Application of an Artificial Neural Tissue Controller to Multirobot Lunar ISRU Operations

Jekanthan Thangavelautham[1], Alexander Smith[1]
Dale Boucher[2], Jim Richard[3], Gabriele M. T. D'Eleuterio[1]

[1]Institute for Aerospace Studies, University of Toronto, 4925 Dufferin St., Toronto, ON M3H 5T6
[2]Northern Centre for Advanced Technology, 1400 Barrydowne Rd., Sudbury, ON P3A 3V8
[3]Electric Vehicle Controllers Ltd, 2200 ValleyView Rd., Val Caron, ON P3N 1L1
416-667-7731, thanga@ecf.utoronto.ca

Abstract. Automation of mining and resource utilization processes on the Moon with teams of autonomous robots holds considerable promise for establishing a lunar base. We present an Artificial Neural Tissue (ANT) architecture as a control system for autonomous multirobot tasks. An Artificial Neural Tissue (ANT) approach requires much less human supervision and pre-programmed human expertise than previous techniques. Only a single global fitness function and a set of allowable basis behaviors need be specified. An evolutionary (Darwinian) selection process is used to train controllers for the task at hand in simulation and is verified on hardware. This process results in the emergence of novel functionality through the task decomposition of mission goals. ANT based controllers are shown to exhibit *self-organization*, employ *stigmergy* (communication mediated through the environment) and make use of *templates* (unlabeled environmental cues). With lunar in-situ resource utilization (ISRU) efforts in mind, ANT controllers have been tested on a multirobot resource gathering task in which teams of robots with no explicit supervision can successfully avoid obstacles, explore terrain, locate resource material and collect it in a designated area by using a light beacon for reference and interpreting unlabeled perimeter markings.

Keywords: ISRU, collective robotics, neural networks, evolutionary algorithms, developmental systems.
PACS: 45.40.Ln, 84.35.+i, 07.05.Mh.

INTRODUCTION

It has been argued that mining and resource utilization can benefit from the autonomous operation of teams of robots on the lunar surface. Indeed, robotic preparation for a human habitat on the Moon is very likely an imperative. Concerns for health and safety limit the productivity of human astronauts. Teams of autonomous robots could work continuously in hazardous conditions, making them very appealing for lunar mining and in-situ resource utilization (ISRU) tasks. One has only to observe the effectiveness of a colony of ants excavating tunnels or termites building towering cathedral mounds with internal heating and cooling shafts (Bristow and Holt, 1997) to see why approaches to such tasks in robotics are often biologically inspired and multiagent oriented (Melhuish, Welsby and Edwards 1999).

Although the teleoperation of lunar rovers is possible from Earth and has been demonstrated successfully with the Lunakhod 1 and 2 rover missions, operator fatigue is of concern (Miller and Machulis, 2005) especially when coordinating actions with teams of robots over extended missions. In addition, these systems must be robust to communication interruptions and bandwidth and latency issues. An alternative approach is the deployment of autonomous systems that require limited human control. There currently exist two major approaches to developing autonomous control systems: human knowledge/model-based controllers and systems based on machine learning techniques. Human knowledge/model-based behaviour control strategies rely on human input in the form of ad-hoc control rules, task-specific assumptions, and human experience and knowledge. In contrast, machine learning systems of the type examined here perform task decomposition through 'emergent' self-organized behavior. In lunar and planetary environments, task-specific assumptions may not always be valid in-situ and may require reassessment during a mission. There is also a growing necessity for the development of generic teams of 'utility' robots that can facilitate in-situ resource utilization and perform specific tasks that may never have been envisioned during mission planning and modeling stages.

CP880, *Space Technology and Applications International Forum—STAIF 2007,* edited by M. S. El-Genk

The approach outlined in this paper involves the application of a machine learning approach called the Artificial Neural Tissue framework (Thangavelautham and D'Eleuterio, 2005) to solve a multirobot resource gathering and berm formation task. With minimal task-specific assumptions and limited supervision, ANT can produce controllers which exploit *templates* (unlabeled environmental cues), *stigmergy* (indirect communication mediated through the environment), and self-organization. Since little preprogrammed knowledge is given, ANT may discover novel solutions that would otherwise be overlooked by a human supervisor.

ANT is particularly advantageous for multirobot tasks in which some global behavior must be achieved without a centralized controller. Even though each individual has no 'blueprint' of the overall task and must operate with only local sensor data, an ensemble of robots is able to complete a mission requiring a global consensus. The desired global behavior 'emerges' from the local interactions of individual robots. Designing controllers of this sort by hand can be very difficult because the process by which local behaviors work to form a global consensus can be difficult to understand and even counterintuitive. Previous work in the field such as (Mataric *et al.*, 1995; Parker, Zhang and Kube, 2003; Wilson *et al.*, 2004; Werfel, Yam and Nagpal, 2005) rely on task-specific human knowledge to develop simple 'if-then' rules or equivalent coordination behaviors to solve multirobot tasks. In contrast, the approach outlined here is a generic framework that could be applied to any number of non-Markovian robotic control tasks, and is not specific to the collective-robotics domain. It has already been shown that ANT can produce emergent controller solutions for a multirobot tiling pattern formation task, a single-robot phototaxis task and an unlabeled sign following task (Thangavelautham and D'Eleuterio, 2005). In this paper, we compare the training performance of fixed-topology versus ANT-based neural network controllers for a multirobot resource gathering task.

BACKGROUND

Collective robotic tasks typically employ some of the same mechanisms used by social insects. These include the use of templates, stigmergy, and self-organization. *Templates* are environmental features perceptible to the individuals within the collective (Bonabeau, Dorigo and Thereaulaz, 1999). *Stigmergy* is a form of indirect communication mediated through the environment (Grasse, 1959). *Self-organization* describes how local or microscopic behaviors give rise to a macroscopic structure in systems which are not in equilibrium (Bonabeau *et al.*, 1997). These methods have been applied to various collective robotics tasks. However, most existing techniques have relied on prior task-specific knowledge to create stochastic or rule-based controllers. Our approach is evolutionary in nature and 'learns' to take advantage of these techniques without explicit human input.

In insect colonies, templates may be a natural phenomenon, or they may be created by the colonies themselves. They may include temperature, humidity, chemical, or light gradients. In robotic applications, template-based approaches include the use of light fields to direct the creation of circular (Stewart and Russell, 2003) and linear walls (Wawerla, Sukhame and Mataric, 2002) and planar annulus structures (Wilson *et al.*, 2004). Stewart and Russell (2004) have used spatiotemporal varying templates, in which one member of the robot collective varies the template thereby altering the global consensus that results to construct a 'loose wall' structure. The use of active rather than passive construction objects, although not biologically plausible, has also been applied to building user-defined colored block structures (Werfel, Yam and Nagpal, 2005).

Stigmergy is an implicit form of communication and involves individuals modifying the environment and in turn alters the perception and behavior of other individuals that encounter the environmental changes. Stigmergy has been used extensively in collective-robotic construction tasks, including blind bull dozing (Parker, Zhang and Kube, 2003), box pushing (Mataric *et al.*, 1995), heap formation (Beckers, Holland and Deneubourg, 1994) and tiling pattern formation (Thangavelautham, Barfoot and D'Eleuterio, 2003).

Self-organization describes the process by which a global consensus emerges from a set of local behaviors (Bonabeau *et al.*, 1997). With self-organized systems, no individual possesses a knowledge of the overall environment or the end goal. Individuals merely react to local sensor data. A global solution arises from the interactions of a colony of individuals. Our approach in this paper uses a Darwinian selection process to evolve robotic controllers for performing a desired task. The evolved controllers make use of templates and stigmergy in order to achieve the level of self-organization necessary to achieve the global goals.

The collective robotic works cited earlier excluding (Thangavelautham, Barfoot and D'Eleuterio, 2003) rely on either user-defined, deterministic 'if-then' rules, or on stochastic behaviors. In both cases, designing these controllers is an ad-hoc procedure that relies on the experimenter's knowledge of the task at hand. However, the global effect of local interactions is often difficult to determine, and the specific interactions required to achieve a global consensus may even be counterintuitive. Thus, at this stage of the field's development at least, designing successful controllers by hand is a process of trial and error.

One approach to reducing the amount of trial and error done by hand is to encode controllers as behavioral look-up tables, and allow a genetic algorithm to evolve the values in the table. This approach was demonstrated for collective heap formation (Barfoot and D'Eleuterio, 1999) and 2×2 tiling pattern formation tasks (Thangavelautham Barfoot and D'Eleuterio, 2003). The limitations to this approach are poor sensor scalability and lack of generalization. An increased number of sensors leads to a combinatorial explosion in the size of the look-up table, resulting in premature search stagnation (the 'bootstrap' problem). As an action must be encoded for each combination of sensor inputs, the controller does not generalize from one state to another one with similar inputs. Neural network controllers can often overcome this second limitation by effectively implementing a compressed representation of the problem space. A neural network controller was able to solve the harder 3×3 tiling formation task (Thangavelautham and D'Eleuterio, 2004). Other fixed-topology neural controller approaches have been used to build walls, corridors and briar patches (Crabbe and Dyer, 1999).

Fixed-topology neural networks present an additional problem: The size and structure of the network must be fixed ahead of time. Inappropriate choices may lead to a network that is unable to solve the problem. The ANT framework is able to overcome this problem. This variable-length neurocontroller model allows for the generalization of sensory input, for improved scalability over fixed-network topologies, and for both stochastic and deterministic arbitration schemes. For the 3×3 tiling pattern formation (collective task) (Thangavelautham and D'Eleuterio, 2005) and for the resource gathering tasks presented here, ANT shows improved performance over fixed-topology neural networks.

ARTIFICIAL NEURAL TISSUE MODEL

The ANT architecture (Figure 1) presented in this paper consists of a developmental program, encoded in the 'genome,' that constructs a three-dimensional neural tissue and associated regulatory functionality. The tissue consists of two types of neural units, decision neurons and motor-control neurons, or simply motor neurons. Regulation is performed by the decision neurons that dynamically exhibit or inhibit motor-control neurons within the tissue based on a coarse-coding framework. Let us discuss the computational mechanism of the tissue first and then outline the process by which the tissue is created.

Computation

The motor neurons is assumed be spheres arranged in a regular rectangular lattice in which the neuron N_λ occupies the position $\lambda = (l, m, n) \in \mathbb{I}^3$ (sphere centered within cube). The state s of the neuron is binary, *i.e.*, $s_\lambda \in S = \{0,1\}$. Each neuron N_λ nominally receives inputs from neurons $N_{\boldsymbol{\kappa}}$ where $\boldsymbol{\kappa} \in \Uparrow(\boldsymbol{\lambda})$, the nominal input set. Here it shall be assumed that these nominal inputs are the 3×3 neurons centered one layer below N_λ; in other terms, $\Uparrow(\boldsymbol{\lambda}) = \{(i, j, k) \mid i = l-1, l, l+1; j = m-1, m, m+1; k = n-1\}$. (As will be explained presently, however, it shall not be assumed that all the neurons are active all the time.) The activation function of each neuron is taken from among four possible threshold functions of the weighted input σ:

$$\psi_{\text{down}}(\sigma) = \begin{cases} 0, & \text{if } \sigma \ge \theta_1 \\ 1, & \text{otherwise} \end{cases}, \quad \psi_{\text{up}}(\sigma) = \begin{cases} 0, & \text{if } \sigma \le \theta_2 \\ 1, & \text{otherwise} \end{cases} \tag{1}$$

$$\psi_{\text{ditch}}(\sigma) = \begin{cases} 0, & \min(\theta_1,\theta_2) \le \sigma < \max(\theta_1,\theta_2) \\ 1, & \text{otherwise} \end{cases}, \quad \psi_{\text{mound}}(\sigma) = \begin{cases} 0, & \text{if } \sigma \le \min(\theta_1,\theta_2) \text{ or } \sigma > \max(\theta_1,\theta_2) \\ 1, & \text{otherwise} \end{cases}.$$

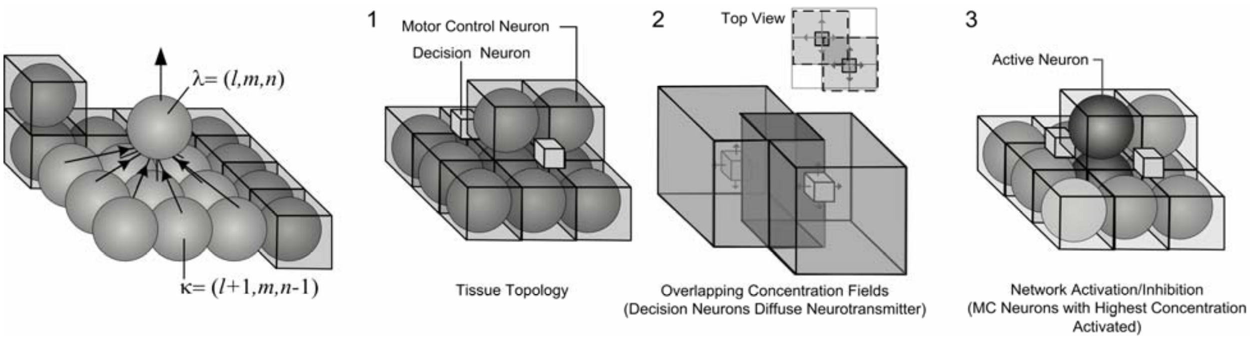
The weighted input σ_λ for neuron N_λ is nominally taken as:

$$\sigma_{\boldsymbol{\lambda}} = \frac{\sum_{\boldsymbol{\kappa} \in \Uparrow(\boldsymbol{\lambda})} w_{\boldsymbol{\lambda}}^{\boldsymbol{\kappa}} s_{\boldsymbol{\kappa}}}{\sum_{\boldsymbol{\kappa} \in \Uparrow(\boldsymbol{\lambda})} s_{\boldsymbol{\kappa}}}, \tag{2}$$

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 $N_{\boldsymbol{\lambda}}$. These threshold functions maybe summarized in a single analytical expression as:

$$\psi = (1-k_1)[(1-k_2)\psi_{down} + k_2\psi_{up}] + k_1[(1-k_2)\psi_{ditch} + k_2\psi_{mound}], \tag{3}$$

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 $\theta_1, \theta_2 \in \mathbb{R}$.



(a) Synaptic Connections. **(b)** Coarse Coding.

FIGURE 1. (a) Synaptic Connections between Motor Neurons from Layer *l*+1 to *l*. (b) Activated Decision Neurons Diffuse Neurotransmitter Concentration Field Resulting in Activation of Motor Control Neurons with Highest Activation Concentration.

It may appear that ψ_{down} and ψ_{up} are mutually redundant as one type can be obtained from the other by reversing the signs on all the weights. However, retaining both increases diversity in the evolution because a single 2-bit 'gene' is required to encode the threshold function and only one mutation suffices to convert ψ_{down} into ψ_{up} or *vice versa* as opposed to changing the sign of every weight. The sensor data are represented by the activation of the sensor input neurons $N_{\boldsymbol{\alpha}_i},\ i = 1\ldots m,$ summarized as $A = \{s_{\boldsymbol{\alpha}_1}, s_{\boldsymbol{\alpha}_2} \ldots s_{\boldsymbol{\alpha}_m}\}$. Similarly, the output of the network is represented by the activation of the output neurons $N_{\boldsymbol{\omega}_j},\ j = 1\ldots n,$ summarized as $\Omega = \{s_{\boldsymbol{\omega}_1^1}, s_{\boldsymbol{\omega}_2^2} \ldots s_{\boldsymbol{\omega}_n^b}\}$, where $k = 1\ldots b$ specifies the output behavior. Each output neurons 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 involve 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:

$$p(k) = \sum_{s_{\Diamond}=1,\, j=1}^{n_k} \frac{s_{\boldsymbol{\omega}_j^k}}{n_k}, \tag{4}$$

and n_k is the number of output neurons connected to output behavior *k* 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 (as shown in Table 1 for the task).

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. Neurons may also cancel out each other such one output commanding a forward step while another commands a backward step. Finally, not all behaviors need be encoded in the neural tissue. This is left to the evolutionary process.

The Decision Neuron

The coarse-coding nature of the artificial neural tissue is provided by the decision neurons. Decision neurons can be thought of as rectangular structures occupying nodes in the lattice as established by the evolutionary process (Figure 1). The effect of these neurons is to excite or inhibit the motor control neurons (shown as spheres). 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.

In ANT, the state of a decision neuron T_μ located at $\boldsymbol{\mu}$ is binary and determined by one of the same activation functions (2) that also serve the motor control neurons. The inputs to T_μ are all the input sensor neurons N_α, *i.e.*, $s_\mu = \psi_\mu(s_{\alpha_1} \ldots s_{\alpha_m})$ where $\sigma_\mu = \sum_\alpha v_\alpha^\mu s_\alpha \Big/ \sum_\alpha s_\alpha$ and v_α^μ are the weights. The decision neuron is dormant if $s_\mu = 0$ and releases a virtual chemical of uniform concentration c_μ over a prescribed field of influence if $s_\mu = 1$. A motor control neuron will be excited into operation if the total concentration of the chemical from all influential decision neurons reaches a predefined critical level. 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 (1) are over only the set $\overline{\Uparrow}(\lambda) \subseteq \Uparrow(\lambda)$ of active inputs to N_λ. 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 $\pm d_\mu^r$, $r = 1, 2, 3$, from $\boldsymbol{\mu}$ is the three perpendicular directions. These three dimensions along with $\boldsymbol{\mu}$ and c_μ, the concentration level of the virtual chemical emitted by T_μ are encoded in the genome.

Evolution and Development

A population of ANT controllers is evolved in an artificial Darwinian manner (Holland, 1975). The 'genome' for a controller contains a 'gene' for each cell with a specifier *D* (see Figure 3) that is 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 (see Figure 2). 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 gene. When repressed, a descriptor protein of the gene content is not created. Otherwise, the constructor protein 'grows' the tissue in which each cell is located relative to a specified 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 for a cell, by committing suicide, still occupies 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.

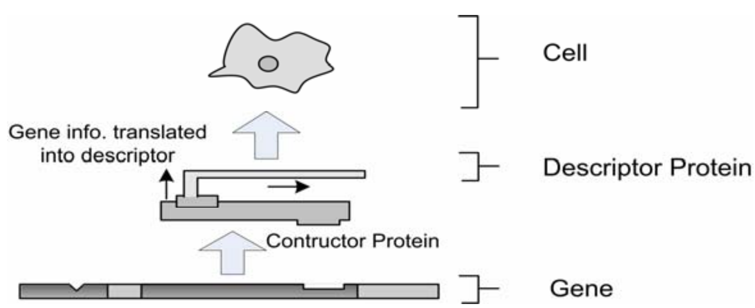


FIGURE 2. Genes Are 'Read' by Constructor Protein that Transcribe the Information into a Descriptor Protein which Is Used to Construct a Cell. When a Gene Is Repressed, the Constructor Protein Is Prevented from Reading the Gene Contents.

In turn mutation (manipulation of gene parameters with 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 neuron replication probability parameter. Cell division requires a parent cell (selected with highest replication probability relative to the rest of the cells within the tissue) and involves 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 with a uniform random distribution. The cell type of each new cell is determined based on the ratio of motor control to decision neurons specified in the tissue gene. The new cell can be located in one of 6 neighboring locations (top, bottom, north, south, east, west) sharing a common side with the parent and is not occupied by another cell.

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			Real [0,1]						Integers [0,1]		Binary	Binary	Real [0,1]	Integer [0,b]	Integer

(a) Motor Control 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_2	...	w_v	θ_1	θ_2	k_1	k_2	G
Integer [0,2]	Integer	Integers			Integers [1,3]			Integers [0,1]	Real [0,1]					Integers [0,1]		Binary

(b) Decision Neuron Gene.

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

(c) Tissue Gene.

FIGURE 3. Gene Descriptions for Elements within the ANT Genome.

SIMULATION EXPERIMENT SETUP

The resource gathering and berm formation task is intended to demonstrate the feasibility of team of robots in gathering surface resources (such as titanium oxide or helium-3 on the lunar surface) into a designated dumping area, where the resources will be further processed. It could be argued that *emergent* task decomposition may be necessary to accomplish the task given a global fitness function. A layout of the simulation experiment area used for training is shown in Figure 4. The experiment region or workspace is modeled as a two-dimensional grid environment with the size of each square in the grid being just able to accommodate one robot. For this task, the controller need to accomplish a number of subtasks including gather resource material, avoiding workspace perimeter, avoiding colliding into other robots, and collecting resources into a berm at the designated location. (In the present experiment, a berm is simply a mound of the resource material.) The berm location has perimeter markings on the floor and a light beacon mounted nearby. The two colors on the border are intended to allow the controller to determine whether the robot is inside or outside the berm location. Though solutions can be found without the light beacon, its presence improves the efficiency of the solutions found as it allows the robots to track the target location from a distance instead of randomly searching the workspace for the perimeter. The global fitness function for the task measures the amount of resource material accumulated in the designated location within a finite number of time steps; in this case, $T = 300$ timesteps.

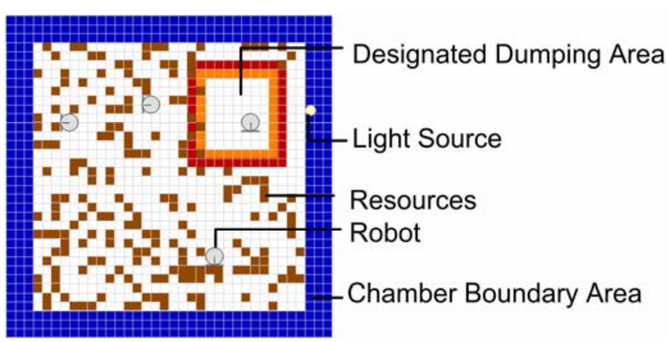


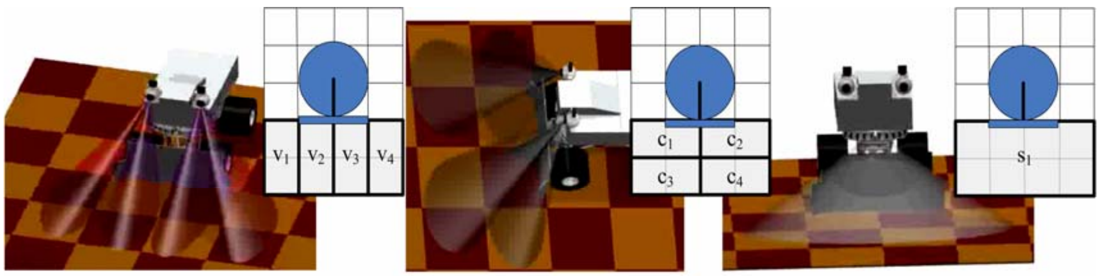
FIGURE 4. 2-D Grid World Model of the Experiment Chamber.

TABLE 1. Sensor Inputs.

Variables	Function	States
$V_1 ... V_4$	Resource Detection	Resource, No Resource
$C_1 .. C_4$	Colour Template Detection	Blue, Red, Orange, Floor
S_1, S_2	Front/Rear Obstacle Detection	Obstacle, No Obstacle
LP_1	Light Position	Left, Right, Center, Not Visible
LD_1	Light Range	0-10 (distance to light)

TABLE 2. Basis Behaviours.

Order	Behaviour	Description
1	Dump Resource	Move one grid square backward and turn 90 ^{o} left, when resource inside shovel.
2	Move Forwards	Move one grid square forwards.
3	Turn Right	Turn 90 ^{o} right.
4	Turn Left	Turn 90 ^{o} left.
5, 7, 9, 11	Bit Set	Set memory bit i to 1, i=1…4
6, 8, 10, 12	Bit Clear	Set memory bit i to 0, i=1…4



(a) Resources. **(b)** Color Template. **(c)** Obstacle Avoidance.

FIGURE 5. Robot Input Sensor Mapping, Simulation Model Shown Inset.

For this task, inputs to the ANT controller are shown in Table 1 (right). The robots have access to a pair of webcams and a set of sonar sensors. All raw input data are discretized. The sonar sensors are used to determine the values of S_1 and S_2. One of the cameras is used to detect resource material (green packing peanuts) and colored floor templates (see Figure 5). The other camera is used to track the light beacon. In order to identify peanuts and colored floor templates, a naïve Bayes classifier is used to perform colour recognition (Hastie, Tibshirani and Friedman, 2001). Simple feature-detection heuristics are used to determine the values of $V_1 . . . V_4$ and $C_{1...}C_4$ based on the grid locations shown. For detection of the light beacon, the shutter speed and gain are adjusted to ensure that the light source is visible while other background features are underexposed. The position of the light LP_1 is determined based on the pan angle of the camera. The distance to the light source LD_1 is estimated based on its size in the image. The robots also have access to four memory bits, which can be manipulated using some of the basis behaviours. Table 1 (left) lists the basis behaviors the robot can perform. These behaviors are activated based on an arbitration scheme mentioned previously and all occur within a single timestep. Darwinian selection is performed based on the fitness value of each controller averaged over 100 different initial conditions. The Evolutionary Algorithm (EA) population size for the experiments is P = 100, crossover probability p_c = 0.7, mutation probability p_m = 0.025 and tournament size of 0.06 P (for Tournament selection).

RESULTS AND DISCUSSION

Figure 6 shows the fitness (population best) of the overall system evaluated at each generation of the artificial evolutionary process. The performance of a fixed-topology, fully connected network with 12 hidden and output neurons is also shown in Figure 7. While this is not intended as some benchmark network, in a fixed-topology network there tends to be more 'active' synaptic connections present (since all neurons are active) and thus it takes longer for each neuron to tune these connections for all sensory inputs. In an ANT-based architecture, the network is dynamically formed based on set of sensory input facilitating smaller network specialized for specific sensory inputs. The average fitness comparison with ANT controllers shows that the performance increases with the number of robots. With an increased number of robots, each robot has a smaller area to cover in trying to gather and dump resources. Some of the emergent solutions indicate that the individual robots all figure out how to dump nearby resources into the designated berm area; however, not all robots deliver resource all the way to the dumping area every time (Figure 8). Instead, the robots learn to pass the resource material from one individual to another during an encounter; a 'bucket brigade' (Figure 8 (d)–(e)). This technique improves the overall efficiency of system as less time is spent traveling to and from the dumping area. Since the robots cannot explicitly communicate with one another, these encounters happen by chance rather than through preplanning. As with other multiagent systems, communication between robots occurs through manipulation of the environment in the form of stigmergy.

While the robot controllers can detect and home in on a light beacon, this source of navigation is not always used. Although not necessary, the light beacon helps in navigation by allowing the robots to locate dumping area. It is surprising that the fitness of the system is not substantially affected by the light source turned off. However, when the light source is on, the controllers do make use of it to home in on the dumping area even though this does not appear to provide any noticeable performance advantages (Figure 7). Once the robots are facing away from the light source, the light beacon sensor is out of view and hence in the 'Not Visible' state.

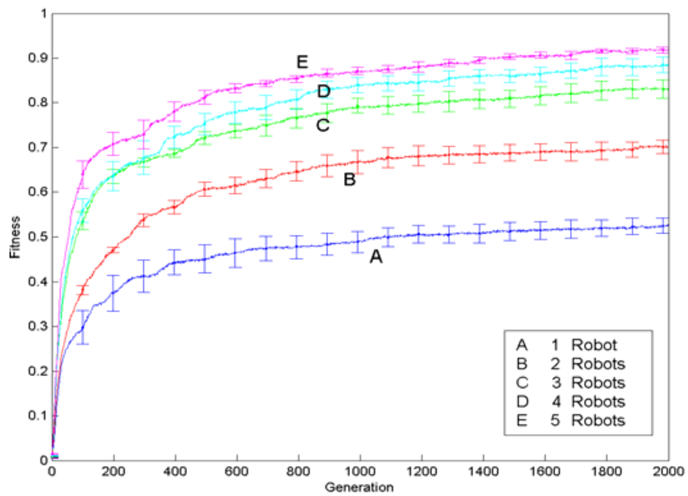


FIGURE 6. Evolutionary Performance Comparison of ANT Based Solutions for between 1 and 5 Robots.

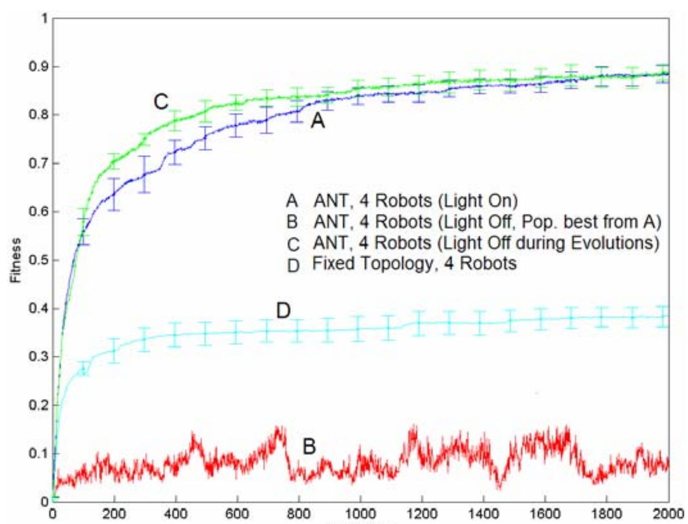
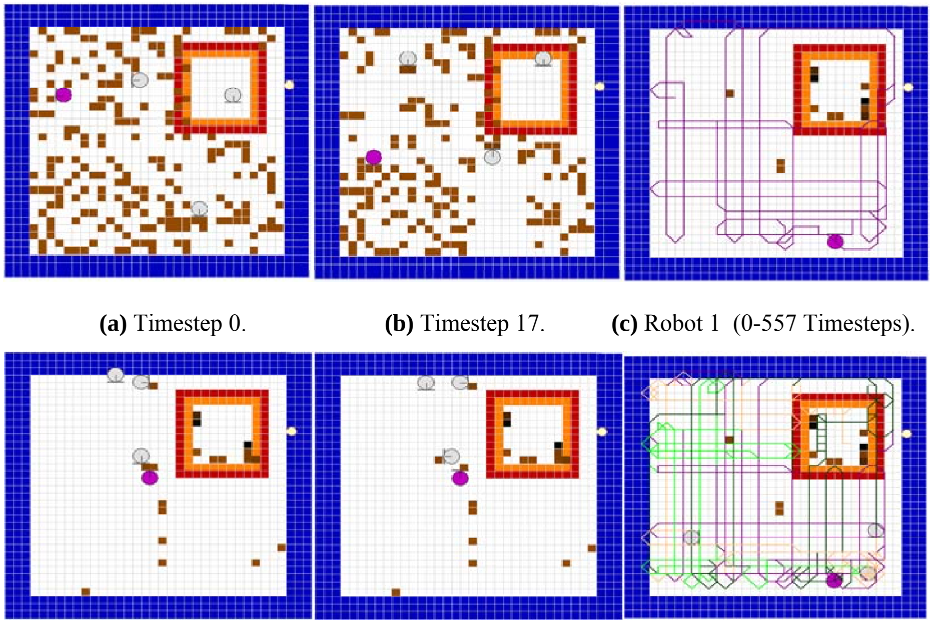


FIGURE 7. Evolutionary Performance Comparison with 4 Robots for Fixed Topology Case and Light Beacon Off.



(a) Timestep 0. **(b)** Timestep 17. **(c)** Robot 1 (0-557 Timesteps).

(d) Timestep 271. **(e)** Timestep 272. **(f)** All Robots (0-557 Timesteps).

FIGURE 8. (a)-(f) Snapshots and Robot Trajectories of a Task Simulation (4 robots).

(a) Frame 1. **(b)** Frame 2. **(c)** Frame 3. **(d)** Frame 4.

FIGURE 9. Movie frames of Argo Rover with ANT Controller Homing onto Designated Area and Dumping Resources.

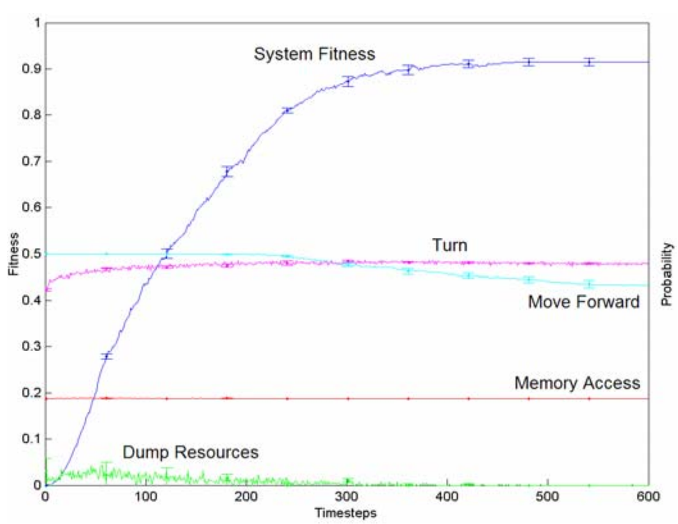


FIGURE 10. System Activity for the Resource Gathering Task (4 Robots, Population Best Solution).

This particular set-up would necessitate reliance on other environment cues to find the dumping area. For example, the controllers are found to follow the blue-shaded boundary until they approach the dumping area, at which time the robot can see and use the light source (Figure 8 (c),(f)). This is in part due to the dumping area being next to the boundary region in our experiments. Thus, where possible, the controllers perform 'sensor fusion,' making use of data from multiple sensors to perform navigation to and from the dumping area. Some of the evolved behaviours particularly, 'dump resources', which depends on both visual cues and the light beacon has been verified on hardware (see Figure 9). In these experiments, the robots have no information about how much time is available; hence, the system cannot greedily accumulate resource materials without periodically dumping the material at the designated area. This explains why we see a steady increase in the amount of resource material gathered over time (see Figure 10). Fitness increases as more robots are added. With fewer robots moving around based on a deterministic rule set, there is an increased possibility of being unable to visit the entire workspace within a finite number of time steps since the robots settle into cyclical loops once there is no more resource material along their trajectories (Figure 8 (f)). With an increased number of robots, these trajectories tend to become chaotic because the trajectories overlap and result in robot-robot encounters, which results in the robots turning away to avoid collision. These changes in trajectory enable the robots to cover an area that may not have been previously visited, thus improving the overall system performance. However, the simulations indicate that the point of diminishing returns is eventually reached. Beyond this point, additional robots have a minimal effect on the performance of the solution.

CONCLUSIONS

A developmental Artificial Neural Tissue (ANT) architecture has been successfully applied to a multirobot resource gathering and berm formation task in support of lunar in-situ resource utilization efforts. ANT controllers require only a global fitness function that merely measures the performance of the controller for a given task and a generic set of basis behaviors. Since little preprogrammed knowledge is given, an ANT architecture may permit novel solutions that might otherwise be overlooked by a human supervisor. ANT controllers are shown to exploit a number of mechanisms known to be used in multiagent systems in unsupervised manner, including environmental templates, stigmergy and self-organization. By exploiting these mechanisms the controllers exhibited novel functionality including use of 'bucket brigades' and various homing behaviors using sensor fusion.

ACKNOWLEDGMENTS

This work was partially funded by the Natural Sciences and Engineering Research Council of Canada. We also wish to thank the reviewers for their helpful comments.

REFERENCES

Barfoot, T. D., D'Eleuterio G.M.T., "An Evolutionary Approach to Multiagent Heap Formation," in proceedings of *IEEE Congress on Evolutionary Computation (CEC)*, edited by Angeline et al., IEEE, Washington DC, 1999, pp. 427-435.

Beckers, R., O. E. Holland, and J. L. Deneubourg, "From Local actions to Global Tasks: Stigmergy and Collective Robots," in *Artificial Life IV*, edited R. Brooks, P. Maes, MIT Press, Cambridge, MA, 1994, pp. 181-189.

Bonabeau, E., et al., "Self-organization in social insects," *Trends in Ecology and Evolution*, **12**, 188-193, (1997).

Bonabeau,E., Dorigo, M., and Theraulaz G., *Swarm Intelligence: From Natural to Artificial Systems*, Oxford University Press, New York, NY, 1999, pp. 183-205.

Bristow, K. L. and Holt, J. A. "Can Termites Create Local Energy Sinks to Regulate Mound Temperature?" *Journal of Thermal Biology*, **12**, 19-21, (1997).

Crabbe, F., Dyer, M., "Second-order networks for wall-building agents," in proceedings of *IEEE International Joint Conference on Neural Networks*, IEEE, Washington DC, 1999, pp. 2178– 2183.

Grasse, P., "La reconstruction du nid les coordinations interindividuelles; la theorie de stigmergie,'' *Insectes Sociaux*, **35**, 41–84 (1959).

Hastie, T., Tibshirani, R., Friedman, R., *The Elements of Statistical Learning*, Springer, New York, NY, 2001, pp. 184-185.

Holland, J., *Adaptation in Natural and Artificial Systems*, University of Michigan Press, Ann Arbor, MI, 1975, pp. 89-118.

Mataric, M. et al. "Cooperative Multi-Robot Box Pushing," in proceedings of *IEEE International Conference on Intelligent Robots and Systems*, IEEE, Washington DC, 1995, pp 556-561.

Melhuish, C., Welsby, J., Edwards, C., *Using Templates for Defensive Wall Building with Autonomous Mobile Ant-Like Robots*, Manchester Tech. Report/UMCS-99-3-1, Univ. of Manchester, Manchester, UK, 1999.

Miller, D., Machulis, K., "Visual Aids for Lunar Rover Tele-operation," in proceedings of *8th International Symposium on Artificial Intelligence, Robotics and Automation in Space,* edited by R. Battrick, ESA Publishing, Noordwijk, Netherlands 2005.

Parker, C., Zhang, H., and Kube, R., "Blind Bulldozing: Multiple Robot Nest Construction," in proceedings of *IEEE International Conference on Intelligent Robots and Systems,* IEEE, Washington DC, 2003, pp. 2010- 2015.

Stewart, R., Russell, A., "Emergent structures built by a minimalist autonomous robot using a swarm-inspired template mechanism," in proceedings of *The First Australian Conference on ALife (ACAL2003)*, Springer, Berlin, 2003, pp. 216-230.

Stewart, R., Russell, A., "Building a loose wall structure with a robotic swarm using a spatio-temporal varying template," in proceedings of *IEEE/RSJ International Conference on Intelligent Robots and Systems,* IEEE, Washington DC, 2004, pp. 712-716.

Thangavelautham, J., D'Eleuterio G. M. T., "A Neuroevolutionary Approach to Emergent Task Decomposition" in proceedings of *8th Parallel Problem Solving from Nature Conference*, edited by X. Yao et al., Springer, Berlin, 2004, pp. 991-1000.

Thangavelautham, J., Barfoot, T.D., D'Eleuterio G.M.T., "Coevolving Communication and Cooperation for Lattice Formation Tasks (Updated)," in *Advances In Artificial Life: Proceedings of the 7th European Conference on ALife*, edited by W. Banzhaf et al., Springer, Berlin, 2003, pp. 857-864.

Thangavelautham, J., D'Eleuterio G. M. T., "A Coarse-Coding Framework for a Gene-Regulatory-Based Artificial Neural Tissue" in *Advances In Artificial Life: Proceedings of the 8th European Conference on ALife*, edited by M. Capcarrere et al., Springer, Berlin, 2005, pp. 67-77.

Werfel, J., Yam Y., Nagpal, R., "Building Patterned Structures with Robot Swarms," in proceedings of *International Joint Conference on Artificial Intelligence*, edited by H.Arabnia, R.Joshua and Y.Mun, Morgan Kaufmann Publishers, San Francisco, CA 2005, pp. 1495-1502.

Wawerla, J., Sukhatme, G., Mataric, M., "Collective construction with multiple robots," in proceedings of *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'02)*, IEEE, Washington DC, 2002, pp. 2696-2701**.**

Wilson, M., Melhuish, C., Franks, A., Scholes, S., "Algorithms for building annular structures with minimalist robots inspired by brood sorting in ant colonies." *Autonomous Robots*, **17**, 115–136 (2004).