Temporally Adaptive Networks: Analysis of GasNet Robot Control Networks

Tom Smith*‡, Phil Husbands†, Andy Philippides* and Michael O'Shea*

Centre for Computational Neuroscience and Robotics (CCNR)
*School of Biological Sciences, †School of Cognitive and Computing Sciences
University of Sussex, Brighton, UK.
‡Email: toms@cogs.susx.ac.uk. Web: http://www.cogs.susx.ac.uk/users/toms/

Abstract

Identification of the fundamental properties necessary for the generation of adaptive behaviour is one of the primary goals for Artificial Life. In this paper, we address the related question of whether we can identify general useful properties of a given solution class. Such an approach provides a potentially scalable framework that may enable us to identify general properties of more complex adaptive systems. We develop a methodology based on analysis of successfully evolved solutions to an evolutionary robotics shape discrimination problem, allowing us to identify properties of solution classes that are potentially useful over a wider class of problems than the original task. We propose that the evolvability of the solution class is due to the fundamental property of *temporal adaptivity*.

Introduction

In a recent call to arms, Bedau et al. (2000) defined one of the primary goals for Artificial Life as identification of the minimal conditions for transition from specific to generic response systems. In other words, what are the fundamental properties necessary for a system to be adaptive?

Answering such a challenge is well beyond the scope of current understanding, however in this paper, we make some headway by investigating the solution classes underlying neural networks used as robot controllers. In particular, we address two questions. First, how can we evaluate the effectiveness of a given solution class on some problem? Second, can we evaluate how effective the solution class is likely to be on different problems? In other words, can we identify general useful properties of a solution class that may hold over a wider range of problems than simply the problem studied? Such an approach provides a potentially scalable framework that may enable us to identify general properties required for more complex adaptive systems.

We develop a methodology based on analysis of successfully evolved solutions to an evolutionary robotics shape discrimination problem, allowing us to identify properties of solution classes that are potentially useful over a wider class of problems than the original task. We go on to test these properties, through analysis of the evolved solutions in modified environments. We apply the techniques to a class of artificial neural network robot controllers inspired by work on diffusible neuromodulation, and argue that the evolvability of this biologically inspired GasNet solution class is due to the fundamental property of *temporal adaptivity*.

GasNet robot controllers

Artificial neural networks possess many useful properties for the generation of behaviour over time. However, the networks of simple units linked by weighted synapse connections typically used in ANN applications are extremely simplified abstractions of the real thing; nervous systems found in biology are far more complex, with a multitude of interacting processes operating over both space and time. The "GasNet" class of neural networks (Husbands et al., 1998) is explicitly inspired by the action of *diffusing gaseous neuromodulation* in real nervous systems, incorporating a mechanism based on the neuron-modulating properties of a diffusing signalling gas into a more standard sigmoid-unit neural network. In previous work the networks have been used in a variety of evolutionary robotics tasks, comparing the speeds of evolution for networks with and without (the "NoGas") the gas signalling mechanism active. In a variety of robotics tasks, GasNet controllers evolve significantly faster than networks without the gas signalling mechanism (see e.g. Husbands, 1998; Husbands *et al.*).

The GasNet and NoGas models

The GasNet is an arbitrarily recurrent ANN augmented with a gas concentration model, in which the instantaneous activation of a node is a function of both the inputs from connected nodes and the current concentration of gas(es) at the node. The basic network model consists of connected sigmoid transfer function nodes overlaid with a model of diffusing gaseous modulators; the gas does not alter the electrical activity in the network directly but rather acts by changing the gain of transfer function mapping between node input and output.

The network underlying the GasNet and NoGas models is a discrete time-step, recurrent neural network with a variable number of nodes. These nodes are connected by either excitatory (with a weight of +1) or inhibitory (with a weight of −1) links with the output O_i^t, of node i at time-step t determined by a continuous mapping from the sum of its inputs. This defines the basic NoGas class:

$$O_i^t \quad = \quad \tanh\left[K_i^t\left(\sum_{j\in C_i} w_{ji} O_j^{t-1} + I_i^t\right) + b_i\right] \qquad (1)$$

where C_i is the set of nodes with connections to node i with connection weights w_{ji}, O_j^{t-1} the output of node j on the previous time-step, I_i^t the external (sensory) input to node i at time t, and b_i a genetically set bias. Each node has a genetically set default transfer function parameter K_i^0, and for the NoGas class this transfer parameter is fixed over the operation of the network: $K_i^t = K_i^0\ \forall t$.

In the GasNet control system, in addition to this underlying network in which electrical signals flow between units, an abstract process loosely analogous to the gaseous diffusing modulators described above is at play. Some units can emit gases which diffuse and are capable of modulating the behaviour of other units through altering their transfer functions. As described below, this modulation changes the transfer parameter K_i^t as the network runs, thus the actual shape of the node's transfer function is altered via the gas modulation mechanism. This form of modulation allows a kind of plasticity in the network in which the intrinsic properties of units are changing during the operation of the network, that is during the robot controller lifetime. In the next sections we describe the gas diffusion and modulation mechanisms.

Gas diffusion in the networks

In order to incorporate the gas concentration model, the network is placed in a 2D plane, with node positions specified genetically. The GasNet diffusion model is controlled by two genetically specified parameters namely the radius of influence r around the emitting node, and the rate of build up and decay s. Spatially, the gas concentration varies as an inverse exponential of the distance from the emitting node with a spread governed by r, with the concentration set to zero for all distances greater than r (equation 2). The maximum concentration at the emitting node is one and the concentration builds up and decays from this value linearly as defined by equations (equation 3 and 4) at a rate determined by s. The governing equations are:

$$C(d,t) = \begin{cases} C_0\, T(t)\, e^{-(d/r)^2} & d < r \\ 0 & \text{else} \end{cases} \qquad (2)$$

$$T(t) = \begin{cases} H(\frac{t-t_e}{s}) & \text{emitting} \\ H(H(\frac{t_s-t_e}{s}) - H(\frac{t-t_s}{s})) & \text{not emitting} \end{cases} \qquad (3)$$

$$H(x) = \begin{cases} 0 & x \le 0 \\ x & 0 < x < 1 \\ 1 & \text{else} \end{cases} \qquad (4)$$

where C(d,t) is the concentration at a distance d from the emitting node at time t. t_e is the time at which emission was last turned on, t_s is the time at which emission was last turned off, and s (controlling the slope of the function T) is genetically determined for each node. To summarise, within a radius of r from the node, gas builds up (and decays) linearly to a maximum of $e^{-2d/r}$ in s time-steps. The total concentration at a node is then determined by summing the concentrations from all other emitting nodes (nodes are not affected by their own concentration, to avoid runaway positive feedback).

Modulation by the gases

There are two virtual gases in the network, gas 1 and gas 2, which increase and decrease K_i^t (see equation 1) respectively in a concentration dependent fashion. Both the type of gas emitted by a node and the conditions under which it emits are specified genetically. Nodes emit either gas 1, gas 2 or no gas, and emission occurs when either the electrical activation at a node exceeds 0.5, or the concentration of gas in the vicinity of the node exceeds 0.1. The concentration-dependent modulation is described by equations 5 to 8, with transfer parameters updated on every time-step as the network runs. Thus we have:

$$K_i^t = \mathbf{P}[D_i^t] \qquad (5)$$

$$\mathbf{P} = \{-4.0, -2.0, -1.0, -0.5, -0.25, -0.125, \\ 0.0, 0.125, 0.25, 0.5, 1.0, 2.0, 4.0\} \qquad (6)$$

$$D_i^t = f\left(D_i^0 + \frac{C_1^t}{C_0 K}(N - D_i^0) - \frac{C_2^t}{C_0 K} D_i^0\right) \qquad (7)$$

$$f(x) = \begin{cases} 0 & x \le 0 \\ \lfloor x \rfloor & 0 < x < N \\ N-1 & \text{else} \end{cases} \qquad (8)$$

where $\mathbf{P}[i]$ refers to the ith element of set $\mathbf{P}$, D_i^t is node i's index into the set $\mathbf{P}$ of possible discrete values K_i^t can assume, N is the number of elements in $\mathbf{P}$, D_i^0 is the genetically set default value for D_i, C_1^t is the concentration of gas 1 at node i on time-step t and C_2^t is the concentration of gas 2 at node i on time-step t. Both gas concentrations lie in the range $[0, 1]$.

Thus, the concentration of each gas is directly proportional to any change in D_i^t, with a corresponding change in K_i^t. Although the change in K_i^t is non-linear these values represent a smooth change in the slope of the transfer function. Since the transfer functions can change throughout the lifetime of the network, this system provides a form of network plasticity not seen in most other ANNs.

Visual shape discrimination

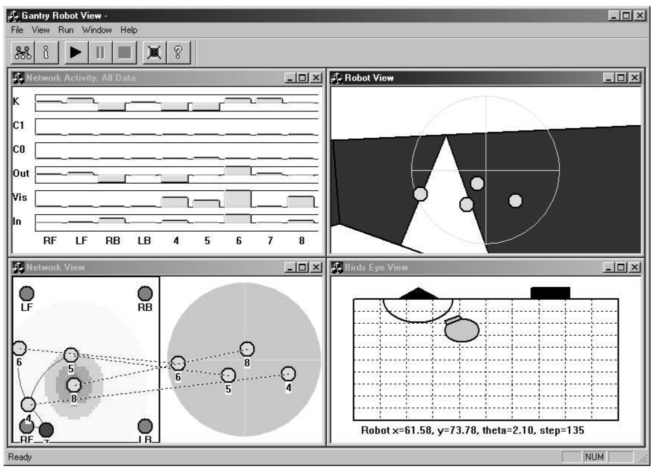


Figure 1: Screen shot of the simulated arena and robot. The bottom-right view shows the robot position in the arena with the triangle and square. Fitness is evaluated on how close the robot approaches the triangle. The top-right view shows what the robot 'sees', along with the pixel positions selected by evolution for visual input. The top-left view shows the current activity of all nodes in the neural network. The bottom-left view shows the robot control neural network.

The evolutionary task at hand is a visual shape discrimination task; starting from an arbitrary position and orientation in a black-walled arena, the robot must navigate to one shape (a white triangle) while ignoring the second shape (a white square), under the extremely variable lighting conditions produced by an array of spotlights flashing on and off at random time intervals. Both the robot control network, an arbitrarily recurrent ANN, and the robot sensor input morphology, i.e. the position of the input pixels on the visual array, were under evolutionary control. Fitness over a single trial was taken as the fraction of the starting distance moved towards the triangle by the end of the trial period, and the evaluated fitness was returned as the average over 16 trials of the controller from different initial conditions. Success in the task was taken as an evaluated fitness of 1.0 over thirty successive generations of the evolutionary algorithm. In the work reported here, fitness evaluations are carried out in a verified *minimal simulation* (Jakobi, 1998), see figure 1 for screen-shot of a fitness evaluation in simulation. A large number of controllers evolved in simulation have been tested on the real robot, with all controllers displaying similar robot trajectories and fitnesses to the simulated fitness evaluations (Husbands et al., 1998).

A distributed asynchronous updating evolutionary algorithm was used, with a $PopSize$ of 100 arranged on a 10×10 grid. Parents were chosen through rank-based roulette-wheel selection on the mating pool consisting of the 8 nearest neighbours to a randomly chosen grid-point. The child solution was a mutated copy of the parent (the mutation operator applied a $\mu = 4\%$ mutation probability per bit, and the same probability per genome of adding or deleting a network node. No crossover was used.) and placed back in the mating pool using inverse rank-based roulette-wheel selection. One generation was specified as $PopSize$ such breeding events.

Speed of evolution results

Over a large sample of evolutionary runs with GasNet and NoGas conditions, GasNet networks allowed to use the gaseous signalling mechanism reached success significantly faster than the NoGas networks. This speed difference was seen in several different evolutionary robotics scenarios (Husbands, 1998; Husbands et al., 1998, e.g.) and over several different mutation rates, e.g. see figure 2.

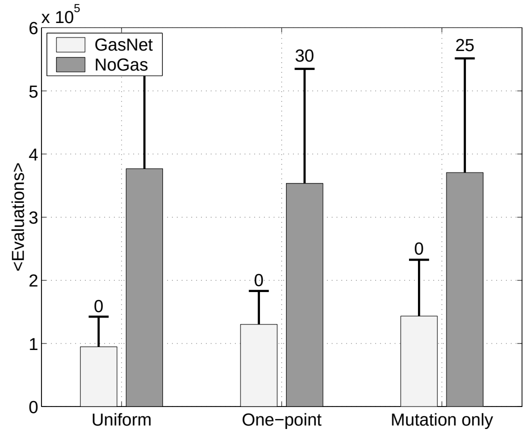
Analysis of evolved solutions

A number of studies have carried out analysis of evolved solutions to robot control problems, notably the work by Beer and colleagues (Beer and Gallagher, 1992; Beer et al., 1999; Chiel et al., 1999, e.g). However, the emphasis has typically been on developing methods for analysis, or characterising the type of solutions generated by evolution. The literature is not well developed in terms of analysing evolved solutions *in order to test the suitability of the underlying solution class*. That is the approach we use in this paper.

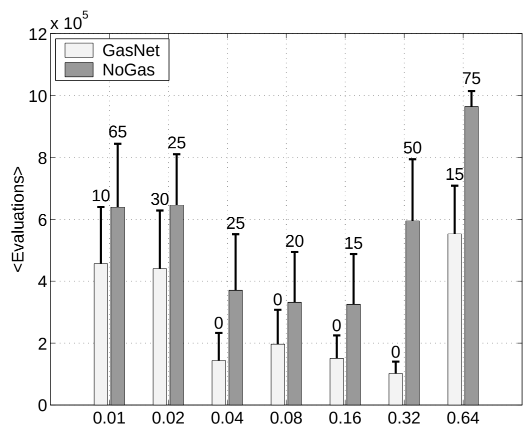
Pattern generation

In a number of evolved robotic controllers, rhythmic pattern generation output was seen to occur, with one or more nodes locked into some limit cycle behaviour (Smith, 2002). Figure 3 shows the behaviour of a two-node subnetwork for a GasNet class solution, with output activity Y, transfer parameters K and gas concentrations C_0, C_1 plotted over 100 time-steps of a sample evaluation. Note the 'spiking' behaviour shown in the node activity Y_2 graph; once every eight time-steps this right-back motor node comes on. This spiking behaviour is crucial to the final fitness of the solution — with both motors on, the robot will move straight-forwards. However, the right-back motor node turning on once in every eight time steps produces a slow clockwise turn in the robot, which in evaluation results in the robot arcing towards the triangle.

Detailed analysis (Smith, 2002) provides an explanation for this (and other) pattern generation, through the interaction between the gas and electrical mechanisms in the network. High electrical output activity of the



(a) Uniform, one-point, and no recombination



(b) No recombination, varying mutation rate μ

Figure 2: The mean number of evaluations required for **(a)** Uniform, one-point and no recombination, and **(b)** No recombination, varying mutation rate $\mu \in \{0.01, 0.02, 0.04, 0.08, 0.16, 0.32, 0.64\}$. Data averaged over twenty runs of the distributed evolutionary algorithm. The error bars represent 95% confidence limits for the mean, the number above the bar gives the percentage of runs failing to finish in 1,000,000 evaluations.

right-back motor node stimulates gas emission from the second node in the network, which in turn inhibits the motor node, in turn stopping the emission of gas which in turn finally allows the motor node to return to high electrical activity, and the pattern repeats. In a number of other successfully evolved GasNet controllers we have observed similar subnetworks; it appears that the properties of the GasNet class lend themselves readily to pattern generation.

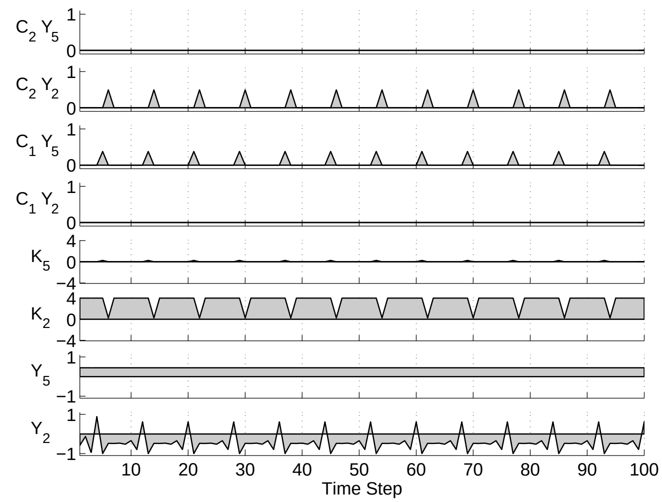


Figure 3: For nodes 2 (the right-back motor node) and 5, involved in the ‘spiking’ subnetwork, the figure shows data over a run of 100 time-steps for node output $Y \in [-1, 1]$, node transfer parameter $K \in [-4, 4]$, positive and negative gas concentrations $C_1, C_2 \in [0, 1]$ at the node site. Area between the output and time axis is shaded for clarity.

Functionally equivalent controllers

Many evolved solutions were found to use functionally equivalent mechanisms for shape discrimination. Figure 4 shows two functionally equivalent subnetworks, one GasNet class controller and one NoGas class controller. Both controllers time the duration over which bright input is received from visual inputs in the upper half of the visual field. A second visual input mechanism (not shown) acts simply as a bright finding detector. This mechanism is ‘later’ in the visual field than the timing mechanism, i.e. the position of visual input to this mechanism is such that it will ‘see’ things after the timing mechanism due to the direction of robot rotation, and is inhibited if the duration of bright input reception is sufficiently long. This inhibition will occur in the situation of scanning across the square but not when scanning across the triangle, so the controllers approach only the triangle. In the next sections we describe the methods by which the GasNet and NoGas classes produce such a timing mechanism.

The GasNet and NoGas “timers” In the GasNet class, a timing mechanism that retains activity for some time after the initial input has been received, is simple to produce. A single node receiving visual input, and with the property that gas emission occurs when the node output activity is high, will start emitting gas when bright input is received. The gas concentration built-up during emission will take some time to decay once bright input is no longer received. Recall from the section on diffusion that gas concentration decays exponentially with distance from the emitting node, but increases linearly

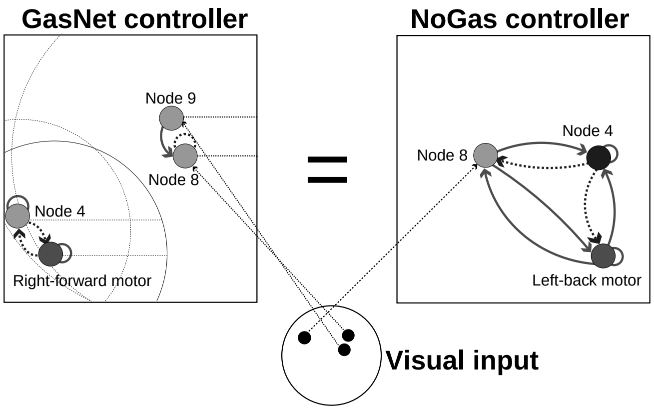
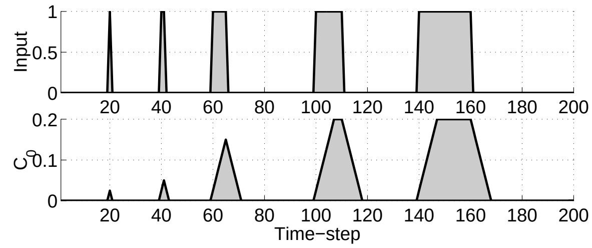
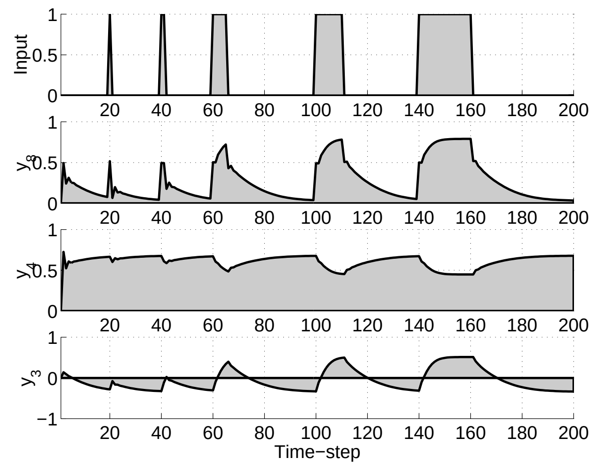


Figure 4: The two functionally equivalent subnetworks. Both employ the same strategy for triangle-square discrimination; timing the duration of receiving bright input in the upper half of the visual field. Due to triangles being narrower than squares at the top, this allows the shapes to be successfully discriminated.



(a) GasNet timing mechanism



(b) NoGas timing mechanism

Figure 5: The GasNet and NoGas timing mechanisms. A square wave external visual input of increasing width is applied as input, to illustrate the differences between the output seen for the triangle and for the square. **(a)** Gas concentration C_0 over 200 time-steps. **(b)** Node output data (ranging from ± 1) for nodes 3, 4 and 8. Area between the output and time axis is shaded for clarity.

over time during emission, and decreases linearly over time once emission stops. Figure 5(a) shows the gas concentration at the motor node over time when square wave visual input is received at nodes 8 and 9.

For this timing mechanism to affect the controller operation, we require the built-up gas concentration to affect the motor node activity. Again, this is relatively simple to effect, as the gas concentration can modulate either the motor node, or as in the controller analysed here, another visual input node which has an output connection to the motor node.

The mechanism underlying the NoGas timer is more complex, and as seen in the re-evolution analysis is harder to tune. Analysis of the fully connected three-node subnetwork (see the right-hand side of figure 4) shows a single stable equilibrium point for the system when visual input is below the input threshold, and a different single stable equilibrium when visual input is above threshold. The key to the timing mechanism is how the system moves between these fixed points when the visual input changes. With no bright input to node 8, the system settles into the first stable fixed point described above, while with bright input the system settles into the second stable point above. Once bright input is no longer received, the system slowly decays back to the first stable fixed point.

The feedback between the nodes ensures the decay between stable states is fairly slow, producing an effect which can build-up and decay over time, in a similar fashion to that of gas concentration. The longer that bright input is received for, the nearer the high visual input stable state the system reaches, and the longer it takes to decay back to the low visual input stable state. Figure 5(b) shows the outputs for the three nodes in the system when a square wave visual input is applied to node 8, and clearly shows the slow change from one state to another when visual input changes from dark to bright, or vice versa. It takes roughly ten time-steps for node 3 to reach the bright input stable state, and roughly thirty steps to decay back to the dark visual input regime. The motor node activity Y_3 is the crucial value; as this goes from negative to positive, the left motor is inhibited, and the robot does not approach the bright object. Only when sufficient bright input has been received will this occur, i.e. when the square has been scanned across.

So why are GasNets evolvable?

From our analysis of evolved controllers, we can frame some preliminary conclusions on the usefulness of the mechanisms utilised in GasNet controllers for the gener-

	Double speed		Quarter speed	
	GasNet	NoGas	GasNet	NoGas
Number of cases	20	20	20	20
Mean evaluated fitness (σ)	0.17 (0.074)	0.15 (0.066)	0.36* (0.10)	0.21 (0.016)
Mean re-evolution generations (σ)	10 (5)**	409 (336)	30 (31)**	591 (346)
Median re-evolution generations	10**	360	19**	608

Table 1: Data for the two functionally equivalent networks shown in figure 4, re-evolved in two modified environments. The robot motors are set to double-speed and quarter-speed respectively, and the two controllers evaluated 100 times for fitness, then seeded back in to the evolutionary algorithm until 100% fitness was reached (20 runs were performed for each controller on each condition). The evaluated fitnesses, and mean, standard deviation σ and median generations of re-evolution required to reach 100% are shown. Significant differences between the GasNet and NoGas controllers are highlighted (both parametric T-tests and non-parametric Mann-Whitney U tests performed $^{*}p < 0.05, ^{**}p < 0.01$).

ation of adaptive behaviour over time.

First, tunable pattern generation is extremely easy to produce using GasNet controllers. In general, pattern generation is based on limit cycle behaviour, with the system cycling through some set of states (Beer et al., 1999; Chiel et al., 1999). The spiking subnetwork (figure 3) operated in exactly such a fashion; the high fitness of the controller was due to this subnetwork slowly turning the robot back towards the triangle. This leads to our first hypothesis for why the GasNet class is more evolvable than the NoGas class; the GasNets are more amenable to being 'tuned' to the specific characteristics of the environment. The pattern produced in which the right-back motor node spiked once in every eight time-steps was perfectly tuned to the speed and size of the robot wheels, the size of the triangle, and the size of the arena in which the robot operated. A different pattern would not have produced such high fitness in this environment, and the same pattern would not have produced such high fitness in a different environment. We hypothesise that the same kind of environmental tuning is more difficult with the NoGas class.

The tuning of generated patterns is closely related to our second hypotheses regarding useful properties in the GasNet class for adaptive behaviour; the ability to switch between stable states, in other words a discontinuous change of behaviour determined by external input, and the ability to mediate such switching. This is clearly possible to achieve without gas modulation, but the features of the gas diffusion mechanism allow such a switch to take place over several time-steps, through the build-up of gas concentration levels. This was seen in the functionally equivalent mechanisms, where the switch from rotation to straight-forwards motion was inhibited by bright input, *only when such bright input was received over several time-steps.* Thus the switching can be based on input patterns received over time, not just at a single time-point.

Finally, the ability to filter out noisy input is straightforward to produce when using the GasNet controller class, also through requiring that input be consistent over several time-steps. This was seen in a number of GasNet controllers, where build-up of gas concentration was used for filtering sensory input. Thus bright flashes and other noisy environment effects were efficiently excluded by the GasNet solutions.

We propose that the crucial property of the GasNet class is *temporal adaptivity*; the ability to be tuned to the specific temporal characteristics of the environment. In the next section we test this through re-evolving solutions in modified environments.

Re-evolution of controllers

Functionally equivalent controllers

The hypothesis that the functionally equivalent GasNet timing mechanism is easier to tune than the functionally equivalent NoGas mechanism can be investigated through the behaviour of the controllers in environments with modified properties. In this section, we analyse the controllers when evaluated in two separate environments, where the robot motor speeds are respectively set to double and quarter the usual motor speeds. This has the effect of making the robot move at a different speed in the arena, in particular spinning past the two shapes at very different rates to the speeds encountered during the original evolutionary phase. Note that the environments could similarly be modified through altering the size and properties of the shapes, and/or the size of the arena. Other modifications could also investigate the effect of re-evolving from lesioned or similarly modified control networks. However, in this work we focus on modification of the robot motor speeds.

We would expect the timing mechanisms to be affected by such speed change, with the time spent spinning past the triangle and square much shorter in the double speed environment, and much longer in the quarter speed environment. However, the hypothesis that the GasNet mechanism is in some sense easier to tune to the particular properties of the environment can be tested through seeding the controllers back into the evolutionary process, with fitness based on evaluation in the modified environments. We can then re-run the evolutionary pro-

	Double speed		Quarter speed	
	GasNet	NoGas	GasNet	NoGas
Number of cases	200	200	200	200
Mean evaluated fitness (σ)	0.27 (0.13)	0.26 (0.18)	0.35 (0.27)	0.29 (0.19)
Mean re-evolution generations (σ)	107 (190)**	240 (363)	108 (229)	116 (252)
Median re-evolution generations	36*	49	13**	21

Table 2: Data for samples of twenty GasNet and twenty NoGas controllers, re-evolved in two modified environments. See table 1 for details.

cess from the controller seeds, assessing how long before controllers of 100% fitness are again achieved. In this re-evolution, we allow only the parts of the genotype involved in the timing mechanism to be affected by the evolutionary process; we are assessing how easy it is to modify the actual mechanism itself, not the rest of the network.

Results for re-evolution studies of the two controllers are given in table 1. In the double speed environment, both controllers drop in fitness to well under 20%, but there is no significant difference between the two controllers in this environment. However, there is massive difference in the number of generations required to re-evolve controllers of 100% fitness; the GasNet controller is much easier to tune to the modified properties of the environment (10 generations on average compared with 409). In the quarter speed environment, the GasNet controller achieves significantly higher fitness than the NoGas controller, but the difference in the number of generations required to reach 100% fitness controllers is much larger than predicted by this fitness difference (the generations of re-evolution required for the NoGas controller to reach the fitness level of the GasNet controller does not affect the results). So in both modified environments, the GasNet controllers are much easier to evolve successful controllers than would be predicted from their fitnesses.

Sample of controllers

It may be argued that the two functionally equivalent controllers investigated are based on a mechanism which is in principle easier to produce using the GasNet control class. Thus re-evolving the timing mechanism in modified environments will unfairly favour the GasNet controller. By contrast other mechanisms may favour NoGas classes; here we counter this argument through extending the re-evolution analysis to a random sample of forty previously evolved GasNet and NoGas controllers of 100% fitness.

The forty controllers were used as initial seeds for the evolutionary algorithm, which was run until controllers once more showed 100% fitness, with fitness evaluated in the same double- and quarter-speed environments described in the previous section. Table 2 shows the results for the two conditions, averaged over ten evolutionary runs of each of the forty controllers. The results are not as striking as those from the functionally equivalent controllers, lending weight to the hypothesis that the previous analysis unfairly favoured the GasNet mechanism. However, the GasNet controllers still showed significantly faster re-evolution than the NoGas controllers. In the double speed environment, both samples of controllers fell to average fitnesses of 0.26, but the GasNet controllers on average re-evolved in 107 generations compared with 240 generations for the NoGas controllers. In the quarter speed environment, the differences are much smaller, with comparable mean numbers of generations for re-evolution, although there is some evidence of faster evolution from the median numbers of generations. Thus from our sample of GasNet controllers, we also see evidence of significantly faster re-evolution to modified environments; the GasNets are more tunable.

Summary

The detailed analysis of a number of GasNet and NoGas controllers allowed us to frame two hypotheses regarding the suitability of the GasNet class to robot control.

First, the ability to both produce and modify central pattern generation output was seen to be central to a number of evolved control solutions. This seems surprising. We are not investigating such behaviours as walking and swimming gaits, or rhythmic feeding, where behaviour is often based on central pattern generation. Our visual shape discrimination task might not at first sight appear to be related to such pattern generation. However, a number of GasNet controllers were seen to use pattern generation subnetworks in the final evolved behaviour.

Second, the ability to switch between dynamical states dependent on external input, and the ability to mediate this switch over a number of time-steps was seen to be extremely useful both in behaviour generation and filtering environmental noise. From analysis of the functionally equivalent GasNet and NoGas controllers, we argued that the kinds of active perception timing strategies able to mediate such behaviour switching and noise filtration were much easier to evolve using the GasNet class. To develop and tune the NoGas timing mechanism required the construction of a complex fully-connected circuit, while the corresponding GasNet timing mechanism was based on the build-up and decay of gas.

The twin hypotheses that GasNet classes were more amenable to both the development and tuning of pattern generation and the development and tuning of switching mechanisms were supported by the re-evolution studies. We saw that the functionally equivalent GasNet controller was much easier to tune to a modified environment than the corresponding NoGas controller. To a lesser extent, although still significant, this same re-evolution tunability was seen over a large sample of previously evolved controllers.

So can we draw any conclusions from this work on what makes an evolvable network class for the visual discrimination problem? The simple answer is yes. The key feature of the GasNets seen to be useful on this task is the ability to smoothly adapt to the temporal characteristics of the environment. This encompasses the initial development and subsequent tuning of the controllers to the detailed properties of the robot and environment in which it finds itself. Included in this ability to smoothly adapt to the temporal characteristics of the environment, is the ability to generate a rich variety of temporal patterns, through the interaction of the gas diffusion mechanism and the electrical synaptic mechanism. The different time courses over which these two mechanisms operate was seen to be crucial to this pattern generation.

Temporally adaptive networks

In the GasNets investigated in this paper, evolved controllers used the modulation of neuron transfer functions as a process operating over a different time course to that of the underlying network synaptic activity. The particular plasticity mechanism used, concentration-dependent neuronal modulation, allowed the GasNet controllers to easily generate and tune temporal patterns, and tune behaviour to the particular temporal characteristics of the environment. And this was seen to have a direct effect on the evolvability of the GasNet class.

By showing that the evolvability of the GasNets is due to this principle of temporal adaptivity, we have provided some support for the intuition of many evolutionary robotics practitioners, namely that robot controllers operating in the real world must incorporate temporal structure, and that the evolutionary process must be able to easily adapt that structure. For example, Harvey (1993) makes the point that "... in environments where physical events have natural time-scales, the dimension of time is not an optional extra, but fundamental." Similarly, Gallagher and Beer (1999) state that "... nontrivial behavior requires the integration of experiences across time and the ability to initiate actions independent of an agent's immediate circumstances."

On this fundamental principle of temporal adaptivity, the GasNet neural network class falls squarely into a larger class that is likely to contain other networks with temporally modifiable properties, such as continuous time recurrent networks (Beer and Gallagher, 1992), pulsed neural networks (Maass and Bishop, 1999), and networks with time-lagged synaptic activity (Harvey, 1993). However, we argue that simple recurrent networks such as the NoGas are not members of this class; although activity is retained over time and the connection architecture may be arbitrarily modified, it is not straightforward for the evolutionary process to temporally modify the activity of the network. For example, as seen in the NoGas timer, it is not easy to setup an effect that lasts for some given period of time. Similarly, generating and tuning patterns is more difficult with simple recurrent networks (Smith, 2002). It seems plausible that if we are to further develop evolvable artificial neural network classes for generating adaptive behaviour over time, the starting point must be from within the class of temporally adaptive networks.

Finally, we come back to the general direction in which this paper has aimed; identification of the fundamental properties necessary for adaptive behaviour. We have shown how analysis of successful solutions on one task can identify more general properties of the solution class that are potentially useful over a wider range of problems. It is of course necessary to apply the methodology developed here to both a number of robot control tasks and a number of solution classes, in order to fully support the temporally adaptive hypothesis presented in this paper. However, further analysis may also shed light on other fundamental properties necessary for the generation of adaptive behaviour over time. Such analysis of more complex adaptive systems can, in principle, similarly illuminate the general conditions necessary to make the transition from specific to generic response systems.

Acknowledgements

The authors would like to thank the members of the CCNR (http://www.cogs.susx.ac.uk/ccnr/) for constructive discussion. We would also like to thank the Sussex High Performance Computing Initiative (http://www.hpc.sussex.ac.uk/) for computing support. TS is funded by a BTexaCT Future Technologies Group sponsored CASE studentship (http://www.btexact.com/projects/ftg/).

References

Bedau, M. A., McCaskill, J. S., Packard, N. H., Rasmussen, S., Adami, C., Green, D. G., Ikegami, T., Kaneko, K., and Ray, T. S. (2000). Open problems in Artificial Life. *Artificial Life*, 6:363–376.

Beer, R., Chiel, H., and Gallagher, J. (1999). Evolution and analysis of model CPGs for walking II. general principles and individual variability. *Journal of Computational Neuroscience*, 7(2):119–147.

Beer, R. D. and Gallagher, J. C. (1992). Evolving dynamical neural networks for adaptive behaviour. *Adaptive Behaviour*, 1:94–110.

Chiel, H., Beer, R., and Gallagher, J. (1999). Evolution and analysis of model CPGs for walking I. Dynamical Modules. *Journal of Computational Neuroscience*, 7(2):99–118.

Gallagher, J. C. and Beer, R. D. (1999). Evolution and analysis of dynamical neural networks for agents integrating vision, locomotion, and short-term memory. In Banzhaf, W. and Smith, R. E., editors, *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO'99)*. Morgan Kaufmann, San Mateo, California.

Harvey, I. (1993). *The Artificial Evolution of Adaptive Behaviour*. PhD thesis, School of Cognitive and Computing Sciences, University of Sussex, UK.

Husbands, P. (1998). Evolving robot behaviours with diffusing gas networks. In Husbands, P. and Meyer, J.-A., editors, *Evolutionary Robotics: First European Workshop, EvoRobot98*, pages 71–86. Springer, Berlin.

Husbands, P., Smith, T., Jakobi, N., and O'Shea, M. (1998). Better living through chemistry: Evolving GasNets for robot control. *Connection Science*, 10(3-4):185–210.

Jakobi, N. (1998). Evolutionary robotics and the radical envelope of noise hypothesis. *Adaptive Behaviour*, 6:325–368.

Maass, W. and Bishop, C., editors (1999). *Pulsed Neural Networks*. MIT Press, Cambridge, Massachusetts.

Smith, T. M. C. (2002). *The Evolvability of Artificial Neural Networks for Robot Control*. PhD thesis, School of Biological Sciences, University of Sussex, UK.