\kappa} \in \Uparrow(\boldsymbol{\lambda})} w_{\boldsymbol{\lambda}}^{\boldsymbol{\kappa}} s_{\boldsymbol{\kappa}}}{\sum_{\boldsymbol{\kappa} \in \Uparrow(\boldsymbol{\lambda})} s_{\boldsymbol{\kappa}}} \tag{3}$$

with the proviso that $\sigma = 0$ if the numerator and denominator are zero. Also, $w_{\boldsymbol{\lambda}}^{\boldsymbol{\kappa}} \in \mathbb{R}$ is the weight connecting $N_{\boldsymbol{\kappa}}$ to $N_{\boldsymbol{\lambda}}$. We may summarize these threshold functions in a single analytical expression as

$$\psi = (1-k_1)[(1-k_2)\psi_{\text{down}} + k_2\psi_{\text{up}}] + k_1[(1-k_2)\psi_{\text{ditch}} + k_2\psi_{\text{mound}}] \tag{4}$$

where k_1 and k_2 can take on the value 0 or 1. The activation function is thus encoded in the genome by k_1, k_2 and the threshold parameters $\theta_1, \theta_2 \in \mathbb{R}$.

D. Evolution and Development

A population of ANT controllers is evolved in an artificial Darwinian manner. The 'genome' for a controller contains a 'gene' for each cell with a specifier D used to distinguish the functionality (between motor control, decision and tissue). A constructor protein (an autonomous program) interprets the information encoded in the gene and translates this into a cell descriptor protein (Figure 4). The gene 'activation' parameter is a binary flag resident in all the cell genes and is used to either express or repress the contents of the gene. When repressed, a descriptor protein of the gene content is not created. Otherwise, the constructor protein 'grows' a cell. Each cell position is specified in reference to a seed-parent address. A cell-death flag determines whether the cell commits suicide after being grown. Once again, this feature in the genome helps in the evolutionary process with a cell committing suicide still occupying a volume in the lattice although it is dormant. In otherwise retaining its characteristics, evolution can decide to reinstate the cell by merely toggling a bit through mutation.

In turn mutation (manipulation of gene parameters with a uniform random distribution) to the growth program results in new cells being formed through cell division. The rate at which mutation occurs to a growth program is also specified for each tissue and is dependent on the cell replication probability parameter T_r. This probability parameter is used to determine whether a new cell is inserted. Cell division requires a parent cell (selected with highest replication probability relative to the rest of the cells within the tissue) and copying $m\%$ of the original cell contents to a daughter cell (where m is determined based on uniform random distribution), with the remaining cell contents initialized from a uniform random distribution. This models a gene duplication process with the first $m\%$ being a redundant copy of an existing gene and the remaining contents being malformed thus resulting in a subfunctionalization process, in which a daughter gene adopts part of the functions of its parental gene [38].

The "Cell Type" of each new cell is determined based on the ratio of motor control neurons to decision neurons, a parameter specified in the tissue gene. The new neuron can be located in one of six neighboring locations (top, bottom, north, south, east, west) chosen at random and sharing a common side with the parent and not occupied by another neuron.

E. Crossover and Mutation

Within ANT, the genome is compartmentalized, through a direct encoding scheme, with each gene specifying the characteristics of each neuron within the tissue. Crossover is interchange of a set of genes between two parents to form a child. Before crossover, a parent affinity parameter, $\varpi \in \{0, 1\}$ is chosen at random for each child genome. The affinity parameter is used to establish if each child genome has closer 'affinity' to one of its parents (either parent A or parent B). Thus if $\varpi = 0$ then the genome has closer 'affinity' to parent A and $\varpi = 1$ if it has affinity with parent B. Once the affinity parameter is established for a child genome, a crossover plane is drawn along the same position shown for the two parents in Figure 5.

It should be noted that each gene remains intact and the crossover operation does not result in an arbitrary segmentation and interchange of gene contents. Each neuron has a unique position parameter $\boldsymbol{\lambda} = (l, m, n)$ relative to rest of the neurons within the tissue, a crossover is performed by drawing a plane (with a normal vector parallel to the *x* or *y*-axis) separating the tissue. Cell genes within the parent genome located on the side of the plane closer to the origin is copied directly onto the child genome with the associated 'affinity' parameter and the remaining genes are interchanged between the parents based on a 'compatibility criterion' (Figure 5).

The 'compatibility criterion' imposes the following condition, that gene for neuron $N_{\boldsymbol{\lambda}_1}$ from parent A and $N_{\boldsymbol{\lambda}_2}$ from parent B could be interchanged if $\boldsymbol{\lambda}_1 = \boldsymbol{\lambda}_2$, i.e., have the same position after development and only when *both* genes are expressed or repressed during development. Thus child 1 with $\varpi = 0$ (affinity to parent A) assumes the gene for ${}^{B}N_{\boldsymbol{\lambda}_2}$ and child 2 with $\varpi = 1$ (affinity to parent B) assumes ${}^{A}N_{\boldsymbol{\lambda}_1}$. If the compatibility criterion is not met, then no interchange occurs, and thus ${}^{A}N_{\boldsymbol{\lambda}_1}$ is passed onto child 1 and ${}^{B}N_{\boldsymbol{\lambda}_2}$ is passed onto child 2. Thus if ${}^{A}N_{\boldsymbol{\lambda}_1}$ is not expressed in parent A and ${}^{B}N_{\boldsymbol{\lambda}_1}$ is expressed in parent B, then this pair of genes fail the 'compatibility criterion.'

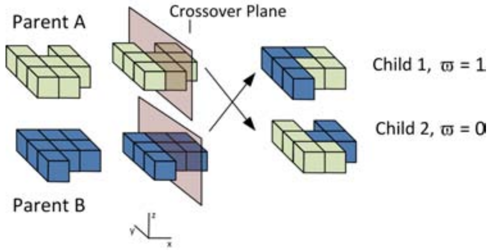
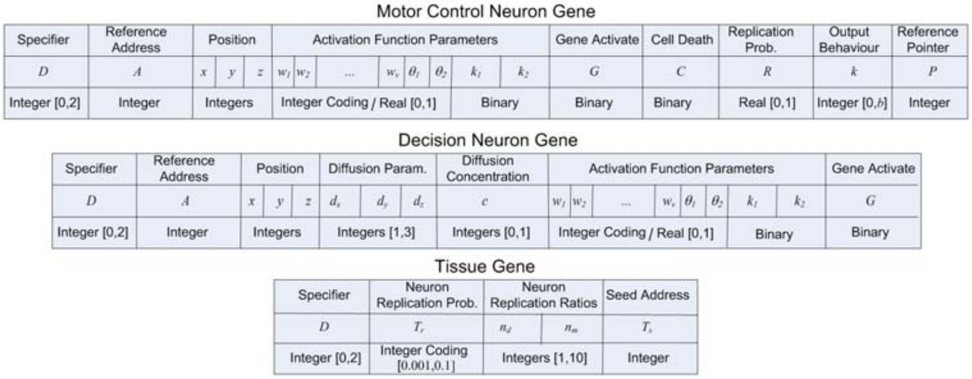


Fig. 5. A crossover operation between two ANT parents. 'Compatible' neuron genes are interchanged as shown resulting in two offspring.

IV. Sign Following Task

The evolutionary performance of the ANT framework is demonstrated in simulation and hardware for an unlabeled sign-following task. The workspace is modeled as a two-dimensional grid environment with one holonomic robot (equivalent to a Khepera$^{\text{TM}}$ robot, equipped with a gripper and camera) occupying four grid squares. For these tasks, the controller must possess several behaviours such as the ability to decipher signs relative to the robot's current frame of reference, to remember the current sign while looking for the next one, and to negotiate obstacles (Figure 6). Each sign is color-coded and represents a waypoint (posted in a fixed frame of reference) that gives direction using one of four cardinal points to the next waypoint leading ultimately to the goal location.

Mines undetectable by the robot are randomly laid throughout the floor except along the pathway. Once a robot encounters a mine, it remains disabled for the remainder of its lifetime. The sensory input map is shown in Table I



Motor Control Neuron Gene

Specifier	Reference Address	Position			Activation Function Parameters								Gene Activate	Cell Death	Replication Prob.	Output Behaviour	Reference Pointer
D	A	x	y	z	w_1	w_2	...	w_v	θ_1	θ_2	k_1	k_2	G	C	R	k	P
Integer [0,2]	Integer	Integers			Integer Coding / Real [0,1]						Binary		Binary	Binary	Real [0,1]	Integer [0,b]	Integer

Decision Neuron Gene

Specifier	Reference Address	Position			Diffusion Param.			Diffusion Concentration	Activation Function Parameters								Gene Activate
D	A	x	y	z	d_x	d_y	d_z	c	w_1	w_2	...	w_v	θ_1	θ_2	k_1	k_2	G
Integer [0,2]	Integer	Integers			Integers [1,3]			Integers [0,1]	Integer Coding / Real [0,1]						Binary		Binary

Tissue Gene

Specifier	Neuron Replication Prob.	Neuron Replication Ratios		Seed Address
D	T_r	n_d	n_m	T_s
Integer [0,2]	Integer Coding [0.001,0.1]	Integers [1,10]		Integer

Fig. 4. ANT gene map showing tissue, motor neuron and decision neuron genes.

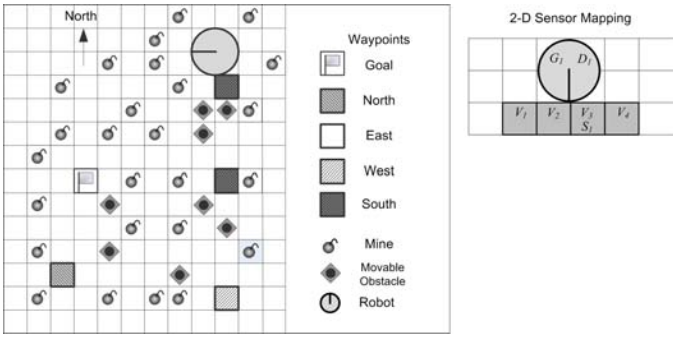


Fig. 6. (Left) 2D grid world model for the sign-following task. (Right) Input sensor mapping from the robot.

(see also Figure 6 right). The task has to be accomplished using a discrete set of basis behaviors specified in Table II. These behaviors are activated based on controller output, are preordered and all occur within a single timestep. The robot is initially positioned next to the first sign, but the initial heading is randomly set to one of the four cardinal directions. Since the robot can only detect signs placed in front, it needs to go into a 'sign searching' mode and perform a sequence of 'turn left' or 'turn right' behaviors to detect the first sign. Once the first sign is detected, the robot then needs to transition into a 'sign following' mode requiring one bit of memory.

Deciphering signs relative to the robot's current frame of reference makes these tasks particularly difficult given a fitness function that measures success in terms of reaching the goal location. So when the goal is reached $f_i = 1$; otherwise, $f_i = 0$. The simulated robot's world is a 20×20 grid with 80 uniformly distributed obstacles and 40 randomly distributed mines (except along the path to the goal). The fitness is averaged over 100 runs with different initial conditions, the elapsed time for each run being limited to $T = 100$ timesteps. For the EA experiments, the population consist of 100 individuals, using tournament selection, with tournament group size of 15 and crossover probability between parent of 70 %.

A. Results

The evolutionary performance of various control system architectures is compared for the sign-following task (Figure 7). For some comparison, a fixed-topology recurrent network with 9 hidden and 4 output neurons is also shown. (Although this is typical of the results obtained for such a network, we did not optimize its performance in any way.) Fixed-topology networks tend to have more 'active' synaptic connections present (all neurons are active) and thus more spurious neurons need to be dealt with simultaneously. The ANT topology with the coarse-coding neuro-regulatory functionality disabled (using modular activation function) shows nearly identical performance to NEAT [39] that uses variable length topology. Both approaches show better performance than the fixed topologies (including H-ESP [26]) but is not sufficient to complete the task for all the initial conditions tested. As we see from these results, the task is found to be intractable (due to performance stagnation) for these fixed and variable network topologies. ANT with access to memory variables successfully solves for all the initial conditions tested and outperforms the other controllers. ANT controller solutions (population best) evolved offline have been demonstrated on hardware using a holonomic LEGO® NXT (Figure 8). In the remaining sections we identify key attributes of the ANT architecture that enables it to solve the unlabeled sign-following task and where conventional neural networks fail.

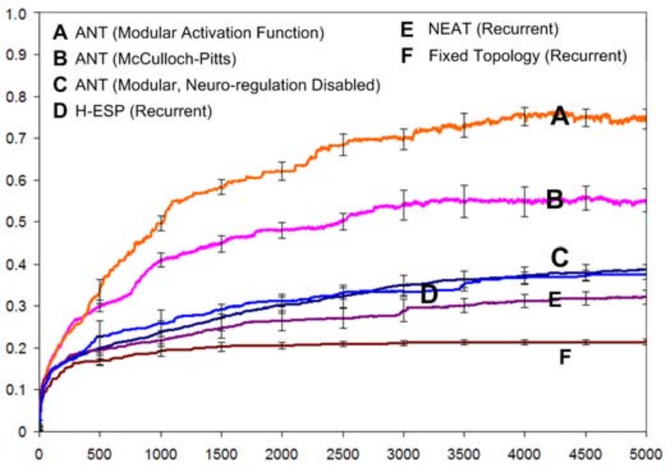


Fig. 7. Evolutionary performance comparison for the sign-following task (population best, averaged over 60 EA runs).

B. Self-Organized Task Decomposition

In this section we present evidence suggesting self-organized task decomposition occurs in the Artificial Neural

TABLE I
SENSOR INPUT FOR THE SIGN-FOLLOWING TASK.

Sensor Variables	Function	Description
$V_1 \ldots V_4$	Object detection	Robot, block, no obstacle
G_1	Gripper status	Holding block, no block
S_1	Sign detection	Red (North), pink (East), black (West), orange (South), green (Goal)
D_1	Heading	North, east, west, south

TABLE II
BASIS BEHAVIORS FOR THE SIGN-FOLLOWING TASK.

Order	Behavior	Description
1	Pick-Up/Put-Down	Pick up from V_2 or V_3 or put down obstacle on V_1 or V_4.
2	Move forward	Move one square forward
3	Turn right	Turn 90° right
4	Turn left	Turn 90° left
5, 7, 9, 11	Bit set	Set memory bit i to 1, $i = 1 \cdots 4$
6, 8, 10, 12	Bit clear	Set memory bit i to 0, $i = 1 \cdots 4$

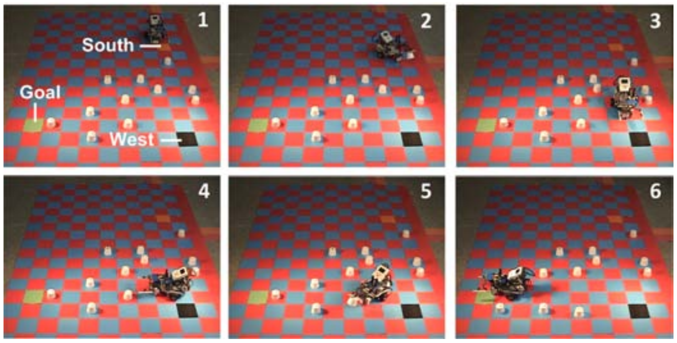


Fig. 8. Snapshots of a holonomic LEGO® Mindstorms$^{\text{TM}}$ NXT robot performing the sign following task using an ANT controller. Frame 1 shows the robot interpreting an orange sign (coding for South) and turns right (robot frame of reference) once having sensed the black sign in Frame 4 (West). Frame 6 shows the robot stopping at the green sign (goal).

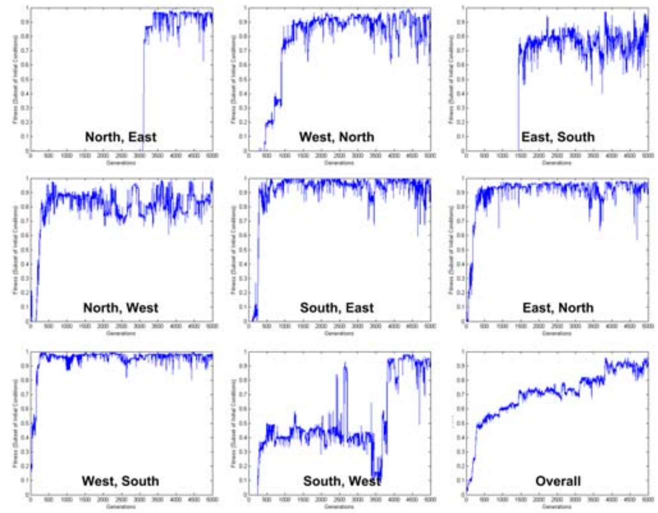


Fig. 9. Fitness of population best for various L-shaped pathways for the sign following task and (bottom right) overall fitness.

Tissue controllers. With human devised task decomposition, the first step to devise the overall goal of the task and then figure out the necessary subtasks, followed by further partitioning of the subtasks until all components of the task are identified and readily solvable. With self-organized task decomposition, there is no supervisor to break up the goal function into the necessary subtasks. Instead this process occurs through trial and error as we show here. For the sign-following task we only specify a global fitness function that doesn't bias for a particular solution strategy. At the end of each trial, the robot is at the goal location or not resulting in either a fitness of 1 or 0 respectively. However, a large number of board layouts (100) are used for fitness evaluation. Some of these board layouts are simpler to navigate (due to their shorter distances) than others. The simpler board layouts maybe solved by a set of memorized movements without needing to interpret the signs. Hence, memorized solutions that result in a robot reaching the goal location for a few scenarios have a fitness advantage over those that don't and hence first dominate the population in the early phase of the evolutionary search.

Individuals with these active tissue segments that randomly succeed have an advantage over those that do nothing or are stuck. Now for the next generation of individuals to have an advantage over their parents strategy/solution they will need to succeed for more cases. Hence 'building blocks' of strategies are developed that permit the controllers to compete and better solve specific scenarios (Figure 9) giving itself a fitness advantage over those that don't. These better strategies are discovered through trial and error. But with a range of board layouts with a varying degrees of difficulty, these slight changes in strategy can be measured in fitness performance. The emergence of the capability to solve specific scenarios is suggested to be a punctuated process, in which building blocks of functionality contribute towards rapid fitness improvement of an individual at certain subtasks. Collectively, the contribution of these rapid punctuated improvements at specialized tasks appears as a gradual improvement of the overall fitness of the population (Figure 9, lower right).

With the creation of these building blocks, evolutionary pressure directs improvement in the fitness performance and leads to expanded task decomposition, by handling more subtasks (Figure 9). This process continues as the population converges to a desired solution. The resultant task decomposition strategy is not as intuitive or simple to decipher in part because there exists redundant (neutral) components that have no appreciable effect on solving the task.

Analysis of the activity of an ANT controller that solves the sign-following task with a fitness of 0.94 is shown in Figure 10. Figure 11 shows the board layout. The robot is initially located beside a sign but is facing South. In this scenario the robot is unable to detect the sign beside it due to the limited range of the vision sensors (see layout in Figure 6).

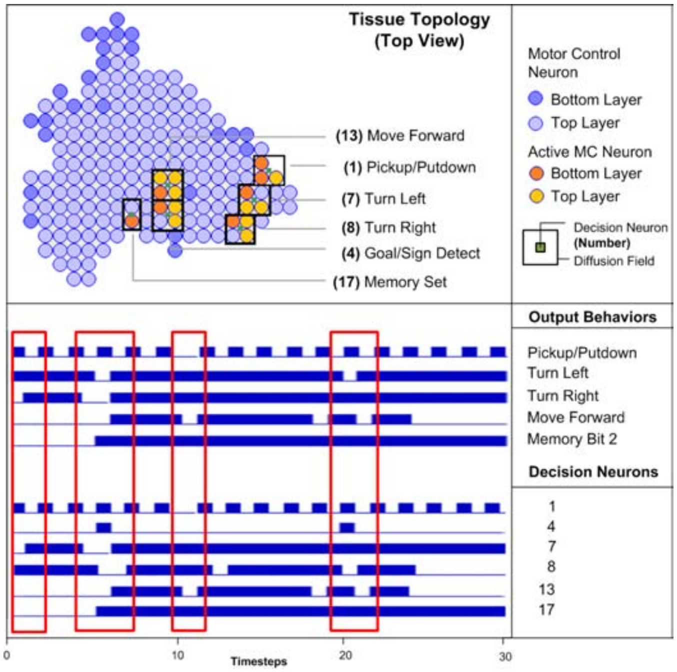


Fig. 10. Tissue Topology and neuronal activity of a select number of decision neurons. These decision neurons in turn select (excite into operation) motor control neurons within its diffusion field.

The ANT controller performs a series of right turns, until the robot is able to detect the first sign. A memory bit (memory bit 2) is set to 1, indicating the controller has switched from 'sign searching' mode to 'sign following' mode. Once the first sign is detected, the robot moves forward until the second sign is detected and proceeds to turn left. Along the way the controller picks up and puts down obstacles from the pathway towards the sides. The robot then proceeds to move forward until the goal sign is reached and stops. The output behaviors trigged by the ANT controller is found to have corresponding decision neurons as shown in Figure 10. The decision neurons and corresponding motor neurons that are activated for each subtask, such as interpreting signs, picking/putting down obstacles, switching from sign-searching to sign-following are distributed within the tissue. These centers of activity appear to be independent, decentralized modules that activate or are inhibited depending on sensory cues.

V. Analysis

In this section we analyze the capabilities of the ANT controller to understand the effects of its topology on task performance.

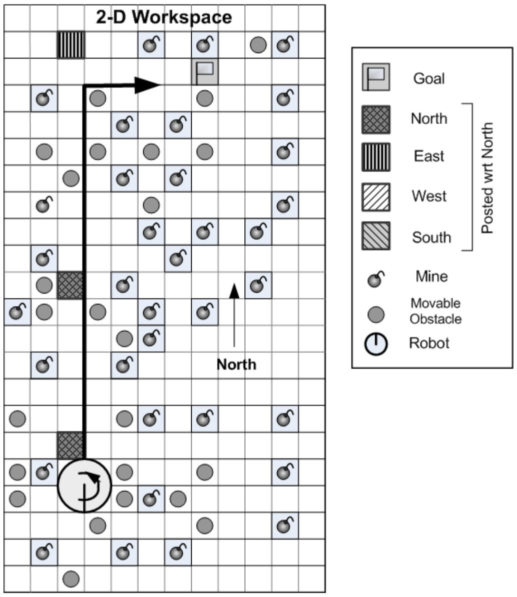


Fig. 11. Experiment layout for the sign following task showing trajectory of the robot from starting position to the goal location.

A. *Effect of Neural Regulation and Coarseness*

Figure 12 shows the evolutionary performance of the ANT topology (for the sign following task), along with ablated topologies for a range of coarseness (diameter of chemical concentration field). The intention in this set of experiments is to determine the role of both coarseness and superpositioning of coarse chemical fields in the overall performance of ANT. It should be noted that both of these features affect the regulatory functionality within ANT and is thus indicative of the role of neural regulation in evolution.

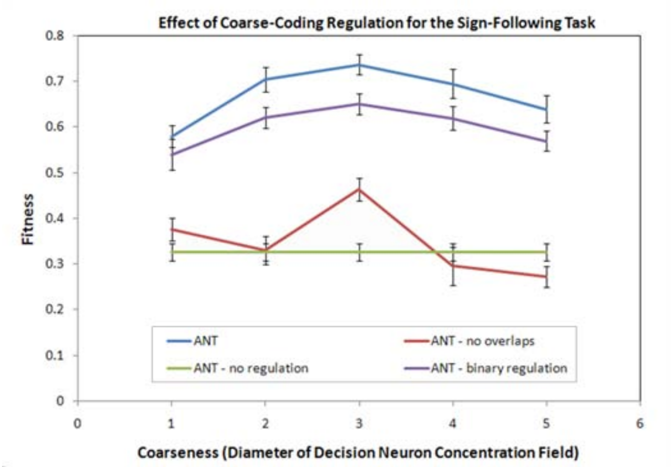


Fig. 12. Effect of coarse-coding neuro-regulation for the sign-following task (population best, averaged over 60 EA runs, 5000 generations). Error bars indicate standard deviation.

What is evident is that when the decision neurons are always active (neural regulation turned off), the controllers show a substantial drop in evolutionary performance compared to ANT with neural regulation turned on and this appears invariant to the coarseness parameter. None of the controllers evolved a solution with fitness of 1.0 when neural regulation is turned off and all the neurons remain active after development. This is equivalent to a fully expressed, variable-length topology and hence lacks any mechanism to selectively shut-off components of the phenotype during its lifetime. Next, we compare the performance with decision neurons able to perform neural regulation, but by disabling neurons selected from multiple overlapping concentration fields. This is to ascertain the role of coarse coding in the regulatory process. It is expected that with increased coarseness, more overlaps of concentration field would occur, given fixed tissue topology and thus more neurons will be inhibited. At the other extreme, with fine concentration fields, chance of overlap will be less but selection of neurons will be disparately distributed, hence it will be improbable to select groups of contiguous/interconnected neurons. For this optimum level of coarseness, while coarse coding interactions are inhibited, coarse selection fields will nevertheless select local clumps of interconnected neurons.

Significantly improved performance is evident when no restrictions are imposed on the overlapping concentration fields. For binary regulation, concentration of neurotransmitter chemicals has an upper bound thresholded to 1. This would result in selection of more neuron ensembles, as the threshold would result in a greater number of ties for the maximum concentration. As with other population coding methods, this approach offers greater redundancy; the information is distributed among multiple neurons but individual selection fields are more prone to alteration/damage because the concentration fields cannot be further strengthened beyond the upper bound.

B. *Effect of Number of Synaptic Interconnections*

In this section, we compare the effect of number of synaptic connections on learning tractability. Instead of having 9 neurons from layers $l-1$ connect to l as described in Section III.A, we vary this number from 4 to 36 to understand the effect of number and layout of synaptic connections between neurons. With increased number of synaptic connections, the number of weights associated with each neuron is greater. This in turn increases the weight search space. Figure 13 shows the fitness performance of population best at the end of 5000 generations of evolutionary training.

We note the fitness is at a maximum when there is 9 interconnected neurons between layers and the coarseness diameter is 3 (covering 9 neurons). This trend is observed when varying the number of interconnected neurons between 4 and 36 and further observed in Figure 13, where we note the maximum appears to be constant for increased number of interconnections and matching coarseness. The fitness is maximized when the coarseness diameter matches the number of interconnected neurons and drops steadily when too many or too little are selected. This result indicates that the topology benefits from selecting motor control neurons in a neighborhood that are feeding into a single output neuron. When the selection field of a decision neuron is smaller than the range of the interconnections, then fewer neurons connecting to the same output neuron is selected simultaneously. This indicates that the process is benefitting from parallel selection of neurons. In other words, the topologies benefit from selecting multiple motor control neurons in a neighborhood that feed into a single output. With a coarse selection field, not all of these neurons will behave as intended in accomplishing certain subtasks. However this wide selection still permits a few that may have the intended behavior in being activated, providing an advantage to individuals with these behaviours over those that don't. Similarly when too many motor control output neurons are selected, the system is less precise in controlling (masking) the flow of signals from one layer to another due to spatial crosstalk [5]. Under this condition, the system requires one or more decision neuron concentration fields to be activated in unison to select for a finer volume. What is interesting to observe is that the system benefits from 'over' selecting motor control neurons than under selecting. This again supports our observation that selecting as many motor neurons in parallel from similar signal sources, helps the tissue effectively evaluate the behaviors of groups of motor neurons.

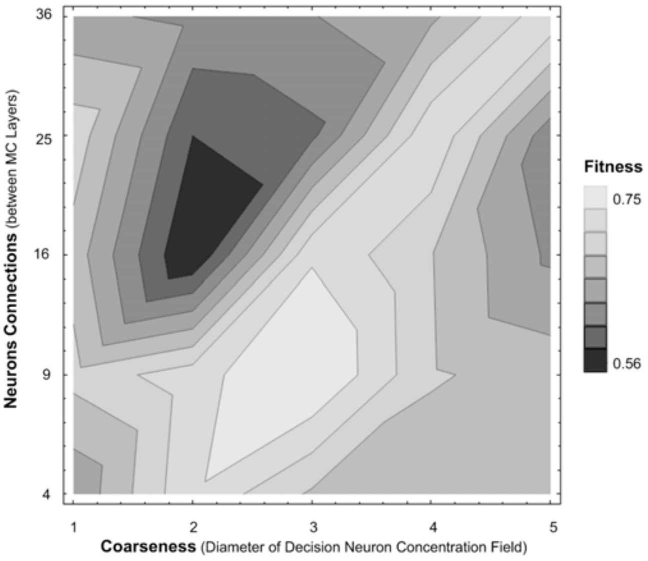


Fig. 13. Effect of coarseness (diameter of decision neuron concentration field) and interconnections between motor control neurons layers on fitness for the sign following task (population best, averaged over 60 EA runs, 5000 generations).

The graph in Figure 14 shows that ANT is largely unaffected by an increase in the number of synaptic connections. This result is expected because the coarse-coding regulatory approach can effectively mask the noisy signals and transmit the required signals cues. These results suggests that with effective masking done through coarse-coding regulation, the evolutionary search process can find incrementally better phenotypes towards a solution.

C. *Effect of Number of Neurons Initialized*

The effect of initial number of neurons on fitness is important consideration when training neural networks because earlier works in the field have cited the need to start with a few neurons that *complexifies* through evolutionary selection [25]. Having to start evolutionary training of a neural network controller with a few neurons for a new task is disadvantageous.

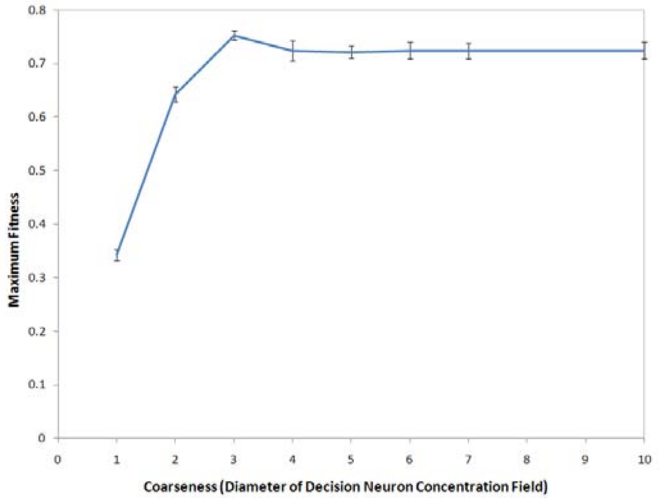


Fig. 14. Effect of coarseness (diameter of decision neuron concentration field) on maximum fitness found (population best, averaged over 60 EA runs, 5000 generations). Error bars indicate standard deviation.

Innovative traits acquired from training for one task cannot possibly be reused for training on another similar task thus limiting the full benefits of a modular system. Ideally, components in a modular system can be reused and hence be available for use on different tasks. Figure 15 shows the performance of the full ANT architecture in comparison with the coarse-coding regulatory functionality turned off for different dimensions of decision neuron concentration fields. The results suggest ANT has good scalability. Initializing the tissues with a large number of neurons as shown does not reduce its learning tractability. Initializing with too few neurons reduces fitness, because it takes longer for the slow evolutionary directed development process to add the necessary number of neurons. Hence, over specifying the number of neurons during initialization maybe beneficial, ignoring software implementation constraints on a conventional computer such as memory usage and CPU cycles.

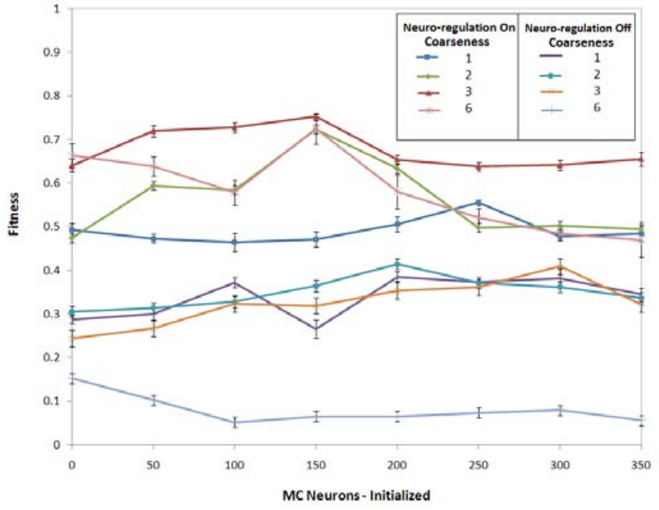


Fig. 15. Effect of number of motor control neurons initialized on fitness (population best, averaged over 60 EA runs). Error bars indicate standard deviation.

Furthermore, turning off the coarse-coding regulatory functionality, we find the controllers stagnate at a low fitness value, unable to evolve a desired solution. Increasing the number of interconnected neurons further decreases fitness at which the population stagnates. With increased number of neurons specified, it is expected to take longer to tune weights of all the neurons that will be functionally active and 'mask out' the noisy neurons. With too many neurons to start with, the system is unable to find incrementally better phenotypes towards a desired solution, thus leading to stagnation. However, the coarse coding regulatory approach can effectively mask and turn off a large number of neurons in one step. This enables the system to functionally work like a small topology, but with the advantage of using a reservoir of unexpressed neurons.

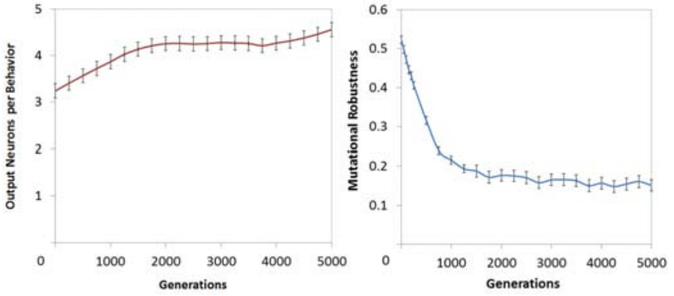


Fig. 16. Number of active output neurons per basis behavior (left) and robustness (R_p) $p = 0.004$ (right) during evolutionary training for the sign-following task.

The advantage with ANT is the topology can better reconfigure itself compared to regular neural networks. But how does ANT manage to increase its robustness to mutational damage? Figure 16 (left) shows the number of output neurons per behavior for the best individual of an ANT population averaged over 60 runs and Figure 16 (right) shows the robustness of the best individual according to the following metric:

$$R_p = \frac{f - \hat{f}_p}{f} \tag{5}$$

where R_p is the robustness of an individual to point mutations with probability p, f is fitness of an individual and $\hat{f}_p$ is the average fitness of an individual after being applied with point mutations with probability p. The results show that R_p decreases for ANT controllers as its converging towards a solution indicating as suggested earlier the controllers are more plastic to mutational change during early phases of evolution and increase in robustness as its converging towards a solution. Figure 16 (left) show a steady increase in the number of neurons per output behaviors. This suggests there is increased degeneracy as the populations converges towards a solution. Comparing the correlation of the two curves in Figure 16, we obtain -0.95, indicating a strong negative correlation between the robustness metric and number of output neurons. This suggests that ANT controllers manage to increase their robustness to mutational damage by incorporating redundant or degenerate functionality into their structure.

VI. Discussion

We present and analyze ANT, a neural network topology and regulatory scheme that addresses one of the critical challenges in machine learning, namely learning intractability when faced with high dimensional search spaces. The technique finds solutions to the sign following task, where other standard neural network techniques are shown to fail. ANT is able to solve the difficult sign following task due to its neuro-regulatory approach, that enables dynamic activation and inhibition of networks neurons within the tissue through coarse-coding. This functionality enables the tissue to be plastic (exploratory) during early phases of an evolutionary search to find innovative traits. Once innovative traits are acquired, these traits are effectively retained and protected from deleterious mutations through increased degeneracy and redundancy of functional modules. Regular neural networks are found unable to explore and retain innovative traits during early phases of evolutionary search, resulting in premature search stagnation resulting in intractability.

In a scenario where there is little domain knowledge of the task at hand and where the objective of the task is clear and quantifiable, then learning strategies that creatively solves a task through task decomposition is beneficial. By creative, we mean methods that defy intuition and are not conventional. For the sign following task, we find conventional fixed and variable topology neural network controllers are unable to solve this task. This critical component of ANT that enables solving these harder tasks is through this coarse-coding regulatory approach. This coarse-coding regulatory approach performs self organized task decomposition as we have shown here by segmenting the task based on sensory inputs and outputs to different modules. These modules evolve necessary traits to solve subtasks consisting of specific scenarios at a time and that are combined providing an incremental improvement in overall fitness through a 'building-block' approach (Section V.B).

This technique facilitates finding novel solutions to a given task and that is not explicitly specified by the fitness function. For the sign following task, this includes functionality such as switching from 'sign-searching' to 'sign-following', correctly interpreting the signs from the robot-frame of reference, moving obstacles out of the way and correctly stopping at the goal sign. Key areas within an ANT solution controller that accomplishes this functionality have been identified and the capabilities have been validated on hardware.

The critical component of ANT is the coarse-coding regulatory scheme. The scheme enables one to many selection of neurons in the tissue where one bit flip in a decision neuron gene can select or disable many motor neurons. This we hypothesize facilitates effective exploration (through numerous activation and inhibitions) of different segments of the tissue. This is beneficial in the early stages of evolution, where the system affords more risk and can attempt to find strategies that are promising based on a fitness advantage. This approach facilities both competition and cooperation. Where multiple selection fields overlap and select the same network of neurons, it is cooperation, where multiple selection fields select different networks of neurons with different functionality results in competition. This competition-cooperation dynamics has different purposes at different times in the training process. In the early stages, competition is beneficial to sample and find effective solution strategies. Later, when the population converges towards a desired solution strategy, cooperation appears to dominate, helping to keep the individuals robust to mutation.

It's not enough that a network topology scheme solves a desired task, it needs to be scalable and extensible to facilitate learning a whole variety of tasks, to have practical merit. What is promising about the ANT approach is that it shows good scalability, based on our studies, with increased number of neurons and interconnections. Increasing the number of interconnections between neurons and selection field in tandem, we find that the maximum fitness remains nearly constant. This capability has additional practical merits, because the approach is not sensitive to parameters such as interconnection and number of neurons initialized.

We note that the ANT controllers not only evolve towards solving a given task, but acquire traits that increase its robustness to mutations. The controllers evolve effective means of acquiring innovative traits, followed by methods to retain and protect them. This as mentioned earlier occurs in a incremental, 'building-blocks' approach based on evidence of task decomposition in punctuated steps (Section V.B), increased robustness and degeneracy of the tissue controllers (Section VI.D). The system benefits from increased plasticity to mutational changes early in the evolutionary search process, but can reconfigure itself to increase its robustness. Once key building blocks are acquired, the population needs to effectively retain and protect them from deleterious mutations. Our results suggests that the controllers accomplish this by evolving degenerate or redundant modules that can increase their robustness to deleterious mutations.

VII. Conclusion

The "Artificial Neural Tissue" (ANT) framework has been presented and successfully applied to the sign-following task. ANT is shown to find solutions to the sign-following task, where other standard neural network approaches are shown to fail. ANT produces solutions to the harder sign-following task through emergent (self-organized) task decomposition that requires only a global fitness function, a set generic basis behaviors and inputs be specified. ANT is able to solve this hard task due to its ability to dynamically activate and inhibit networks of neurons within a tissue using a coarse coding approach. The controllers evolve to solve subsets of a given task through trial and error in punctuated steps. In combination, the controllers solve the task in an incremental manner. This ability to dynamically activate and inhibit networks of neurons enables the controllers to be (plastic) exploratory during early phases of evolution to effectively acquire innovative traits. Once the necessary traits are acquired, the controllers evolve to protect these innovations from deleterious mutations. Analysis of the ANT solution controllers indicate areas of specialization develop, with functionality distributed within a tissue. The controllers show excellent scalability and robustness properties both in terms of number of neurons and interconnections. The ANT approach is being extended to other challenging areas such as the control of multi-robot systems.

A. Acknowledgements

The authors wish to thank the reviewers for their helpful comments and suggestions. This work was funded by the Natural Science and Engineering Research Council of Canada and is gratefully acknowledged.

References

[1] G. Cybenko, "Approximation by Superposition of a Sigmoidal Function," *Mathematical Control Signal Systems*, vol. 2, pp. 303–314, 1989.

[2] S. Nolfi and D. Floreano, *Evolutionary Robotics*. Cambridge: MIT Press, 2000.

[3] M. Dorigo and M. Colombetti, "Robot Shaping: Developing Autonomous Agents through Learning," *Artificial Intelligence*, vol. 71, pp. 321–370, 1994.

[4] P. Stone and M. Veloso, "Layered Learning," in *Proc. of the 11th European Conference on Machine Learning*, 2000, pp. 369–381.

[5] R. Jacobs, M. Jordan and A. Barto, "Task decomposition through Competition in a Modular Connectionist Architecture," *Cognitive Science*, vol. 15, pp. 219–250, 1991.

[6] J. Thangavelautham and G.M.T. D'Eleuterio, "A Coarse-Coding Framework for a Gene-Regulatory-Based Artificial Neural Tissue," in *Proc. of the 8th European Conference on Artificial Life*, 2005, pp. 67–77.

[7] J. Thangavelautham, *A Regulatory Theory of Cortical Organization and Its Applications to Robotics*. Ph.D. Thesis, Institute for Aerospace Studies, University of Toronto, Canada, 2008.

[8] J.S. Albus, "A Theory of Cerebellar Function," *Mathematical Biosciences*, vol. 10, no. 1-2, pp. 25–61, 1971.

[9] G. Hinton, "Shape Representation in Parallel Systems," in *Proc. of the 7th International Joint Conference on Artificial Intelligence*, 1981, pp. 1088–1096.

[10] J. Garthwaite, S. Charles and R. Chess-Williams, "Endothelium-derived relaxing factor release on activation of NMDA receptors suggests role as intercellular messenger in the brain," *Nature*, vol. 336, pp. 385-388, 1988.

[11] C. Hölscher, "Nitric oxide, the enigmatic neuronal messenger," *Trends in Neuroscience*, vol. 20, pp. 298–303, 1997.

[12] G. Hinton, "Product of Experts," in *Proc. of the 9th International Conference on Artificial Neural Networks* vol 1, 1999, pp. 1–6.

[13] Y. Liu, X. Yao and T. Higuchi, "Evolutionary Ensembles with Negative Correlation Learning," *IEEE Trans. on Evolutionary Computation*, vol. 4, pp. 380–387, Nov. 2000.

[14] T. Kohonen, *Self-Organizing Maps*. Berlin: Springer- Verlag, 1997.

[15] K. Tasdemir, P. Milenov, and B. Tapsall, "Topology-Based Hierarchical Clustering of Self-Organizing Maps," *IEEE Trans. on Neural Networks*, vol. 22, pp. 474-485, Mar. 2011.

[16] E.L. Rubio and E.J. Palomo, "Growing Hierarchical Probabilistic Self-Organizing Graphs," *IEEE Trans. on Neural Networks*, vol. 22, pp. 997-1008, July 2011.

[17] J. Walter and H. Ritter, "Rapid Learning with Parametrized Self-Organizing Maps," *Neurocomputing*, vol. 12, pp. 131–153, 1995.

[18] W. Maass, T. Natschlaeger and H. Markram. "Real-time computing without stable states," *Neural Computation*, vol. 14, no. 11, pp. 2531–2560, 2002.

[19] H. Jaeger. "The Echo State approach to analysing and training recurrent neural networks," German National Research Center for Information Technology, Tech. Rep. GMD 148, 2001.

[20] D. Verstraeten, B. Schrauwen, M. DHaene, and D. Stroobandt, "An experimental unification of reservoir computing methods," *Neural Networks*, vol. 20, pp. 391-403, 2007.

[21] E. Goodman and D. Ventura, "Effectively using recurrently connected spiking neural networks," in *Proc. of the IEEE International Joint Conference on Neural Networks*, vol. 3, 2005, pp. 1542-1547.

[22] D. Norton and D. Ventura, "Preparing More Effective Liquid State Machines Using Hebbian Learning," in *Proc. of the IEEE International Joint Conference on Neural Networks*, 2006, pp. 8359-8364.

[23] B.J. Grzyb, E. Chinellato, G.M. Wojcik and A.W. Kaminsi, "Facial Expression Recognition based on Liquid State Machines Built of Alternative Neuron Models," in *Proc. of the International Conference on Neural Networks*, 2009, pp. 1011–1017.

[24] T. Yamazaki and S. Tanaka, "The Cerebellum as Liquid State Machine," *Neural Networks*, vol. 20, pp. 290–297, 2007.

[25] K. Stanley and R. Miikkulainen, "Continual Coevolution through Complexification," in *Proc. of the Genetic and Evolutionary Computation Conference*, 2002.

[26] F. Gomez and J. Schmidhuber, "CoEvolving Recurrent Neurons Learn Deep Memory POMDPs," in *Proc. of the Genetic and Evolutionary Computation Conference*, 2005, pp. 491–498.

[27] P. Eggenberger, G. Gomez and R. Pfeifer, "Evolving the morphology of a neural net for controlling a foveating retina," *Artificial Life*, vol. 8, pp. 243-251, 2002.

[28] J. Astor and C. Adami, "A developmental model for the evolution of artificial neural networks," *Artificial Life*, vol. 6, no. 3, pp. 189–218, 2000.

[29] J. Hawkins and S. Blakeslee, *On Intelligence*. New York: Times Books, 2004.

[30] S. Ohno, *Evolution by Gene Duplication*. Berlin: Springer-Verlag, 1970.

[31] K. Stanley and R. Miikkulainen, "A Taxanomy for Aritificial Embrogeny," in *Proc. of Artificial Life IX*, 2003, pp. 93–130.

[32] F. Gruau, D. Whitley and L. Pyeatt, "A Comparison between Cellular Encoding and Direct Encoding for Genetic Neural Networks," in *Proc. of the First Conference on Genetic Programming*, 1996, pp. 81–89.

[33] H. Kitano, "Designing Neural Networks using Genetic Algorithms with Graph Generation Systems," *Complex Systems*, vol. 4, no. 4, pp. 461–476, 1990.

[34] K. Sims, "Evolving 3D Morphology and Behavior by Competition," in *Proc. of Artificial Life IV*, 1994, pp. 28–39.

[35] G. Hornby and J. Pollack, "Creating High-Level Components with a Generative Representation for Body-Brain Evolution," in *Proc. of Artificial Life VIII*, 2002.

[36] F. Dellaert and R. Beer, "Towards an Evolvable Model of Development for Autonomous Agent Synthesis," in *Proc. of Artificial Life IV*, 1994, pp. 246–257.

[37] D. Federici and K. Downing. "Evolution and Development of a Multicellular Organism," *Artificial Life*, vol. 12, pp. 381–409, 2006.

[38] J. Zhang, "Evolution by Gene Duplication: An Update," *Trends in Ecology and Evolution*, vol. 18, no. 6, pp. 192-198, 2003.

[39] K. Stanley and R. Miikkulainen, "Evolving Neural Networks through Augmenting Topologies," *Evolutionary Computation*, vol. 10, no. 2, pp. 99–127, 2002.

Jekanthan Thangavelautham received his Bachelor's degree in Engineering Science (Aerospace) from the University of Toronto in 2002 and his Ph.D. degree from the University of Toronto Institute for Aerospace Studies in 2008. His research interests include robotics, neural networks, artificial life, machine learning, and mobile fuel cell power systems. He is presently a Postdoctoral Associate at the Field and Space Robotics Laboratory at M.I.T.

Gabriele M.T. DEleuterio was born in Teramo, Italy. He received his B.A.Sc. degree in Engineering Science (Aerospace) from the University of Toronto in 1980 and his M.A.Sc. and Ph.D. degrees from the University of Toronto Institute for Aerospace Studies in 1982 and 1985. His current research interests include space robotics and biologically inspired approaches to robotic control. He is presently a Professor at the University of Toronto Institute for Aerospace Studies.