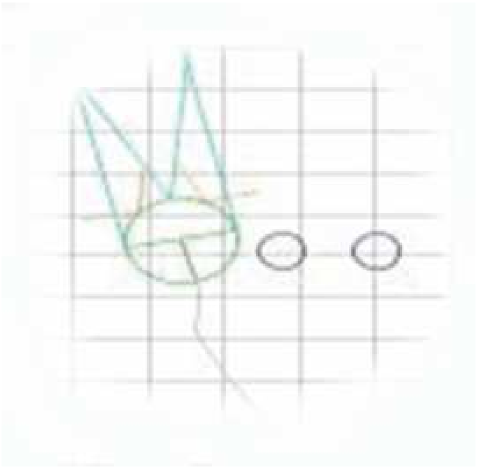
Towards Avoidance and Detour Behaviour.

An Exploration of Domains of Plasticity in Evolutionary Robotics.



D. J. Cowan. September 2002.
School of Cognitive and Computing Sciences,
University of Sussex.

Special thanks to:

Caroline Davies for taking care of Oli through a long summer holiday while I worked.

Phil Husbands for weekly supervision and world class support.

Tony Poppleton for algorithms and sandwiches.

E. P. S. R. C. for footing the bill.

Thank you.

Abstract:

Cognition has evolved such that it can quietly reflect upon the deep mystery of its origin. However, such highs were not achieved by evolving synaptic profiles alone. Natural life-forms are the result of an evolutionary lineage in which every aspect of a species was at some time subject to change.

Traditional experiments in evolutionary robotics have focussed on evolving neural network weights. Here we extend the techniques of evolutionary robotics to evolve further domains of plasticity: sensor and network morphologies. Our aim is to explore the effect of these additional domains on the evolutionary process.

Avoidance and detour behaviours are evolved in populations of simulated robots controlled by perceptron, dynamic-recurrent and sparsely connected neural networks. Each network class is trialled with fixed, seeded and fully evolved sensor morphologies. Experimental results suggest that with full evolution many more behavioural traits can be explored but that this opening up of trait space slows down and sometimes confounds evolution. Despite start-up difficulties, fitter agents are possible when more of the design process is left to evolution.

Contents:

1 Introduction:

The purpose of this project is to begin an exploration into the effects of additional domains of plasticity on evolvability. In this introduction we begin with some general theoretical groundwork to define what is meant by a domain of plasticity and its connection to embodiment and behaviour. After introducing recent work in this area we focus on the task in hand and the hypothesis under examination.

1.1 Embodiment.

One possible characterisation of artificial life research is that of the study of embodied and embedded cognition. The language is found in diverse works from the theoretical biology of Maturana and Varela to the practical robot building of Rodney Brooks (Maturana and Varela 1987, Brooks 1990). Yet this widespread usage conceals a veritable plethora of meanings; considering that one may 'embody an ideal' the situation is certainly complex.

Attempts to clarify our understanding (Quick, Dautenhahn and Nehaniv 1999, Sharkey and Ziemke 1999, Ziemke 2001) point out this wide usage and derive a number of connected concepts. Quick *et al.* suggest an embodied system be structurally coupled to its environment through mutual perturbation, either in this world or in virtual environments (*ibid.*) Although this offers little in discrimination it defines a baseline minimum condition for embodied cognition. This leads on to physical and morphological embodiment, the significance of which is highlighted by Rolf Pfeifer amongst others (Pfeifer 1999). The shapes and material affordances of our physical embodiment both enable and constrain cognitive function. For example, Barbara Webb's work exploring con-specific mate location in crickets demonstrates how the shape and material of the cricket tracheal tubes enables phonotaxis (Webb 1995).

Ziemke observes that cognitive systems are also historically embodied; their behaviour is determined by the history of their interactions with their world (Ziemke 2001). Maturana and Varela consider two distinct forms of historical embodiment, those of the phylogenetic and the ontogenetic histories of an entity (Maturana and Varela 1987). They embrace both the slow adaptation of species to their changing environment and the developmental or growth cycle, including lifetime learning, of individuals within that species. A final sense of embodiment is derived from the phenomenological experience of our worlds (Sharkey and Ziemke 1999). VonUexkull discussed the importance of '*umwelt*' as early as 1928. This is the world of attention, a perceptual subset peculiar to a species. Maturana and Varela argue for their being a phenomenological experience of an *umwelt* to cohere and unify a meta-cellular autopoietic entity's being (Maturana and Varela 1987). This phenomenological world can also be understood through the value projections an entity makes in attending to its world's affordances.

So, one can be embodied across (at least) four related modes: the structural, physical, historical and the phenomenological, each of which enables cognition in some way. We concur with Maturana and Varela in characterising cognition as that which living things do to maintain their autopoietic identity (*ibid.*). Cognitive entities adapt to environmental perturbation so as to maintain their being in the world. Now, adaptation requires plasticity. We suggest that the plasticity required to support cognition is found across each of the aforementioned modes of embodiment. Further, the remarkable complexity and ingenuity found in high order cognition is enabled by these modes in concert. A fully embodied agent supports plasticity across all the interrelated modes of its being. It is this total plasticity that enables cognition to reach such highs as self investigation.

Just as embodiment might be understood through connected modes that enable total plasticity, total plasticity can be bundled into connected domains. A domain contains a set of closely related parameters that determine an entity in a way which might have been otherwise. For example one might consider a cross-species domain such as leg number, natural creatures can have 2, 4, 6, 8 or many legs; a domain of phylogenetic plasticity. At a closer level there are material and mechanical possibilities for each leg type and in individual legs there are ontogenetic morphological variations. In some sense each of these domains of plasticity supports the complexity of cognition found in creatures embodying them.

This substantial philosophical thesis is left largely unsupported here. We wish only to introduce the notion that there are many modes of embodiment, each of which is significant to our understanding of cognition in natural systems. Each supports plasticity, and plasticity is what enables the richness of adaptive behaviour in living creatures. No strong identification or denomination is intended. Domains of plasticity are linguistic constructs intended as tools in our exploration; we are not to be bewitched (c.f. Wittgenstein 1949).

1.2 Evolving Embodiment.

In the 10 years since its conception evolutionary robotics has become a flourishing research discipline. Notable successes include navigation (Floreano and Mondada 1996a), garbage collection (Nolfi 1997) and even legged locomotion (Jakobi 1998). These are classic examples in a field dominated by the evolution of controllers for fixed morphology agents. Evolution of this single domain of plasticity reflects the easy availability of 'off the shelf' robots such as the ubiquitous Khepera and the considerable difficulty of real world physics simulation. Nevertheless, as early as 1994 Karl Simms began the co-evolution of body morphology and controller, and Harvey, Husbands and Cliff evolved sensor morphologies for a vision system in a real world robot (Simms 1994, Harvey *et al.* 1994).

In recent years there has been a growing appreciation for the role of physical and morphological embodiment. For example Pfeifer (1999) has shown how clustering behaviour in a population of real world agents depends critically on their morphology. Developmental cycles exploring ontogenetic embodiment feature in the work of Eggenberger (1996) and Dalliert and Beer (1996). Although these often focus on controller genesis, Josh Bongard has evolved box-pushing agents whose simulated physical structure was developed through a model genetic regulatory network (Bongard and Pfeifer 2001). In similar work, Lipson and Pollock (2000) have evolved real world agents based on L-system fractals where the evolved locomotion behaviours depend on their physical morphology as concatenations of limbs.

To employ the parlance each of these projects extends evolutionary robotics into a new domain of plasticity. Some of this work involves ceding the parameterisation of additional morphological features to an evolutionary algorithm; seen in the evolution of biped morphology by Bongard and Paul (2001) or the development of neural network connectivity in Gruau (1994). Other work evolves a process to support lifetime plasticity such as the plastic neurocontrollers of Di Paolo (2000) or Floreano and Mondada (1996b). Of particular interest to this work are those domains of plasticity which directly impinge on the evolvability of an agent.

Evolvability is to be understood in two senses. In simple terms, that system which produces fitter agents more quickly might be considered the more evolvable system. On a deeper level we look for behavioural strategies which have been enabled by additional plasticity. A strategy or behaviour can be understood as a trajectory through behaviour space; a meta-dynamical-system determined by the agent-environment coupling and their individual dynamical sub-systems. For example looping and direct strategies are seen in simple phototropism; both are attractors within the behaviour space of the given agent-world dynamics.

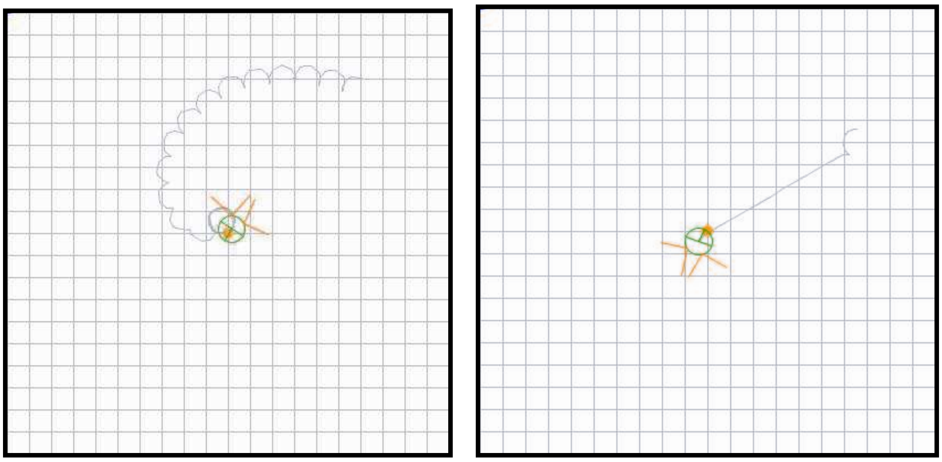


Figure 1.2.1: Common trajectories in single light phototropism space.

A domain of plasticity may increase evolvability through enabling new attractors in behaviour space which could not have been found otherwise. This dynamical systems description opens the door to another characterisation of a domain of plasticity. Such a domain would contain a closely related set of state space variables, such as sensor parameters or network weights or structure parameters.

1.3 The Hypothesis.

Having operationalised a domain of plasticity as a connected set of variables embodied by an agent to support adaptive behaviour, we can state the driving hypothesis of this work:

that additional domains of plasticity increase evolvability.

Such a hypothesis is explored through the evolution of avoidance and detour behaviours in populations of simulated agents. Exploration is facilitated through comparable evolution of sensor morphologies and control architectures - two domains of plasticity.

1.4 Related Work.

Systematic theoretical analysis of artificial evolution began with Holland's work on schema theory (Holland 1975) and extends to modern work understanding neutrality (Barnett 2001), the Baldwin Effect (Mayley 1996) or co-evolution (Noble and Watson 2001). Similarly, the philosophical and practical implications of embodiment are widely explored within cognitive science (Merleau-Ponty 1962, Clarke 1997). This work brings both of these strands together as a practical project in evolutionary robotics.

Sensor evolution also features from the very beginning. We have already mentioned the work of Husbands, Harvey and team at Sussex University in evolving visual morphologies for real world robots (see Husbands *et al.* 1996). Cliff and Miller applied an L-system grammar to developing both the sensors and control architecture in a simulated predator-prey system (Cliff and Miller 1995). More recently Balakrishnan and Honavar evolved the relative placement and range of sensors on a simulated agent and demonstrate increased task performance (Balakrishnan and Honavar 1996). Mark, Polani and Uthmann have explored eye width and number in evolved Braitenberg vehicles trialled on an incrementally more complex photo-tropism task (Mark *et al.* 1998). Agents are shown to evolve higher resolution vision as task complexity increases.

Object avoidance has been studied extensively in Floreano and Mondada (1994), Salomon (1996) or Huber, Mallot and Bulthoff (1996), amongst many others. The detour task was inspired by studies of detour behaviour in frogs by Michael Arbib (1982, 1987) although the work reported here is not meant to be biologically defensible or to implement schema theory. Use of the detour task *in abstraction* is found in Migliano, Denaro and Tascini (1998). Theirs is a very much simplified model; the agent can see both the light and continuous walls at all times.

This work is intended to contribute to our theoretical understanding of artificial evolution through the extension of evolved parameters into the realm of embodiment. Following inspiration from the literature, evolved sensor and controller morphologies are applied to increasingly complex tasks and their contribution to evolvability assessed.

2 Methods:

In this section we introduce the methods used to complete the project. Beginning with the controllers, we embody and situate the agents in their world before discussing the algorithms responsible for their evolution. We end by drawing these elements together into the experiment.

2.1 Artificial Neural Network Controllers.

The agents are controlled by artificial neural networks. Three architectures of increasing complexity were trialled based on perceptrons (Rosenblatt 1962) and dynamic-recurrent neural networks (Beer and Gallagher 1992).

In the perceptron network each node is governed by the equations:

$$a = \sum_{\forall j} w_j x_j \qquad \text{and} \qquad y = \sigma(a) \qquad (1,\ 2)$$

where y is the output of the node, a the activation, x_j the input from the j^{th} node of the preceding layer and w_j the synaptic weight. The network morphology includes two layers, a non-summing input layer with one node per sensor and a two node output layer. Evolved network parameters include the weights and bias terms for each node; evolved within the range [-5, 5] to best utilise the sigmoidal transfer function.

$$\sigma(x) = \frac{1}{1+e^{-x}} \qquad (3)$$

The dynamic-recurrent neural networks are more complex; each node integrates its mean membrane potential with respect to its previous potential so introducing internal state - a kind of 'memory'. Nodes are governed by the state equation:

$$\tau_i \frac{dy}{dt} = -y_i + I(t) + \sum_{\forall j} w_{ij} \sigma(g_j.(y_j - \theta_j)) \qquad (4)$$

where τ_i is the time constant of the i^{th} node, y the output or mean membrane potential and $I(t)$ the input at time t. w_{ij} is the weight between node i and node j, σ represents the mean short-term firing frequency of the node j given in (3), g is that node's gain and θ its threshold value.

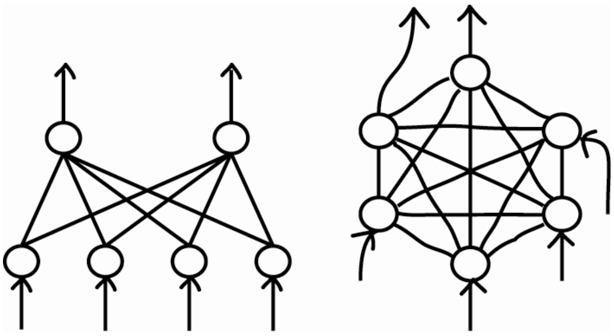


Figure 2.1.1: Perceptron and dynamic-recurrent neural network architectures

The fully interconnected architecture employs a simple genotypic encoding for fixed size networks. For each node there is a time constant, bias term, gain and also one weight per node in the net. In all there must be $n^2 + 3n$ loci upon the genotype. The ranges for the time constants, biases, gains and weights are not evolved, being established by experimentation and literature review (Beer 1996, Husbands 1998). Bias and weight terms lay in the range [-5, 5] to maximise the utility of the sigmoidal transfer function. Time constants range between [0.2 and 5] for 'fast thinking' while, in these experiments, gain was always 1.0.

Full connectivity yields two potential difficulties. First there is an explosive increase in genotype length for additional nodes. Furthermore, mean membrane potential increases with connectivity eventually swamping the relative perturbation by sensory input; full connectivity tends toward rhythmic oscillation. These difficulties are addressed through the use of spatially distributed sparse network encodings.

Spatial encodings (e.g. Husbands *ibid.*) have been employed to great effect and are generally much simpler than the alternative biologically inspired encodings (Gruau 1995; Eggenberger 1996). Our encoding uses just nine loci per node: three to specify the time constant, bias and gain and six to generate connectivity. Each neuron is located on a 2D-grid by evolved coordinates. Neurons have a sector of influence specified by a radius, direction and arc. Any node falling within that compass is counted as an input to that neuron.

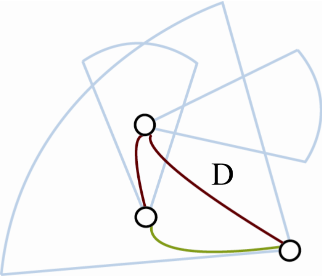


Figure 2.1.2: sparse connectivity determined by spatial arrangement and sectors of influence

$$w = \cos(D + \theta)$$

where θ is the offset or phase

Weights are calculated as the cosine of the absolute distance between nodes plus an evolved offset. The cosine of the distance yields a weight in the range [-1, 1] for both inhibitory and excitatory synapses. The size of the spatial array is scaled to some order of π determining the number of sine wave propagations across the array. With a single propagation there is a strong bias towards weights within the centre of the range, some weight sets not being reachable.

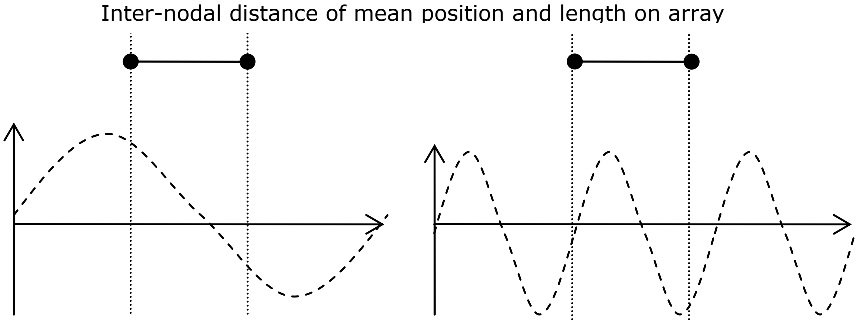


Figure 2.1.3: Restricted weight set in single wave case

Moreover, weights become bound to each other as the movement of a node on the array necessarily affects its entire weight space. Multiple wave propagation alleviates the bias and allows weight independence because there is always more than one position in a given weight dimension for any value. A node might move a small amount in respect of one neighbour, so changing that weight, but by an order of π in respect of a second neighbour, leaving that weight unchanged.

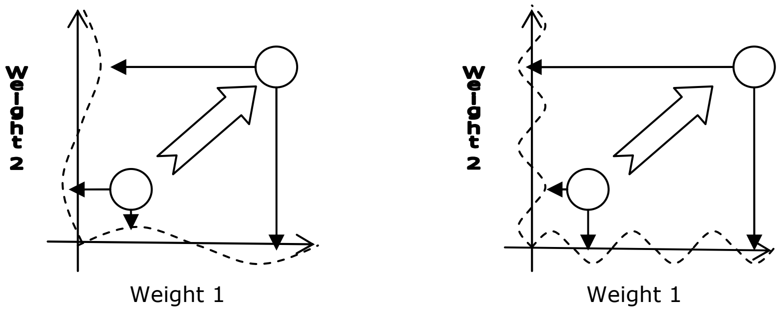


Figure 2.1.4: In case one the move affects both weights,
in case two, only weight 2 changes

The encoding is parameterised to interpret weight and bias terms in the range [-5, 5], time [0.2, 5] and gain [1, 1]. Maximum radius is set at some factor of the grid size (usually 0.5) and the grid scaled by some factor of π (usually 5).

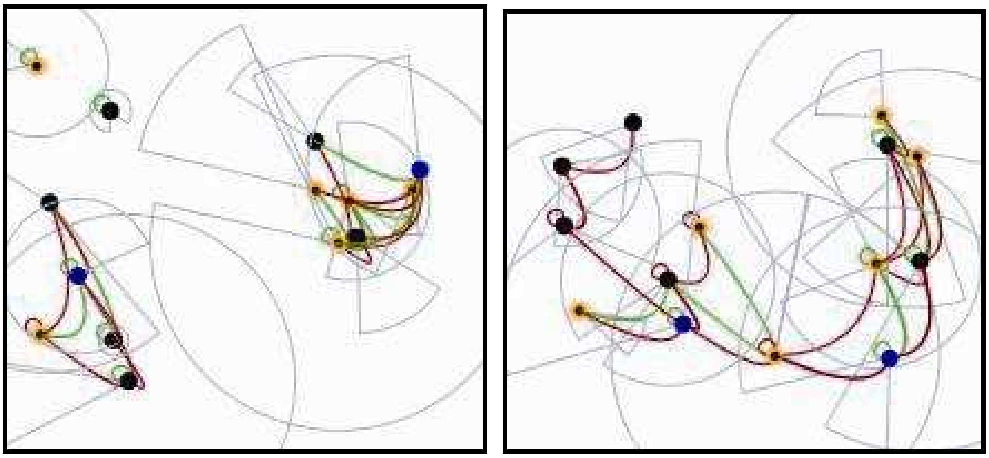


Figure 2.1.5: Fit spatially encoded neural networks.

2.2 The Agents and their World.

The agent was originally derived from a model of a real-world robot built by the author. The real agent's large wheels, knobbly tyres and low ratio motors justify many of the simplifications to the world physics necessary for a fast simulation: it does not bounce or skid and has imperceptible acceleration and deceleration.

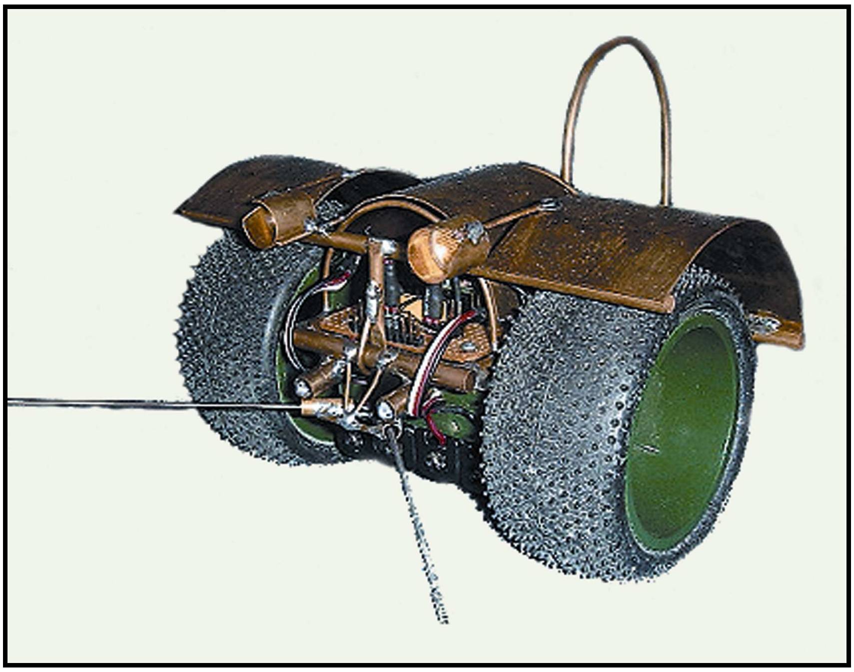


Figure 2.2.1: Wheelie, a robust research platform.

However, since the original simulation has been extended to include ray traced light and laser ranging sensors additional to the ambient sensors, robot and model have diverged. The agent has become a circular body equipped with two motors and a variable sensor array, now resembling a certain traditional model in evolutionary robotics.

Motors are driven by motor neurons with variable and unexpected output ranges so the motor neuron activations are passed through a sigmoid. Neuronal output was banded such that:

$$m = \begin{array}{l} 4 \cdots \text{if } y > 0.6 \\ 0 \cdots \text{if } y < 0.6 \wedge y > 0.4 \\ -4 \cdots \text{if } y < 0.4 \end{array} \qquad (5)$$

The motor values are taken as the speed of each motor and the agent's progress is derived via a simplified physics from these values.

Of the available implemented sensors only directional light and laser ranging sensors were used. For laser ranging sensors a single ray is traced over their range and a normalised value returned: 1.0 for objects close to and 0.0 for objects at the limit of the range. Light sensors return this normalised range multiplied by the intensity of the light sighted. The directional light sensors trace rays between the sensor and each light in the environment. Those rays which are both uninterrupted and within the sensor's angle of acceptance are summed to return the reading.

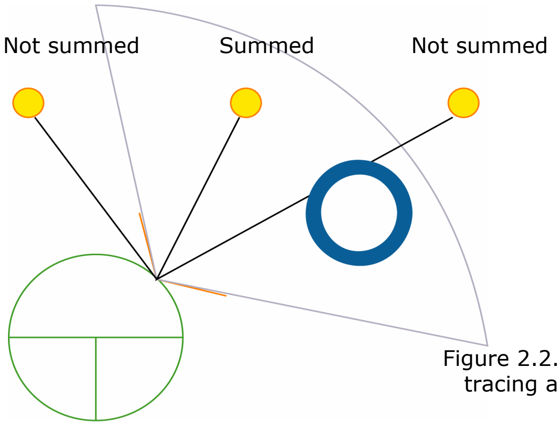


Figure 2.2.2: Fast ray tracing algorithm.

Two methods of generating sensors were employed. The simple method reads pre-defined sensor arrays from a file using these to populate a sensor chromosome. The second generates wholly random chromosomes. In both methods the number and type of sensors is not evolved. If the sensor chromosome's mutation rate is set to 0.0 then no sensor evolution will occur, comparable to the traditional evolution of the neural network weights on a fixed architecture agent. Each sensor is encoded by 3 parameters specifying its location on the agent's body, its range and direction. Range is interpreted as the angle of acceptance for directional light sensors.

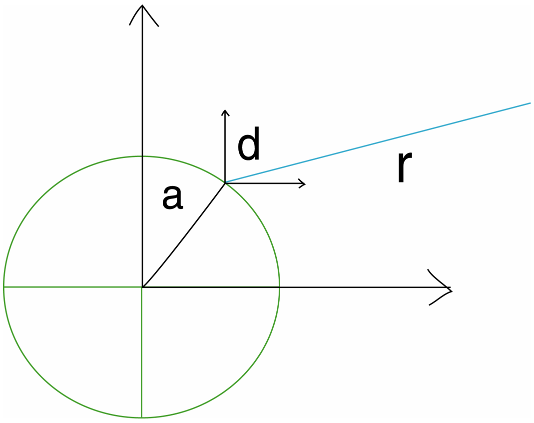


Figure 2.2.3: Sensor chromosome: 'adradradradr'

The world comprises an infinite two dimensional plane featuring a greatly simplified physics. Forces are not calculated beyond motor speeds, collisions are inelastic and friction is crude, allowing movement but no skidding. Noise, at variable levels, is introduced at the sensory-motor interface. Entities include walls, cylinders and lights with fixed positions. A world is populated with entities from an ASCII description with a particular fitness test in mind. In order to better explore the strategies employed by each successful agent a number of worlds are designed for each fitness test. Niche worlds, in which the evolution takes place, are supported by further worlds which facilitate analysis of the evolved behaviours.

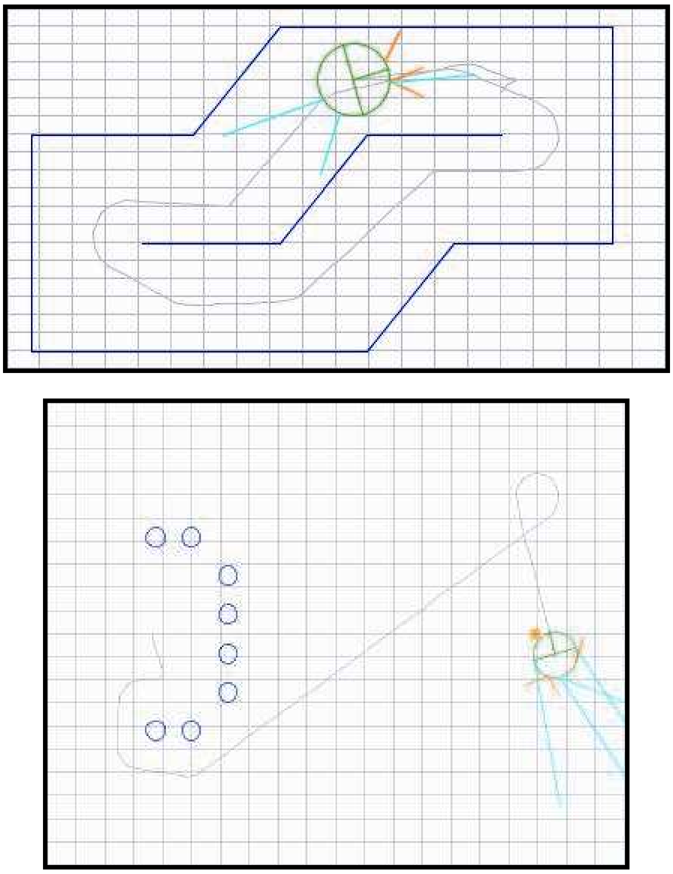


Figure 2.2.4: Main avoidance and detour worlds.

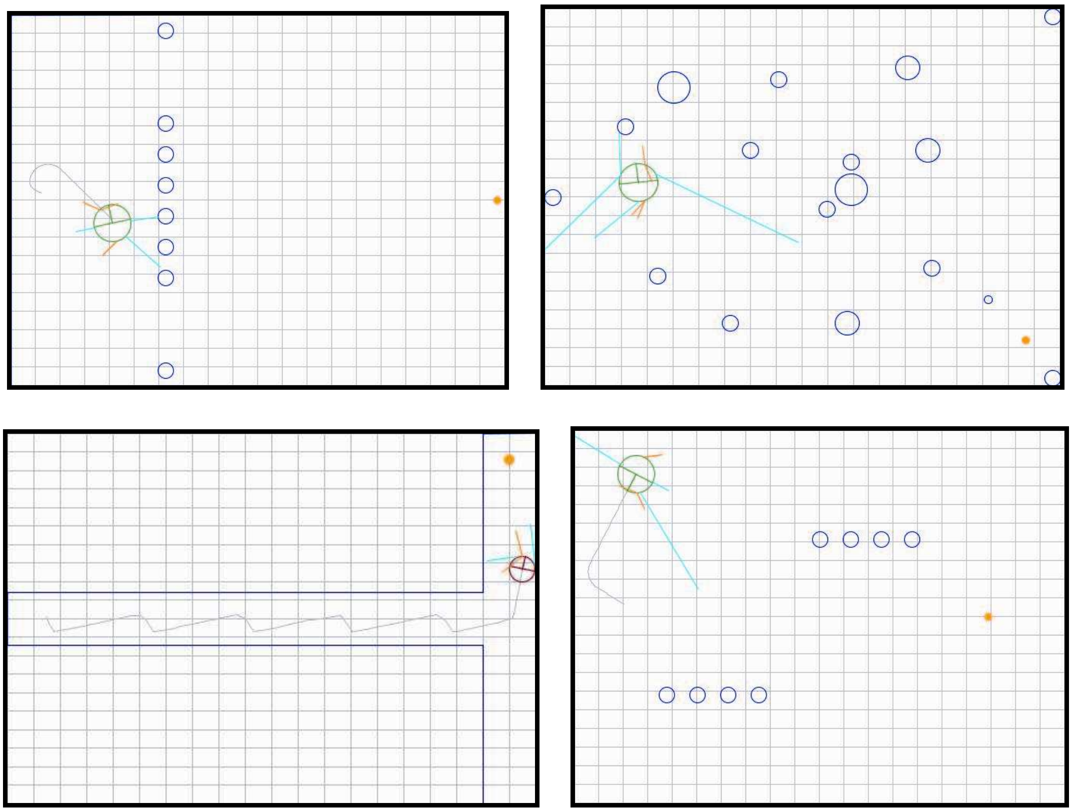


Figure 2.2.5: Various support worlds.

2.3 The Genetic Algorithm.

Evolution is driven by a spatially distributed genetic algorithm (Collins and Jefferson, 1991). Genomes are distributed upon a square toroidal grid. A genome is selected at random and a breeding pool is generated from the eight nearest neighbours. The pool is ranked according to ascending fitness and a partner selected by inverse rank proportionate roulette wheel selection governed by the equation:

$$P(a) = \frac{a}{28}, \text{ where } a[0,\ 7] \qquad (6)$$

where P is the probability of selecting the agent a in the list. Offspring are produced using crossover and mutation operators and placed back in the population by overwriting one among the less fit of the neighbours, again selected proportionate to rank.

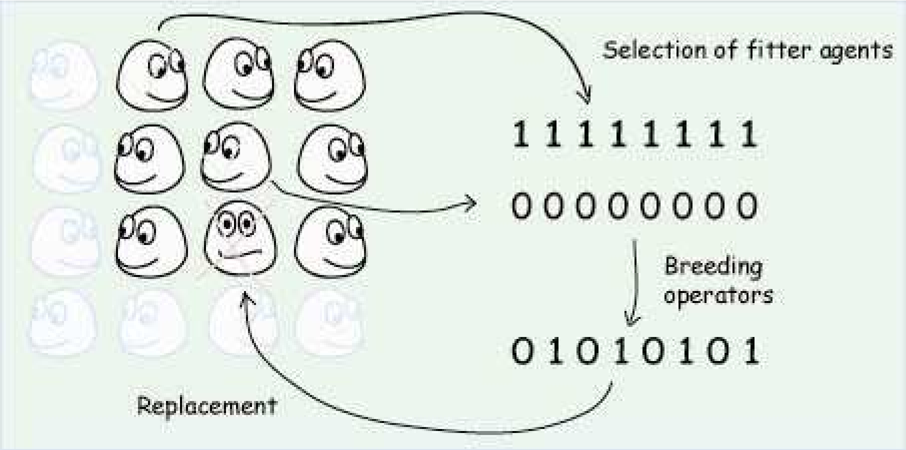


Figure 2.3.1: One iteration of a distributed G.A.

The selection pressure hides a form of elitism in as much as it is not possible for the fittest agent to be selected for overwriting nor for the least fit to be selected for breeding. Further elitism is built in to the algorithm and utilised through flags in the program. The algorithm always prepares both possible offspring (one being the 'negative' of the other). If the 'elite-child' flag is set then the fitter of the two children is returned to the population, otherwise one is returned at random. If the offspring is less fit than the target selected for overwriting a further flag, 'elite-replace', determines whether the offspring will in-fact overwrite the neighbour or be discarded.

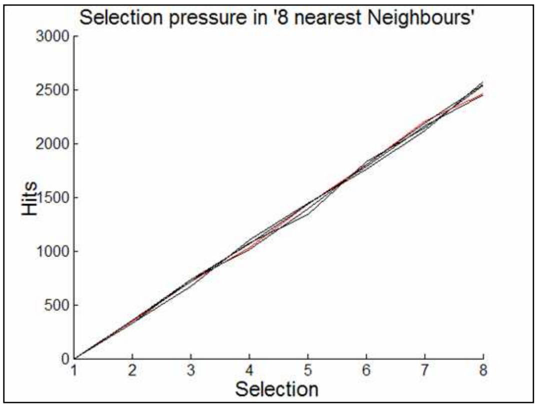


Figure 2.3.2: Selection pressure.

A genome comprises of three chromosomes - one for the control architecture and two for sensor profiles. Chromosomes are strings of real numbers in the range [0, 1] to double precision (64bit). The chromosomal model supports multiple operator sets allowing different rates and styles of genetic operation on different parts of the agent.

For example the controller might be evolved with a fixed sensor profile and then the sensors evolved at a slower rate once capable sub species have emerged in the population. Agents are derived from their genomes through the encodings discussed above and specified on initialisation. The encodings scale the genetic range [0, 1] to the appropriate phenotypic ranges and establish the fixed elements of the experiment.

Uniform crossover was used in all trials. Two parental genomes are crossed into two children with an equal chance that each loci on parent A will end up in either child.

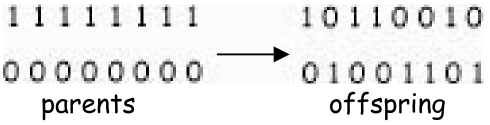


Figure 2.3.3: Uniform crossover.

In this way both variant offspring are produced with no bias as to which parent was selected first. As stated above, the choice of return offspring is determined through flags and ranges from a purely random return to the return of fitter agents only.

Three mutation operators were available including Gaussian point mutation, stepped Gaussian mutation and vector mutation. The first operator merely returns a random number from a Gaussian distribution centred on the loci value with a standard deviation of 0.2. Stepped Gaussian mutation is a composite of two operators found in Husbands (1998); a small Gaussian variation is returned with a probability of *P(0.8)* and a non Gaussian random number in the range [0, 1] otherwise. Point mutations were employed at a rate of approximately 1 mutation per genome. Vector mutation is a variation of the technique reported in Beer (1996). The vector length of a chromosome is varied by some small amount from a Gaussian distribution about the length. This change is then distributed among the component reals. Network chromosomes were mutated with a mean distance between 1 and 3 in various trials. All three mutation operators were used but the vast majority of trials used vector mutation for the network parameters and stepped Gaussian point mutation for the sensors. Legal bounds were maintained by capping and collaring.

One generation of a distributed grid algorithm is approximately n iterations, n being the size of the population. Each genome in the population is evaluated on initialisation. The problem of dominant 'lucky' agents is addressed in the testing regime. Each agent is evaluated repeatedly, between 6 and 10 trials, from random starting positions and orientations within their world. Each evaluation score is ranked inversely proportional to fitness and overall fitness calculated as the weighted mean such that:

$$\Phi = \sum_{n=0}^{n=evaluations} n.f_n \tag{7}$$

where Φ is the recorded fitness and f_n is the fitness score of the n^{th} evaluation. An agent must score consistently well across all evaluations for a good overall fitness score under such a weighting regime.

2.4 The Experiment.

The investigation involved exploring the effect of domains of plasticity on the evolvability of detour and avoidance behaviours. The project allowed for variation across 2 domains of plasticity:

1. Controller architectures:
    - Perceptron network.
    - Fully interconnected dynamic-recurrent network.
    - Sparsely connected dynamic-recurrent network.
2. Sensor morphology:
    - Seeded sensor profiles.
    - Non-seeded sensor profiles.

Plasticity was varied from trial to trial across one or more of the available domains generating statistical data for analysis. Data produced was examined for signs of increased evolvability.

Preliminary experiments into avoidance behaviour were trialled in a world based on Floreano's experiments into object avoidance using real Khepera robots (Floreano and Mondada 1994). The world encourages both left and right turns and should be narrow enough to cause some sensor stimulation at all times. Floreano introduced a fitness function based on rewarding desirable component parts of the behaviour, which was initially adopted:

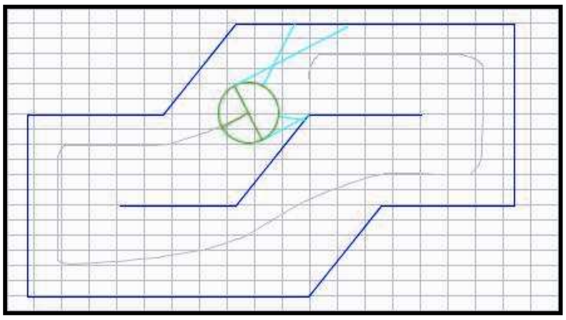


Figure 2.4.1: Floreano-world.

$$\Phi = \sum_{n=0}^{n=steps} v_n.(1-\sqrt{dv_n}).(1-I_n) \qquad (8)$$

where Φ is fitness, v is the absolute distance moved, dv the normalised absolute rate of turn and I the highest distance-sensor reading.

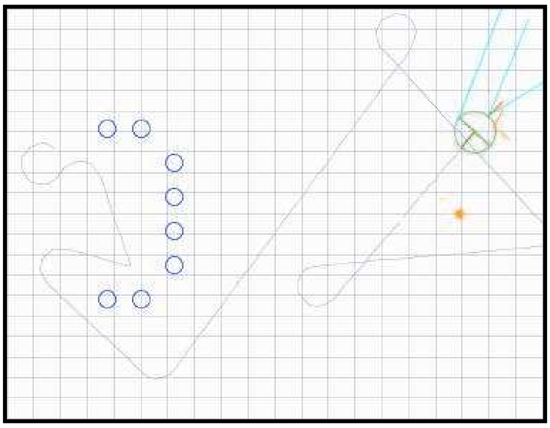


Figure 2.4.2: Barrier-world

Detour behaviour was initially inspired by toad and frog detour experiments (Arbib 1987).The detour worlds are designed such that the agent begins facing a fence of cylinders and a bright lamp. The agent is expected to move towards the lamp, while detouring the fence as necessary. A component based fitness function was once again employed, this time rewarding lamp approach behaviour and straight line motion. Hard lessons were learned during avoidance trails; the detour fitness test resembles the avoidance test but differs in key ways:

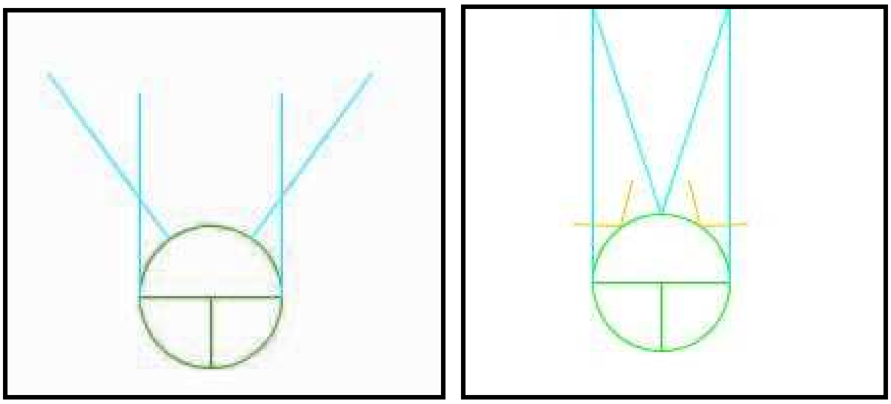
$$\Phi = \sum_{n=0}^{n=steps} v_n . l_n \qquad (9)$$

where Φ is fitness and *v* and *l* are calculated:

$$l = \frac{600 - |d|}{150} \qquad v = \begin{cases} 4 & \text{if both motors full forwards} \\ 1 & \text{if one motor forwards, one stationary} \\ 0 & \text{otherwise, (full turn and reverse cases)} \end{cases}$$

d being the agent's distance from the source. The *v* term causes both straight line motion and implicit object avoidance while the *l* term causes phototropism. With its discontinuous sensor readings and dual nature, detour behaviour is a considerably more complex task.

In each of these problem domains, trials were run using different network architectures. For each architecture three sensor regimes were tested: fixed, seeded and fully evolved. The fixed array was designed having observed this class of agent performing object avoidance. It was decided to permit two forward pointing sensors at the agent's girth to allow the detection of sizeable gaps and two angled forward pointing sensors emanating from the agent's front to allow collision detection. This array was trialled against comparable arrays and adopted as standard for offering a balance between too little and too much information. Other possible sensor arrays would be searched in those parts of the experiment in which sensor arrays were to be evolved; more will be said about this later.



2.4.3: Adopted patterns of fixed and seeded sensor arrays.

The same array was used to seed sensor evolution, optimisation being carried out through mutation whilst fully evolved sensor arrays began with random chromosomes. The detour behaviour utilised the modification to the right incorporating 2 forward pointing directional light sensors with an angle of acceptance of π/3.

3 Results:

In all this made for 15 experiments involving repeated evolutionary runs in each class for generality. For each experiment we will describe the evolutionary statistics and results of the run before discussing the evolved behaviours. In the following section, analysis, we will bring all the data and comments together for reflection.

Appropriate mutation for each trial was established through a number of short dry runs. In each run the genetic variance of the population was estimated by examining the mean and standard deviation of each locus upon each genome. For a random population the loci average should approximate 0.5 and the standard deviation 0.27. As a particular locus in the population tends to converge the locus' mean will change and the standard deviation decrease. Variation across the whole population can be measured with the mean standard deviation per locus. Although the mean mean per locus should still approximate 0.5, the mean standard deviation will be minimised in a converged population. Mutation rates were selected to slow down the convergence driven by the selection pressure.

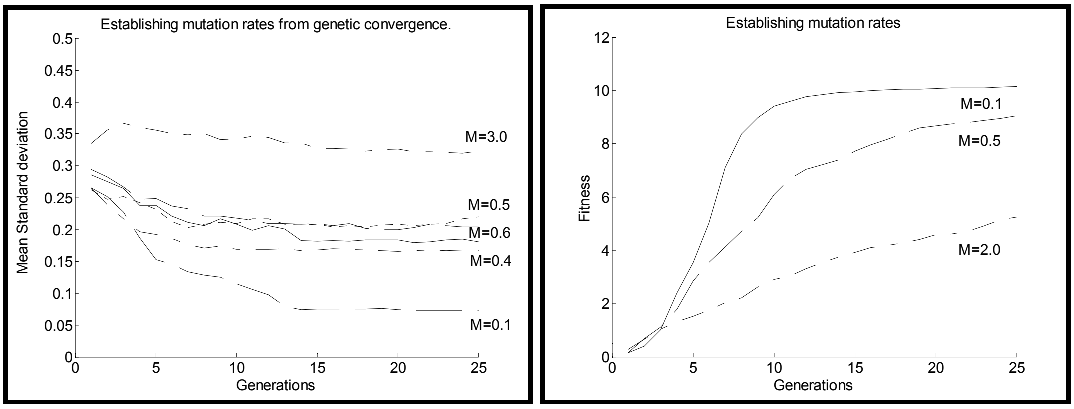


Figure 3.0.1: M = 0.1 shows rapid convergence while M = 3.0 shows disrupted convergence tending towards random search. At M ≈ 0.5 sufficient variation is maintained around the genome values to facilitate the search amongst available neutral space on the fitness landscape. Convergence continues at a much slower rate. Mutation rates around 0.5 show better increase in mean fitness than those higher rates of mutation.

3.1 Avoidance in Feed-forward networks.

The baseline experiment adopted a fixed and non-evolving sensor profile with four distance sensors. Feed-forward networks quickly evolved to perform this task with competence. Evolved populations were then observed on repeated runs both in the niche world and in the support worlds mentioned above. Agents capable of whole circuits emerged as early as generation 7. Best agents generally accrued fitness around 3.5 and populations had converged by generation 100.

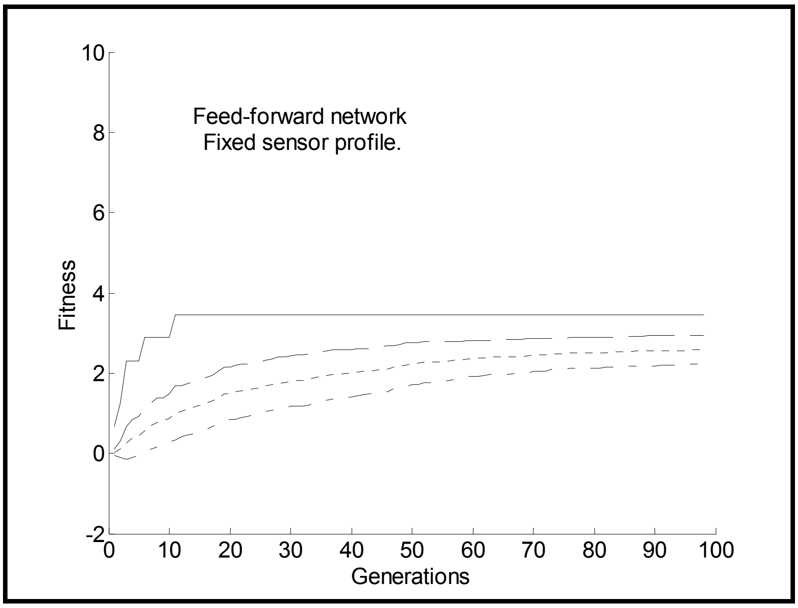


Figure 3.1.1: Fixed sensor feed forward network statistics.

Figure 3.1.1 graphs the standard evolutionary statistics for the fixed sensor feed-forward trial. The solid line graphs the best fitness while the dotted line graphs the population's mean fitness. The dashed lines mark the population's standard deviation around the mean. This convention is adopted throughout the project.

Agents adopt a strategy which attempts to roughly equalise the readings from their sensors, so keeping the agent centred in the corridor as far as possible. In one variant the resting state of the vehicle is stationary; when the sensors are not perturbed by obstacles the agent does not move.

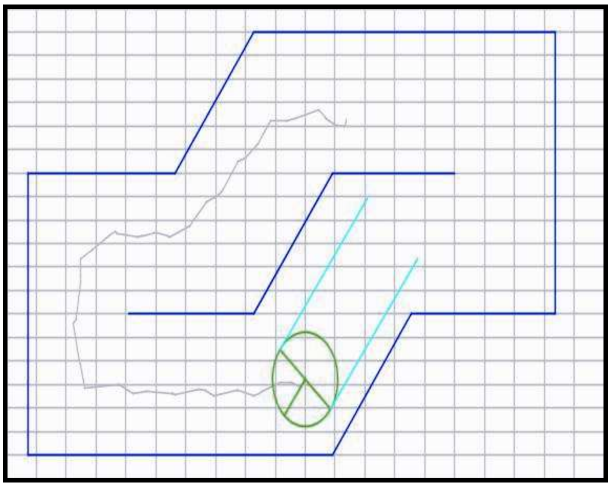


Figure 3.1.2: A typical avoidance trial.

In another variant the agent progresses in a straight line until perturbed, when it swerves. These basic strategies have become known as static and mobile avoiders. A third common strategy involves wall following. Here the agent moves with a slight left or right handedness. Once this slow turning brings the outermost sensor into contact with a wall there is a slight turn in the other direction. This oscillating between the handed turning and the sensor driven avoidance leads to wall following; the agent literally feels its way along the obstacle in question.

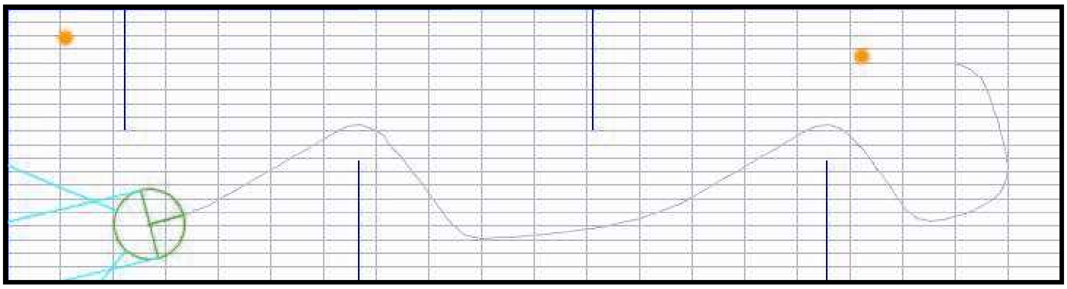


Figure 3.1.3: Wall following uncovered in a support world.

Wall followers too can have a static or dynamic resting state although the static state is characterised by tight circling rather than stillness. This attractor is far less common because the practice of driving forwards with a very slight turn maximises the first term of the fitness function much better than turning and driving forwards on perturbation.

All successful strategies show a handedness; agents would rather circuit the course either clockwise or anti-clockwise. Some agents find themselves pointing in the wrong direction at the start and U-turn at the first unfamiliar corner before continuing as normal. Due to the nature of the two corners in the niche-world, agents are evolved left-handed (anti-clockwise orbits as watched from above) about eight times as often as they are right handed.

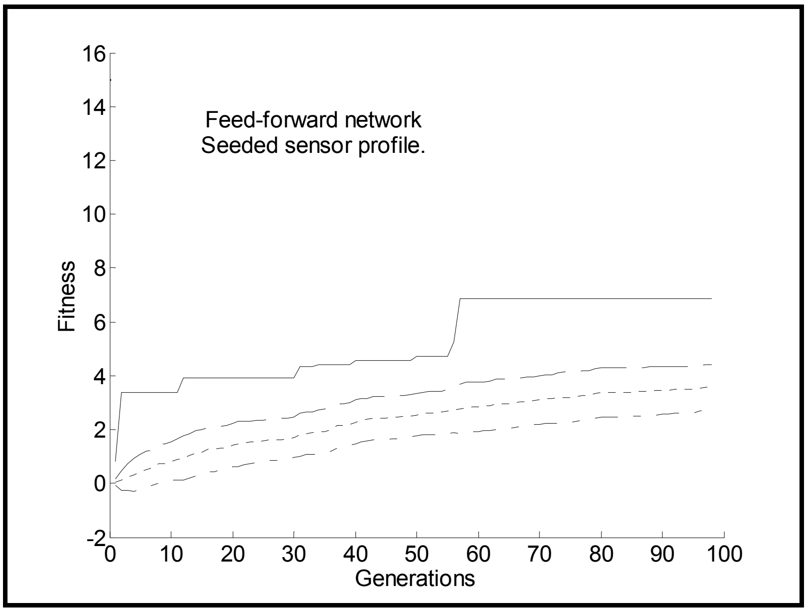


Diagram 3.1.4: Mean fitness per generation along with the standard deviation and best agent's fitness for populations evolved with seeded sensor profiles.

Competent agents (making at least one whole circuit) emerge by generation 7. In generation 20 all three strategies are active within the population. By generation 100 avoider strategies have dominated the gene-pool.

It was immediately clear from observed agents that although the same three basic attractors in behaviour space existed there was a tendency to rationalise the sensor profiles through grouping and shortening.

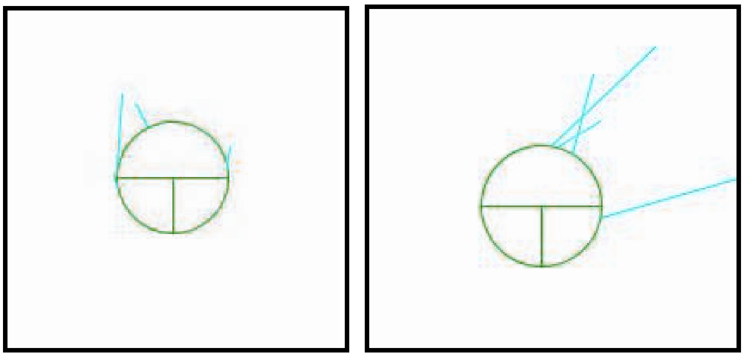


Figure 3.1.5: Sensor rationalisation in evolved sensors.

When grouped, a single agent-environment juxtaposition will minimise several sensors simultaneously. Underpinning shrinkage is the return value of a distance sensor which ranges between 1.0 and 0.0 inversely proportional to the sensor's length. Minimising the sensor's length has the effect of minimising the sensors average reading per average distance in a given environment.

The same general trend was uncovered in the third experiment with fully evolved sensors. Once again task success was quickly achieved but fitness this time far exceeded the previous experiments. The trend was to develop sensor profiles like bristles with perhaps one or two longer distance sensors to facilitate following behaviour. Pure evolution took rationalisation to a further extreme for not having the seeded array to modify through mutation alone. In this way much shorter sensors were selected for so the sensor penalty was drastically minimised.

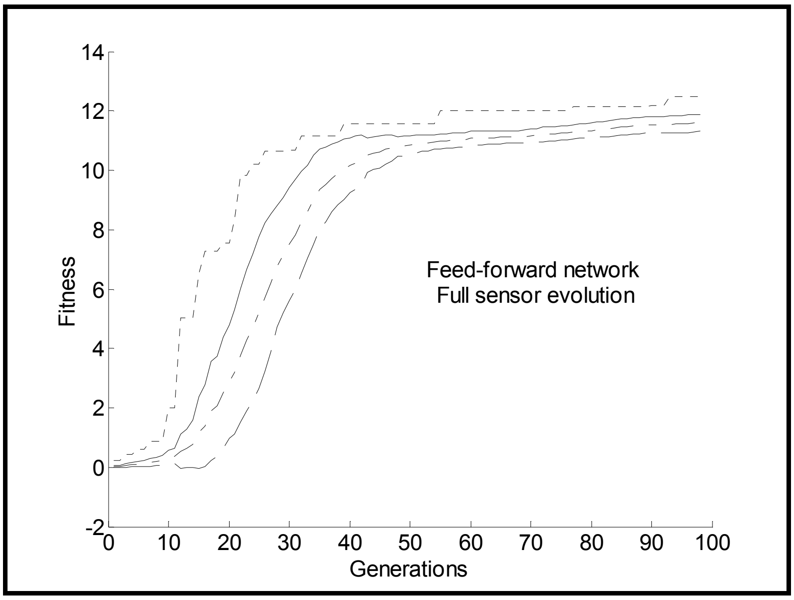


Figure 3.1.6: Avoidance with fully evolved sensor profiles.

Pure evolutionary runs also uncovered a third form of rationalisation. The angle of the sensor becomes extremely acute, essentially laying the sensor along the side of the agent's body. This way a longer sensor would still be perturbed considerably less often than if it were pointed out into the world. The advantage of this is that any perturbation of the sensor can be considered significant to navigation, no threshold is involved. The sensor has become less of a distance sensor and more of a bumper, its reading not a linear measure of distance but rather a Boolean measure of collision.

The shortening of sensor lengths also permitted the development of new strategies: turning almost on the spot and pacing up and down a short length of corridor. This arose because the minimal sensors meant that circling strategies now scored reasonably well despite the associated turning penalty, turning loses were made up for by sensor minimisation.

The graphs of typical evolutionary runs shown above would appear to support the notion that increased plasticity increases evolvability. The maximum fitness of a fully evolved agent is more than four times that of partially evolved agent and this greater fitness is reached in a similar timescale. A further attractor in behaviour space was also enabled, even if this attractor was not a good solution. The extra plasticity allowed the rationalisation of the sensor array favouring smaller sensors covering a smaller area. Such clear results might be interpreted to support the hypothesis; there remains one important consideration.

Sensor rationalisation should properly be considered an artefact of the fitness test:

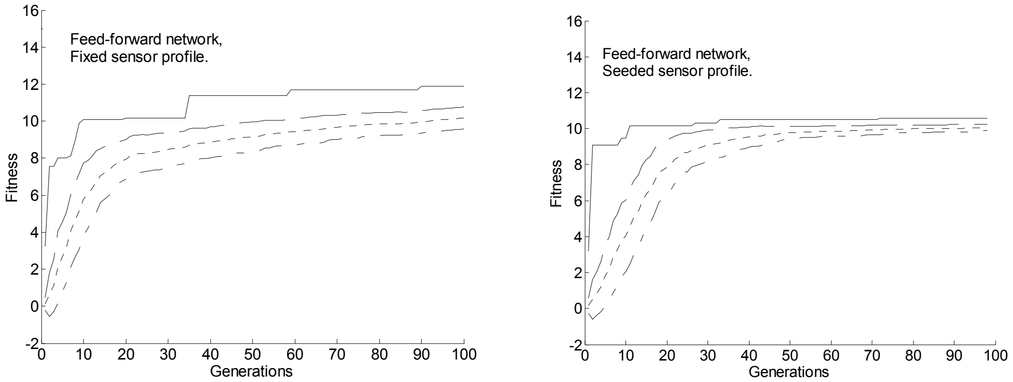
$$\Phi = \sum v.(1-\delta v).(1-I) \qquad \text{(from 8 above)}$$

where the third term requires that the agent minimise its highest distance-sensor reading. Sensor readings can be minimised by avoiding obstacles *or* by adjusting the sensor profiles through grouping and shortening, so evading the task in hand. There is an element of this rationalisation pertinent to the enquiry; the overall mean fitness of a population is greatly increased by ceding evolutionary control to the sensors. It follows that the given sensor array sets hard limits about what it was possible to achieve under evaluation.

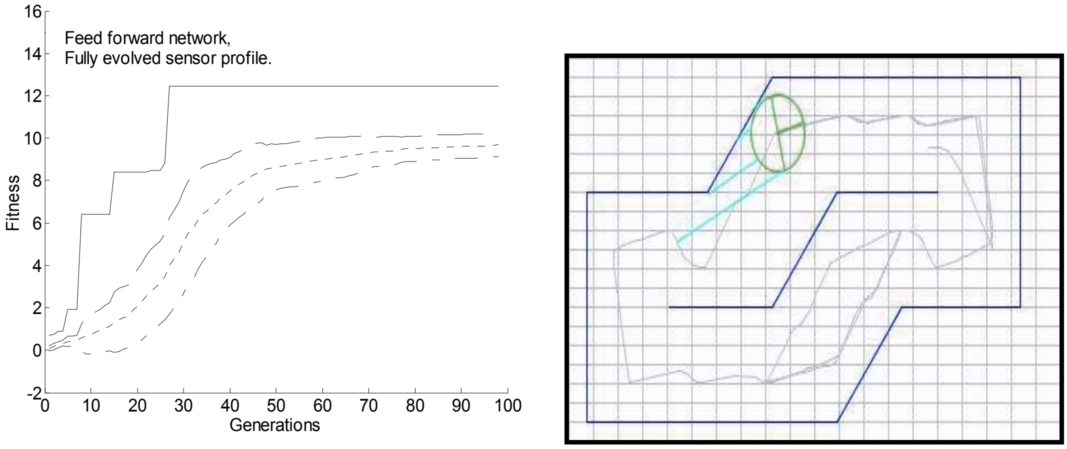
Such explicit elements in the fitness function can be problematic (indeed, in this case *are* problematic). With this in mind the experiments were re-run, agents being evaluated according to:

$$\Phi = \sum v.(1-dv) \qquad (11)$$

where Φ is fitness, *v* velocity and *dv* normalised rate of turn as before. Here object avoidance is implicit in the *v* term. The second set of experiments demonstrated the same four basic behavioural attractors and robust evolution over a similar time scale. Graphs of typical fitness after 100 generations are shown in figure 3.1.7 overleaf.



3.1.7: Evolution of avoidance with fixed and seeded sensor profiles.

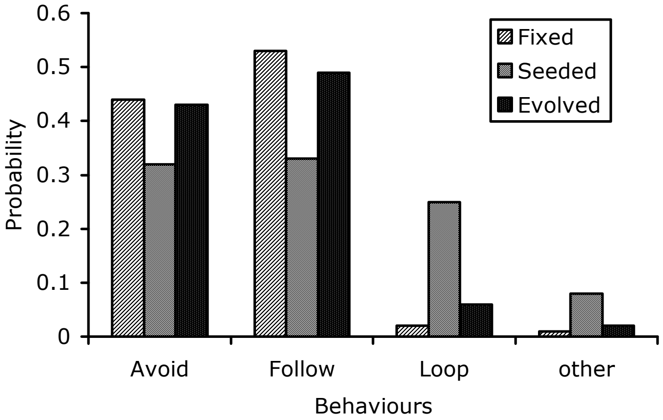


3.1.8: Evolution with fully evolved sensors with sample behaviour.

The highest mean fitness is found in the run without sensor evolution (mean = 10.1968, best = 11.3388) whereas the fittest agent was found using full sensor evolution (mean = 9.7300, best = 12.4443). The populations with seeded sensor arrays did consistently worse than the others. This might suggest that certain sensor combinations do indeed facilitate certain healthier strategies but that the landscape around these fitter attractors is rough terrain. Sensor sets which are near-by in trait space perform less well, making mutation around a well designed sensor set less fruitful. When full control of the sensors is ceded to evolution more distant sensor sets can be evaluated leading to increased performance.

Although mean best fitness was considerably higher in these trials, a significantly higher number of populations converged on the circling and bouncing strategies mentioned above. These dominated early in a population leading to inappropriate convergence.

The statistical probability of an agent adopting one of the strategies was estimated by counting the numbers of strategies found in each run after 100 generations. In the feed forward case 4 classes of strategy were counted: avoiding, following, looping and 'other'. These statistics also suggest that the best results come from either no sensor evolution or full sensor evolution, more poor-scoring looping agents are found in the constrained case.



3.1.9: Distribution of behaviours over 3 evolved populations.

3.2 Avoidance in Dynamic-recurrent networks.

The second set of experiments involved trialling fully interconnected dynamic-recurrent networks with the same three sensor regimes. This class of network brings internal state into the equation allowing temporally extended behaviours to emerge. It also involves far more loci per genotype so increasing the dimensionality of the search space.

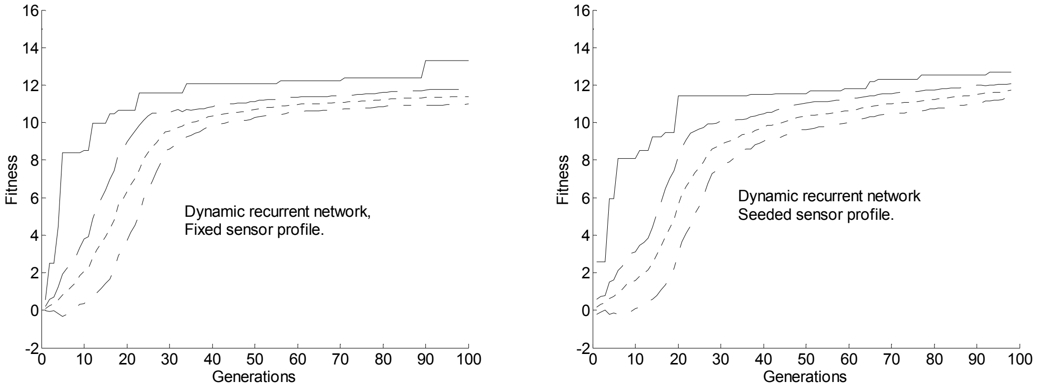


Figure 3.2.1: Typical statistics for evolutionary runs with fixed and seeded sensor arrays.

Once again populations quickly showed competent behaviour although the best individuals in each case were not as strong as the best feed-forward networks. Observed agents showed a tendency to react rather late to sensory perturbation and suffered far higher numbers of collisions on their way round.

New attractors or behavioural traits were also found. In populations with fixed sensor profiles many agents adopted a 'three-point-turn' cornering strategy which involved a short reverse before adjusting the heading.

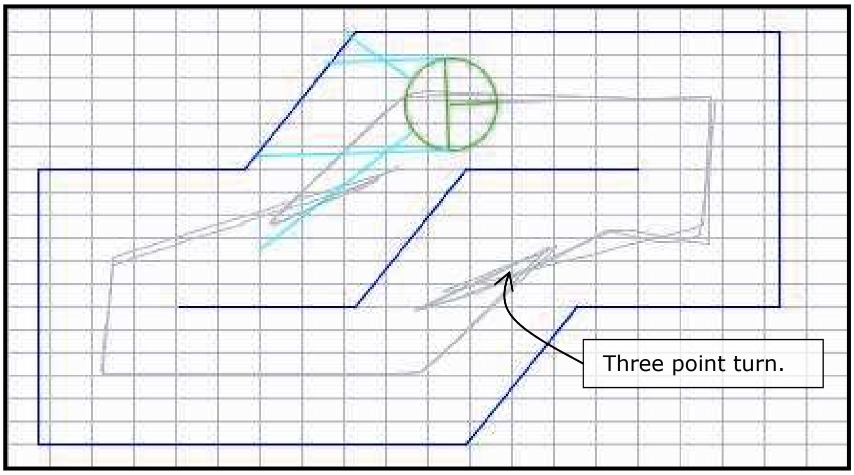
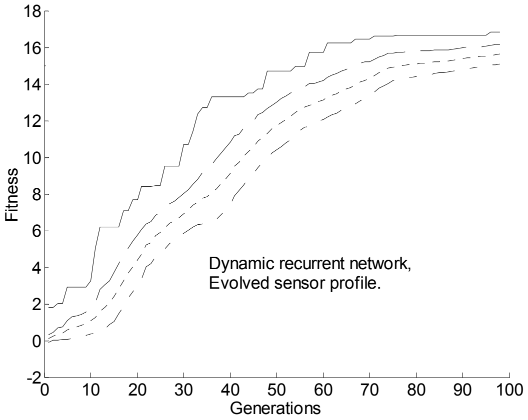


Figure 3.2.2: Example reversing behaviour.

A second less useful trait developed from the temporal capacities of the dynamic-recurrent network. Agents would collide with an obstacle and pause for a while before turning sharply away and continuing. The pause is only observational; on the inside the agent slowly decreases membrane potential in the motor neurons until the forward moment stops and then reverses. The apparent pause is a feature of the banded motor profile. This pausing is symptomatic of 'slow-thinking' in the networks; sensor perturbation takes a few steps to have a significant effect on the network activation as a whole. Neuronal time constants were evolved within a range that should allow faster thinking and indeed other agents did think and react more swiftly. It seems strange to find such suboptimal traits present although they may well have been evolved out of the population over longer time spans.

In the case of fully evolved sensor profiles the difficulty of inappropriate attractors or behavioural traits was more acute. Graphing the evolutionary statistics for a typical run shows slower convergence but eventually finds a much higher best and mean fitness. Yet very many more of these populations converged on inappropriate strategies. This result points to the interesting notion that the desired sort of behaviour is sub-optimal in this fitness landscape. The fittest agents seen to date score around 16.00 and score such through rapid oscillations essentially leading to bouncing along the corridor. Most of these agents do not perform circuits and spend their time going forwards and backwards, scoring optimally for straight line travel. Some do bias their bouncing towards circuiting the course although this is rare. The oscillations are an extreme form of the circling strategy flagged above. In each case the small loss of fitness involved in the corners, ricochets and bounces is not enough to prevent dominance. These strategies are intrinsically more straightforward to develop because they involve less precise tuning of sensory motor invariants and have less 'respect' for their environment. They score well early and begin to dominate the gene pool.



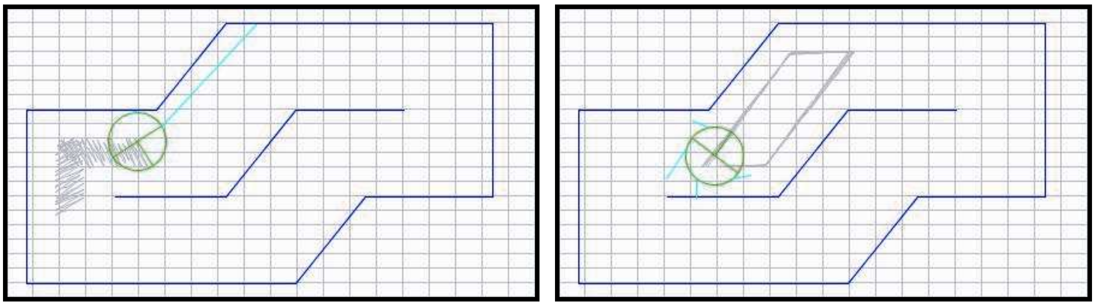


Figure 3.2.4: Inappropriate strategies which dominate the gene pool quickly.

Sensor evolution showed the same tendency as in the feed forward networks with wall followers tending towards a side pointing sensor and avoiders tending towards forward pointing sensors at their extremes. Additionally, dynamic-recurrent agents evolved sensor arrays where one or more sensor had been so far turned towards the agent that they were effectively turned inwards and reporting closest contact at all times. This constant signal was ignored or used as a steady reference signal. The overall effect of this is the same tendency to simplify the problem space as was found in the original trials involving sensor rationalisation.

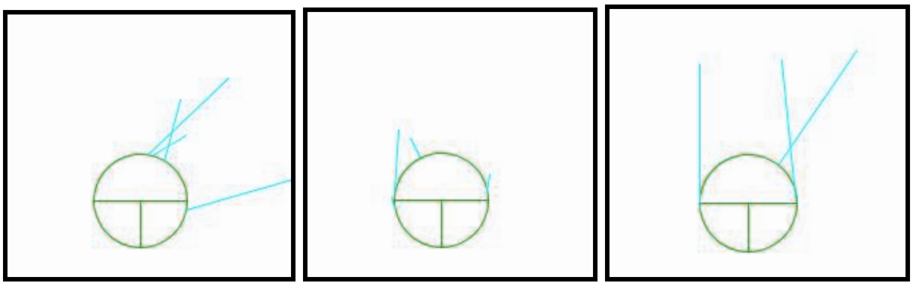


Figure 3.2.5: Sensor rationalisation.

It would seem that evolution is selecting for bare minimum sensor arrangements to keep the complexity of the resulting dynamical system behaviour down. Although parameterising a sensor like this makes no difference to the genotypic search space, there is a simplification in the resulting dynamic system in behaviour space. Fewer variables varying in that state space make for more simple trajectories and simplified behaviour.

It was much harder to estimate behavioural probabilities in the dynamic-recurrent case as there would almost always be a surprise somewhere in each population examined - the range of possible behaviours is vast.

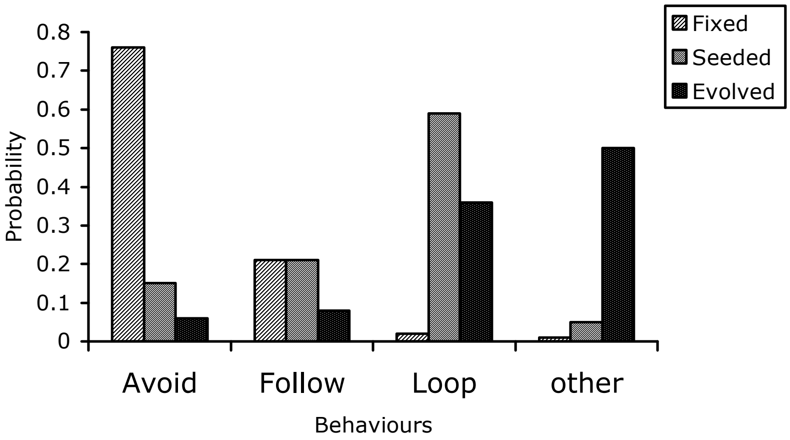
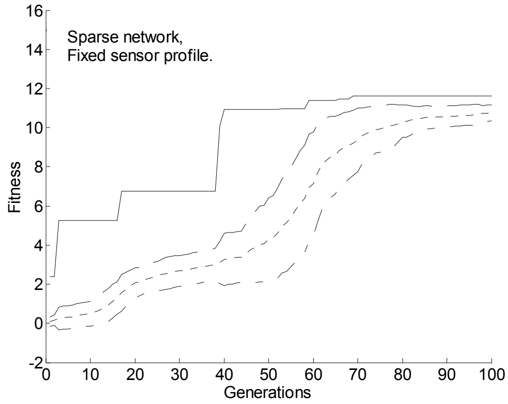


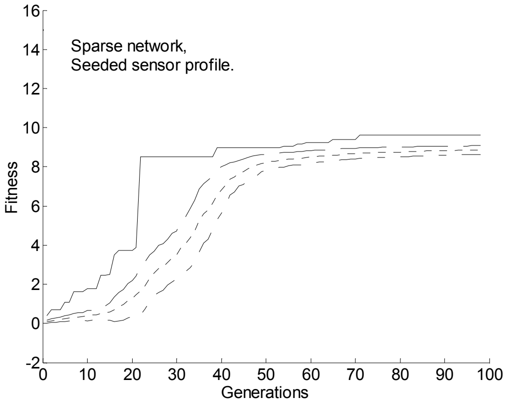
Figure 3.2.6: Distribution of behaviours over 3 evolved populations.

Reversing behaviour was always associated with avoidance strategies. It is remarkable to note how little wall following behaviour evolved in these trials and how well the fixed architecture DRNN does. The probabilities reflect the increased dominance of looping and bouncing in full evolutionary runs.

3.3 Avoidance in Sparse networks.

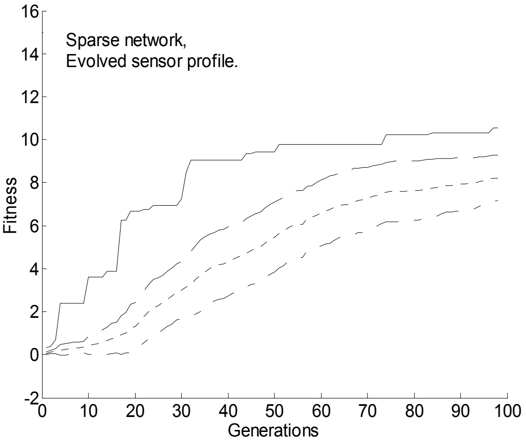
As mentioned above, it was suspected during planning that the dynamic-recurrent network would tend towards rhythmic oscillation as the node count and so connectivity increased. The third set of avoidance experiments involved sparsely connected dynamic-recurrent networks. Typical statistics once again show successful evolution of agents in all three scenarios. In the case of fixed sensor profiles evolution begins more slowly as a network with dynamics appropriate to the sensor profile must be found. Once found optimisation achieves the usual strong result. Agents of both wall following and avoidance behaviours were found in the resulting populations.





Evolution of a seeded sensor profile was faster to get started. The need for sensors and networks to be in some kind of agreement still holds but sensor mutation speeds the time to integration, the network and sensor profile meeting half way. Seeded sensor profiles evolve more quickly but generally score less well. Too often one of the inappropriate attractors dominates the gene pool before anything interesting can happen.

Fully evolved sensor profiles showed slower evolution again. At cessation populations showed competence but had not converged as much as seeded and fixed sensor populations. Running the algorithm for longer periods showed the same overall fitness achieved. Sensor profiles showed the same vague pattern found above with agents evolving a sensor for wall following or sensors for look-ahead. The increasing number of inappropriate strategies encourages the development of rather random sensor profiles.



Amongst the fittest of the fully evolved sparsely connected agents were those who showed the least handed-ness in their turning preferences. As was already mentioned almost all agents adopted a handedness governed by the turns in their world. Diagram 3.3.3 shows a brain in which forms of symmetry and structural modularity have emerged.

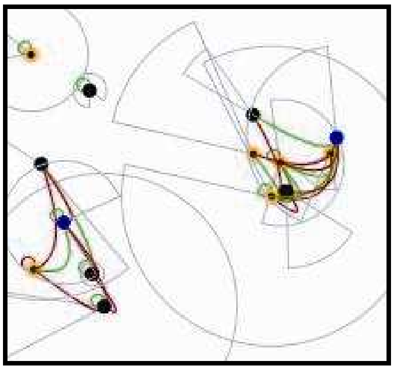


Figure 3.3.4: The birth of symmetry.

The agent using this brain was able to circuit in both directions most of the time. It must be said that these graphical interpretations of the network appear more symmetrical than the associated weight sets suggest.

Again estimation of the behavioural probabilities was difficult, the dynamic nature of the networks making behaviour difficult to qualify at times. The same basic categories were adopted as in previous cases.

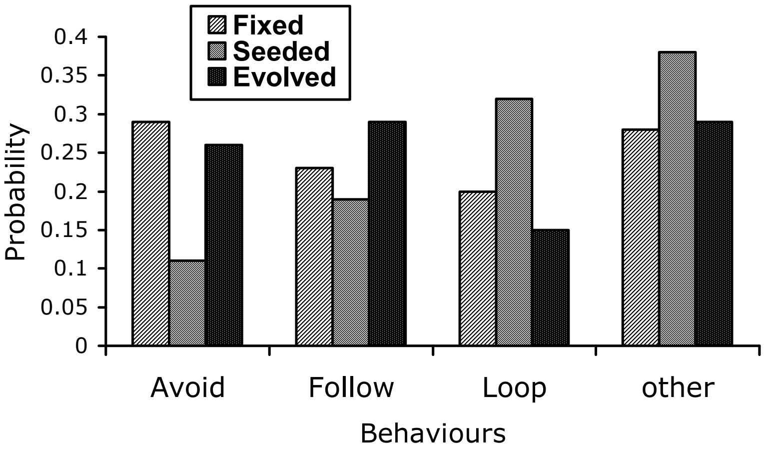


Figure 3.3.5: Distribution of behaviours over 3 evolved populations.

These probabilities reflect the greater variety of behaviours found in the dynamic networks. The distributions once again suggest either full or no sensor evolution is preferable to seeded sensor evolution.

Having examined a simple case the same set of variations were to be tested while attempting to evolve detour behaviour; sparse and fully interconnected dynamic-recurrent nets were to be used, trialled with and without sensor array evolution.

A two step approach was adopted where agents would be evolved for general phototropism in a cluttered environment. Competent agents could then be transferred to the niche world for further optimisation.

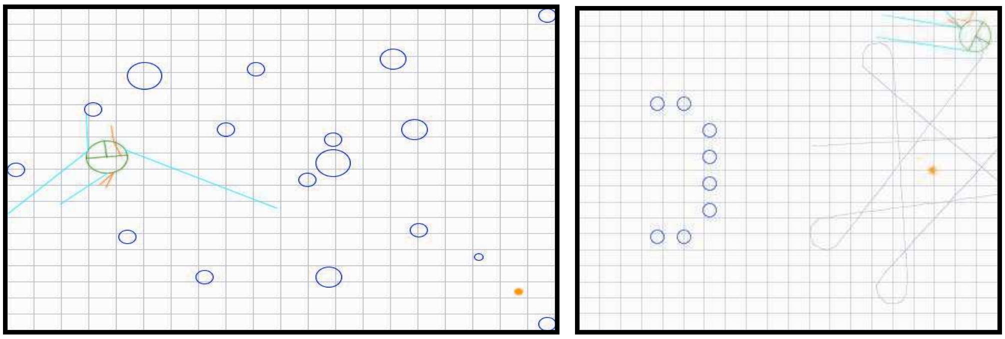


Figure 3.0.3: two steps to detour behaviour.

Mutation levels were established as before by examining the mean standard deviation per loci per genome. Due to the increased complexity of the task space and increased dimensionality of the search space graphs of mean standard deviation are less clear. It takes longer for the algorithm to find a solution with greater than latent ability, once found such a solution rapidly dominates. Figure 3.0.3 shows mean standard deviation and fitness for mutation rates between 0.05 and 1.

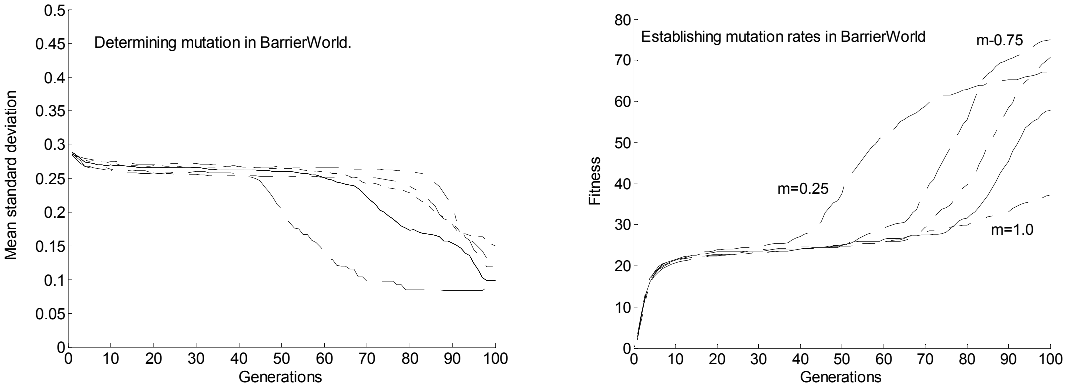


Figure 3.0.4: Establishing good mutation rates.

An initial rate of 0.75 was selected for these trials giving a balance between convergence and high fitness.

The fixed sensor array was the same as that used above with the addition of two forward pointing directional light sensors with an angle of acceptance of π/3 radians. Six sensors require a minimum of 8 fully connected nodes yielding a chromosome length of 88.

3.4 Detour in Dynamic-recurrent Networks.

The following figures graph evolutionary statistics for fixed, seeded and evolved sensor profiles in typical evolutionary runs along with typical strategies evolved. Interesting agents evolved in all cases.

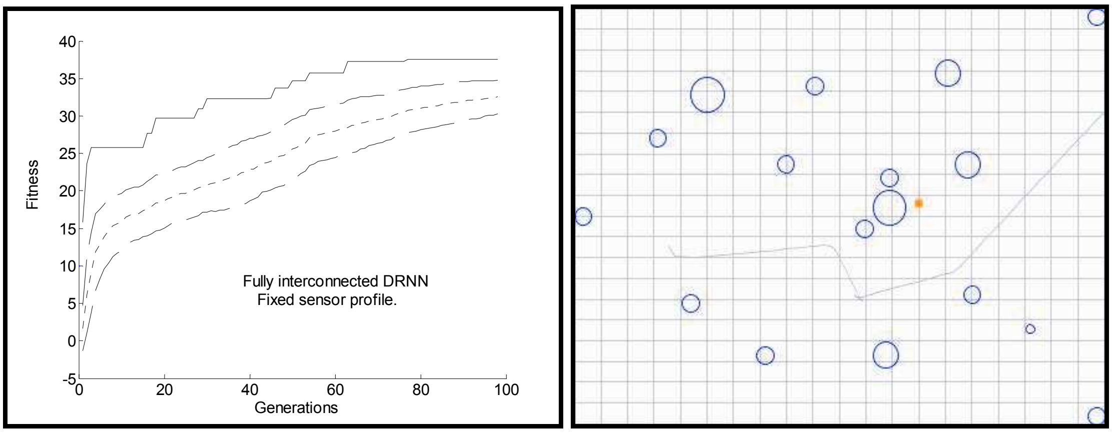


Figure 3.4.1: Evolution of detour with fixed sensors in cylinder-world.

In the case of fixed sensor profiles agents evolved to roam around their environment avoiding the cylinders with great success. Unfortunately, as the diagram shows, the majority of agents had no significant phototropic response.

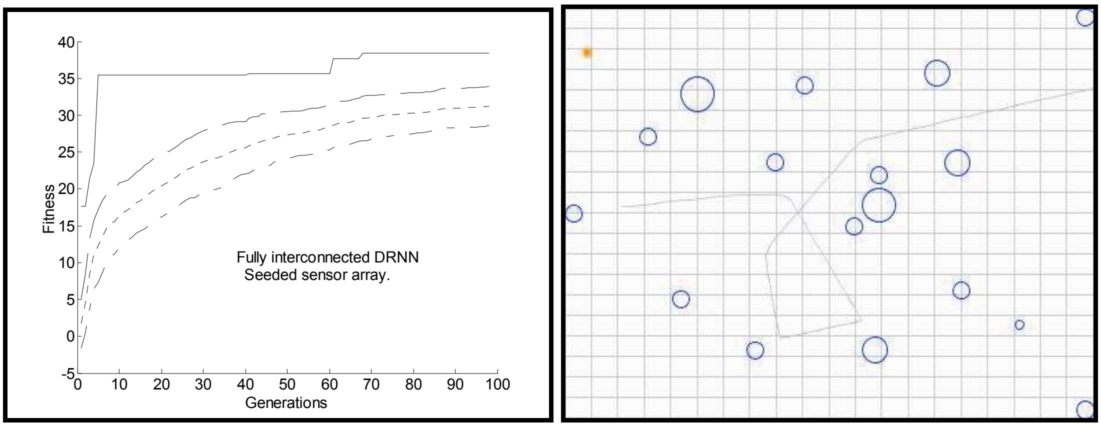
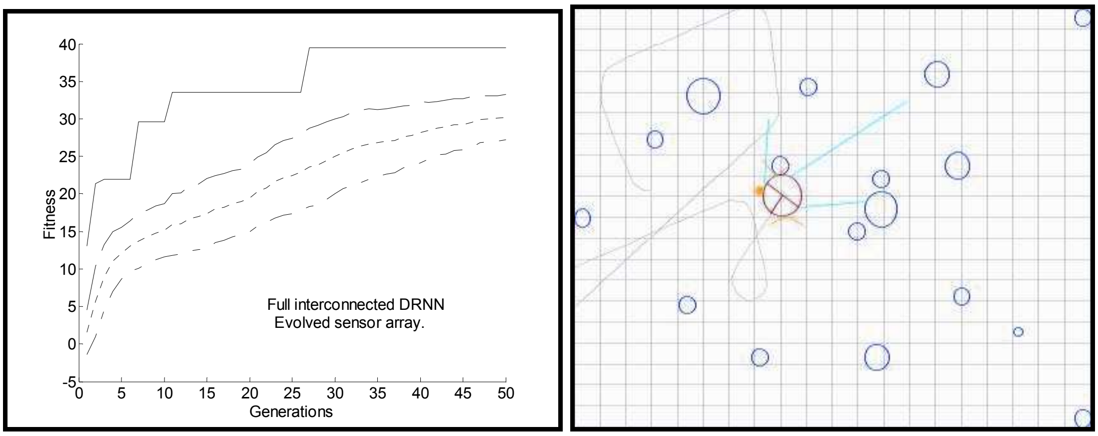


Figure 3.4.2: Evolution of detour with seeded sensors in cylinder-world.

Significantly, some populations with seeded sensor profiles converged on phototropism without object avoidance. Such populations scored poorer overall fitness to match the rather hit or miss nature of their strategy. It would seem, given the dual nature of the fitness test, that there are local optima representing both tropism and avoidance behaviours. We are looking for tropism optima situated on top of an avoidance optimum.

Such a fortunate juxtaposition was found in the case of agents with fully evolved sensor profiles. In these populations, agents evolved to place their eyes in both forwards and backwards positions. This increases the scope of the agents' vision and so must increase the profitability of attempting phototropism as well as object avoidance.



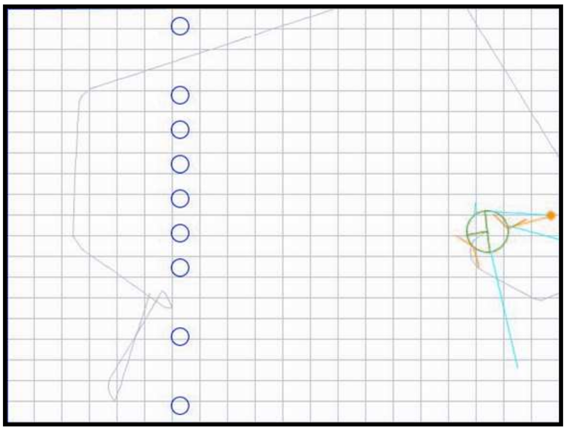
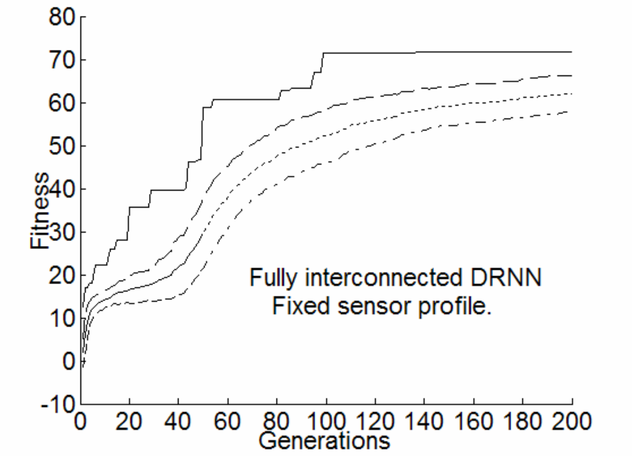


Figure 3.4.3: Successful agents showed remarkable ability across all the trial worlds in which they were tested. The figure on the left shows the agent escaping from a constrained enclosure in a fence world and successfully finding the light. It is worth mentioning that although these agents are winners in all fitness tests they are so for the wrong reasons. The great fitness breakthrough comes from evolving a rearward pointing eye. This allows the agent to see when it is going the 'wrong' way and turn around. A combination of rushing blindly in the forwards direction with good object avoidance and the ability to turn having gone too far makes for a very successful agent. The path in the cluttered world shows the agent hunting around with a bias towards the light; this is too chaotic for detour behaviour.

These agents were destined to be further evolved in the target niche world to perfect their combined behaviour. Unfortunately, the nature of part evolved solutions made this a frustrating and ultimately unhelpful practice. In the case of the earlier populations which had evolved towards either tropism or avoidance optima further evolution created a mess. Agents had their accumulated fitness scores zeroed for the new trials and being largely optimised for either avoidance or tropism they faired very poorly. The generally bad fitness scores led to chaotic breeding and the overall capabilities of the populations fell even when returned to their original world. The fully evolved populations that had succeeded in the cluttered world also triumphed in the fenced worlds, with no net change in behaviour for the change in environment. Although these agents solved the problem they did not detour according to the previous strict definition.

Given that the incremental approach had failed, agents were evolved directly in the niche world under the same fitness test. Once again the usual three variations in sensor morphology were trialled; statistics for each evolutionary run are given along with sample evolved behaviours.

Figure 3.4.4: Evolution of detour behaviour with a fixed sensor profile.



The most common behaviour found in these trials was a form of looping behaviour. The agent circles, maintaining a certain distance between themselves and any obstacles, if the light sensors become stimulated the agent travels forwards. This way the agents can locate a light and progress towards it, any obstacles initiate the distorted looping which causes the agent to roll around them before relocating the light and continuing. This sort of ultra-efficient looping strategy is often found in evolutionary robotics and was a design feature of the earliest biologically inspired robots: Grey-Walter's Elmer and Elsie (Grey-Walter 1950)

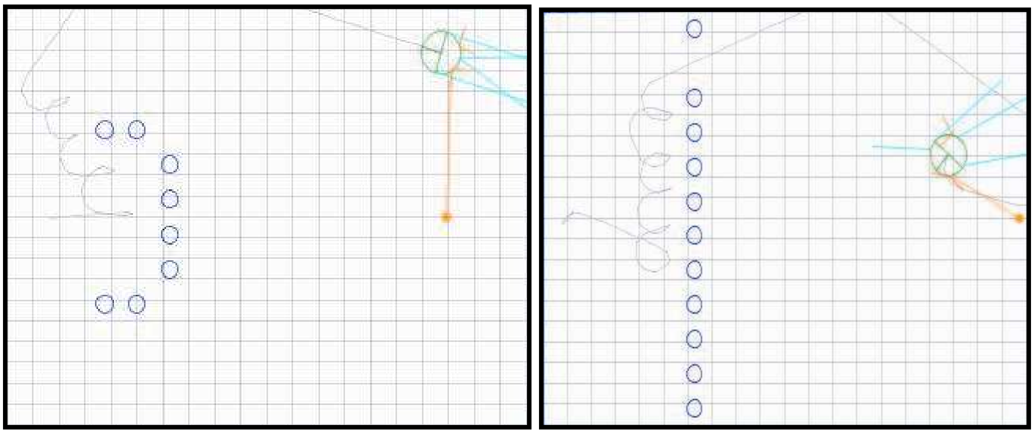
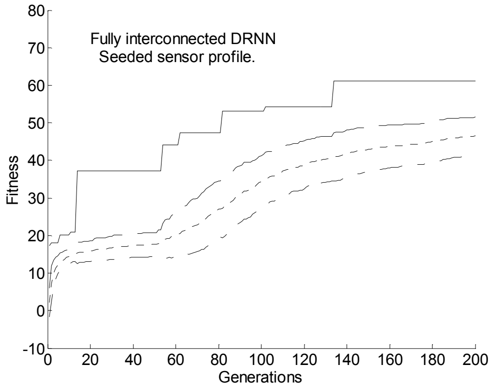
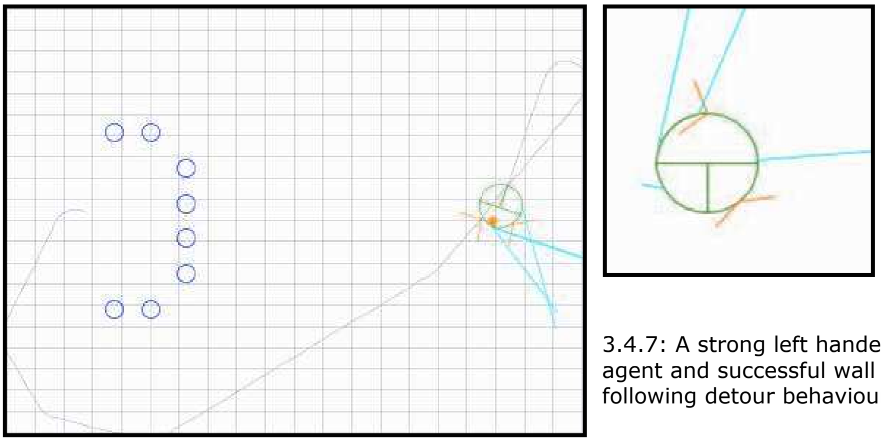


Figure 3.4.5: Looping strategies.

Seeding and evolving the sensor profiles led to very much more variable evolutionary runs. The fittest agent was evolved in this class with a sccore over 86, at the same time many runs ended with converged populations scoring as little as 55. The fittest agent employed one of the aforementioned looping strategies coupled with sensors tuned to strong left-handed ness. Handedness was as common amongst these agents as in previous experiments and, as was found in previous experiments, left handedness was far more common. Unlike the previous experiment there is no environmental bias to account for this and



so no straight forward explanation. A second very sucesfull strategy employed this handedness in wall following. Thus, when the obstacle was detected, the agent would follow its perimeter untill the light once again took over navigation.



3.4.7: A strong left handed agent and successful wall following detour behaviour.

Full evolution of the sensor array led to a very similar set of results with variable best fitness across numerous runs. Agents generally adopted the same looping strategies with varying degrees of success. The more elegant wall following solutions were less often found in fully evolved trials.

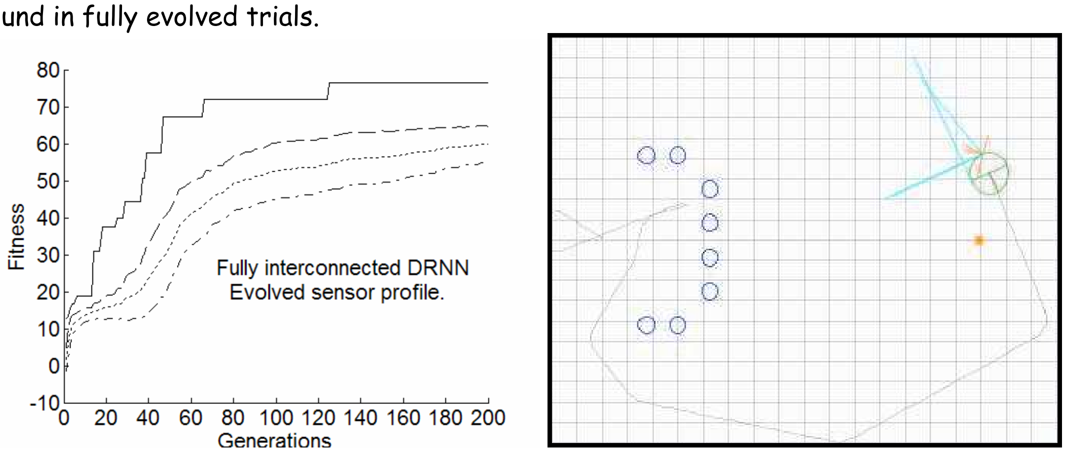


Figure 3.4.8: Detour evolution with fully a evolved sensor profile and a sample agent.

Although the looping behaviour and the slightly more refined wall following pictured here are undoubtably high scoring and consistently successful strategies it is not clear that they would be accepted as detour behaviour. A full analysis of these agents and their behaviours follows in the next section.

3.5 Detour in Sparse networks.

In this final set of experiments the same sets of sensor arrays were trialled with sparsely connected dynamic-recurrent networks. Once again a two step approach had been envisaged and agents were begun in the cluttered environment. Figure 3.5.1, below, graphs the evolutionary statistics for typical runs in this series.

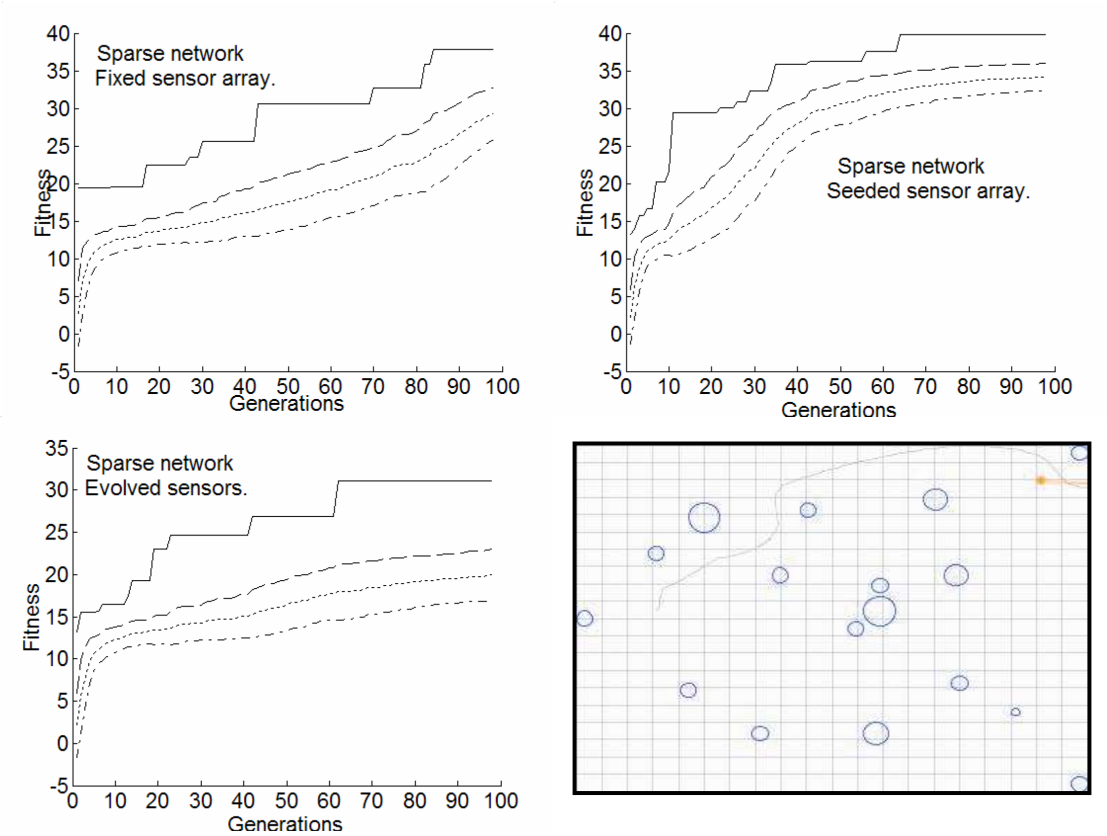


Figure 3.5.1: Evolutionary statistics for detour in sparse networks with fixed, seeded and fully evolved sensor profiles along with a sample path through cylinder world.

Performance of the sparse network encodings was once again poorer than the fully interconnected encoding across all trials. Populations tended to evolve either avoidance or tropism as was previously uncovered, the probability of avoidance behaviour being very high. In the case of populations sporting fully evolved sensors the overall performance was significantly slower with many populations not evolving either behaviour to a capable minimum within the usual time span. Populations evolved for longer fared much better and it was within fully evolved populations that the first signs of a more integrated behaviour were found.

The incremental evolution of such poor behaviours met with the same bad results as the previous attempt. Populations did not hone allready present skills, zeroing of their fitness for the new world led to chaotic breeding and resultingly lower fitness means.

The final experiment involved sparsely connected networks evolved directly in the barrier detour world, evolutionary statistics being graphed in figure 3.5.2 below. The same pattern emerged once more with fastest convergence in the seeded populations and much slower evolution in both fixed and fully evolved populations. The dominant strategies involved looping and wall following as found in previous generations with sensor profiles tending towards supporting the handedness found in the behaviour.

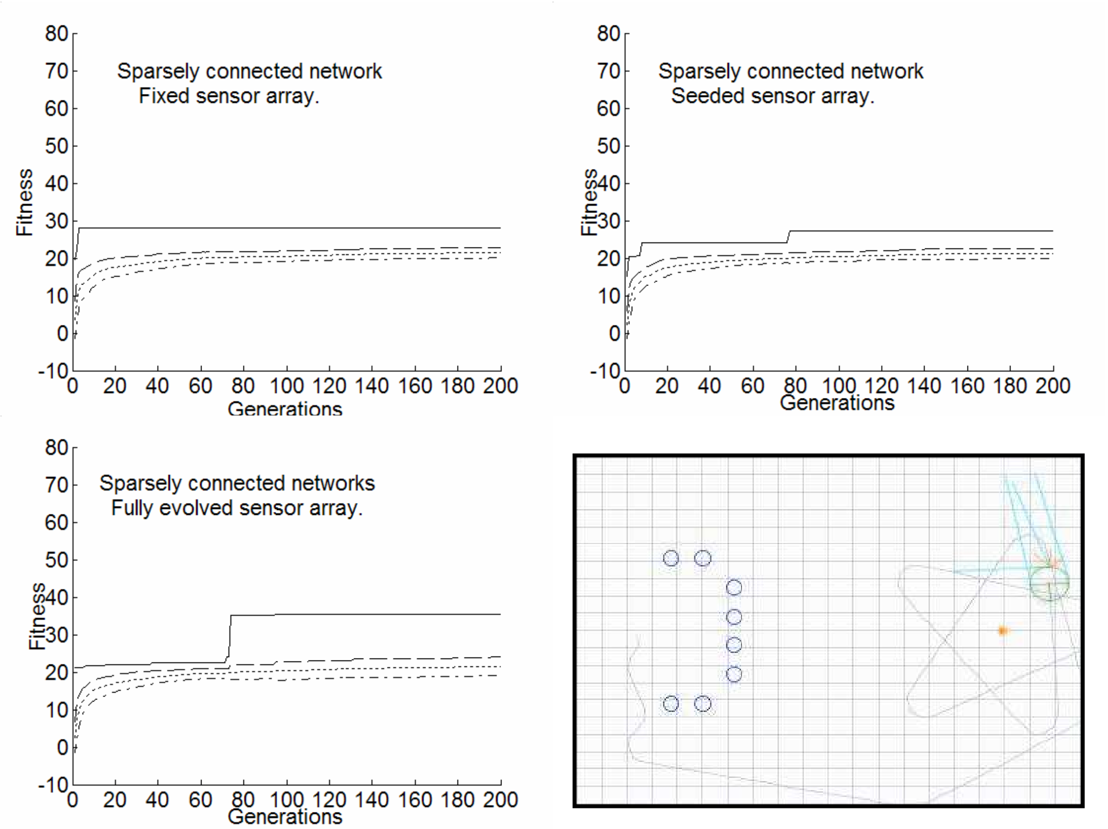


Figure 3.5.2: Typical evolutionary statistics for detour in sparse networks with fixed, seeded and fully evolved sensor profiles along with the best seeded agent's path through barrier world.

Agents from these populations scored considerably lower mean and best fitness. The mean is partially explained by the slower evolutionary progress found in all runs involving sparse networks. Despite the capable run shown above, the best agents in these runs were only capable of good behaviour in less than half of their trials. Success depended as much upon fortuitous starting conditions as real ability. Once semi-capable agents had emerged they quickly dominated the gene-pool as the graphs suggest.

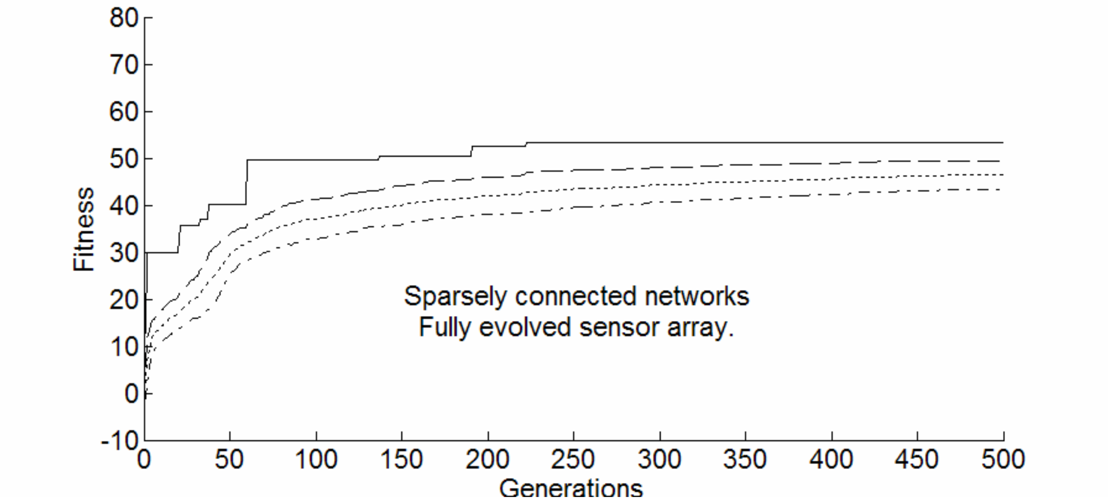


Figure 3.5.3: One of the better sparse network populations
Even in this longer run the population shows convergence gy generation 200.

4 Analysis:

Data gathered from the 15 experiments is grouped into a number of more digestible strata; there were avoidance and detour fitness tests, each involving classes of network trialled with different sensor profiles. Questions will be asked of the data on two levels: the ease and speed of evolution can be explored at the population level with evolutionary statistics, while the appearance of new and more complex behaviours must be sought amongst populations at the level of individual agents.

4.1 Evolution of Avoidance Behaviour.

The mean, standard deviation, best and worst individual fitness scores were recorded for each generation of the evolutionary runs. When combined with 'field notes' attached to each of the runs during observation and analysis the following observations were made.

- Sensor evolution:
  1. The highest mean fitness was consistently found with fixed sensor profiles.
  2. The highest individual fitness was consistently found amongst evolved-sensor agents.
  3. Seeded populations converged more quickly upon poorer strategies, scoring least well across all trials.
  4. Evolved-sensor populations were slowest to evolve.
- Network Morphology:
  1. The most reliable evolution was consistently found in feed forward populations.
  2. The most complex behaviours evolved in sparse networks.

The extent to which these propositions support the hypothesis is unclear. We would argue that additional domains of plasticity do indeed increase the explorative potential of evolution but that this comes at a cost - the search is more difficult, the results more unreliable. Critical to explaining these results is the realisation that the control architecture and the sensor profile are involved in a reciprocal co-evolutionary dance. Ideally each slowly enables the others best expression but this first requires that these two domains fall into step. In the genetic algorithm both have individual crossover and mutation operators separately parameterised. This can make for a fragile coupling.

In the case of controller evolution with a fixed sensor array there is no co-evolutionary dynamic. Evolution progresses to search out and optimise the best possible strategy under the given circumstances. In such a traditional run sensors and motors are effectively part of the environment of the controller; the agent as a whole does not evolve. Good behaviours emerge straightaway and the populations tend towards convergence.

Populations with fully evolved sensors develop more slowly. In part this is because the dimensionality of the search space is increased and in part the co-evolutionary dynamic requires that the controller and the sensor array become structurally synchronised. The additional dimensionality of sensor evolution means there are many more possible combinations to search before a reasonable combination is found.

Once bootstrapped, the fully evolving populations achieve the highest fitness scores. Best agents are able to utilise sensor evolution in developing better strategies. It has been argued (Conrad 1990, Bongard and Paul 2001) that additional morphological parameters can enable an extradimensional bypass between distal fitness peaks within a complex landscape. Such a bypass permits a population to migrate across an otherwise impassable valley in one dimension by following a ridge in another dimension, in this case provided through sensor space. It would be tempting to believe that is what enables the consistently higher fitness in fully evolved populations but this should await further exploration.

Seeded populations did not fare as well as either fixed or fully evolved populations with a tendency to converge more quickly upon less fit strategies. We suggest that the sensor morphology landscape is hilly with many possible sensor combinations proving viable. When these hills are woven into the controller landscape, as co-evolution surely does, it makes for profitable search along many more lines. The co-evolved fitness landscape now has many more basins of attraction leading to more behavioural trajectories. It would appear from the statistics that many of these additional attractors are less than optimal. Further, it would explain the observed poverty of the seeded case if good attractors are surrounded by poor attractors. This would make the seeded case doubly disadvantaged: it would have the slower start of a co-evolutionary run combined with a constrained search in sensor space.

Statistics derived from counting attractors are insufficient to support such grand claims although they do add further support to the observations. Only an exhaustive search of the 75 dimensional fitness landscape could determine such detail. Limited trials were made using fit controllers from fixed profile experiments with sensor-only evolution. Net gain in fitness after a further 100 generations of seeded sensor evolution was less that 1%, a figure best accounted for by noise. If there were better peaks in sensor space nearby this exploration should have found them.

Varying the network architecture followed interpretably similar lines. The increasing complexity of the search space which comes with increasing plasticity slows the evolutionary process. Fastest evolution of the most robust agents occurs in the simplest populations. However, most interesting behaviour and fitter agents emerge from populations with greater plasticity. The best agents, with least handedness and most efficient lines evolved in sparsely interconnected networks. Once again the additional plasticity increased the number of possible solutions which in turn further hindered evolutionary progress; there was faster convergence on poorer strategies. A poor strategy only has to be a little better that the latent potential of a random population to dominate in a co-evolutionary dynamic because of the additional time required for good co-evolutionary partners to get into step.

At the behavioural level, increasing network plasticity did enable the larger range of possible strategies with such additions as reversing or delayed reactions. As has already been suggested, this increase in attractors in behaviour space makes for less robust evolution with many more populations converging on poor strategies. Yet amongst these many local maxima lie the best agents with some really superb strategies.

Increased sensor plasticity was utilised more creatively by the evolutionary process. In both sets of avoidance trials sensor rationalisation was found with sensors being turned inwards and sparse nets evolving with many sensor neurons disconnected. This demonstrates a tendency for evolution to simplify the situation as far as possible. A similar result was seen in Balakrishnan and Honavar's work with Teller's box pushing task (Balakrishnan and Honavar 1995, Teller 1994). Fully evolved agents reduced their sensors from eight to as few as three with no loss of fitness. Husbands *et al.* found a similar trend towards very simple visual systems, even where no direct selection pressure is applied (Husbands *et al.* 1996, Husbands 1998). We suggest an implicit pressure exists to keep the system dynamics as simple, and therefore stable, as possible.

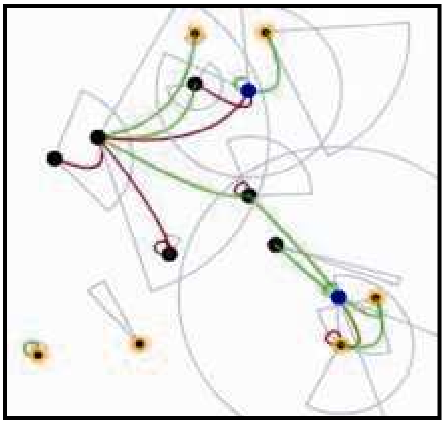


Figure 4.1.1: Sensor dissociation.

At the co-evolutionary level sensors were seen to migrate to support the 'handed' bias found amongst all agents. Similar coherent sensor profiles were seen amongst wall followers and avoiders.

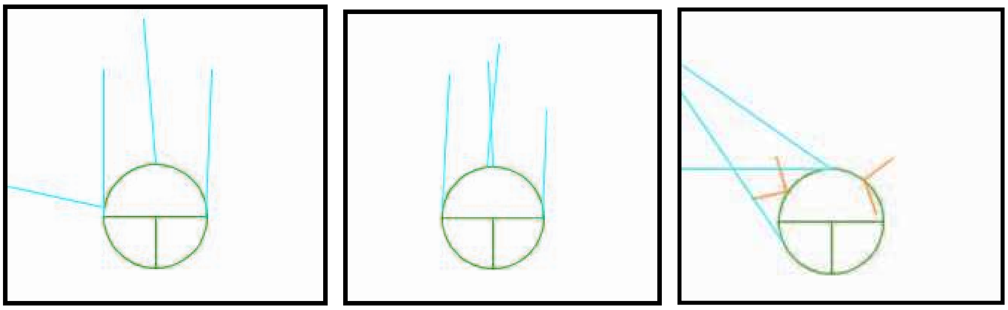


Figure 4.1.2: Wall following, avoiding and handed sensor profiles.

It is not clear that the pictures above represent anything more significant than a bias in the observer for recognisable traits, as many strange and innovative sensor profiles also emerged.

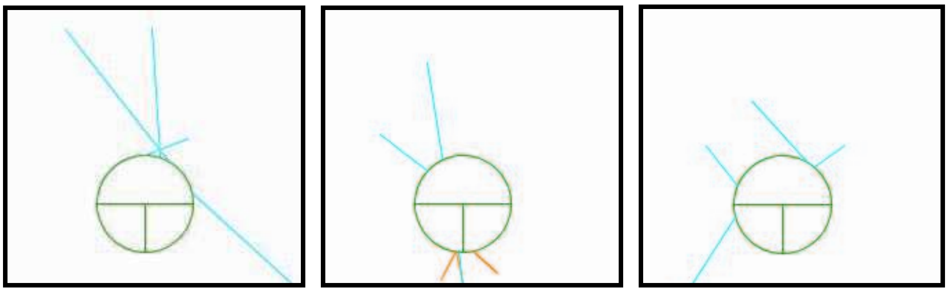


Figure 4.1.2: Less obvious sensor profiles.

4.2 Evolution of Detour Behaviour.

The more complex task environment led to more complex results - less clear in interpretation, less general in their scope. Following the pattern of the previous analysis, statistics and observations suggested:

- Sensor evolution:
    1. The highest mean fitness and best individual was consistently found with seeded sensor populations.
    2. Evolved-sensor populations were slowest to evolve and obtained the poorest best fitness and mean fitness.
    3. Evolved-sensor populations converged more quickly upon poorer strategies.
- Network Morphology:
    1. The most reliable evolution and best results occurred with fully interconnected dynamic-recurrent networks.

At least in a first pass these results do not tally with those of the previous experiment. Here seeded populations excel over fixed, and simple networks over the complex; only the evolution of fully evolved populations remains as it was: slow and unreliable.

It is likely that the fully interconnected networks evolve more quickly than the sparse networks simply because the space of possible networks is skewed. Sparse and full connection regimes contain the same number of possible networks but the sparse case has a great bias towards zero weighted links. The potential of having insufficient connection or critical disconnection (of a motor) is higher yielding an increase in volume of dysfunctional networks to be explored and disregarded. Hence the increased co-evolutionary synchronisation delay in this case; evolution is markedly slower to get started. This much is in agreement with previous results.

It was certainly possible that the features of the sparse encoding used such as the sinusoidal weight generation were innovations unsuitable to evolution. To test this, two other encodings, including one based directly on the two sector technique employed in Husbands (1998), were trialled. Similarly poor results were had in all cases. The sparse encodings, although considered a possible domain of plasticity in respect of network architecture, were partly inspired by the possibility of excess oscillation in the fully interconnected networks. Yet the fully interconnected networks behaved very well.

We suspect that the unusual success of the seeded populations has a simple explanation. In the avoidance trials the seeded array happened to prove among the better of its neighbours. Search around the seed did not show up better arrangements nearby. For detour experiments the seeded profile was not so good. Both the light and distance detectors were forward pointing whilst successful agents always evolved a sideways slant. If the natural affordances of the fixed array was higher in avoidance than in detour trials these results would be expected.

Once again the additional plasticity enabled more attractors in behaviour space. It was not possible to divine what behavioural differences were grounded in architectural plasticity; the behaviour of dynamic neurons in these numbers is often too complex and behaviours did not emerge in sparse networks worthy of detailed analysis.

The evolution of sensors made a much clearer contribution to possible strategies. The best examples of this were seen in the cluttered environment experiments where the best agents evolved rearward vision. It was a feature of this world that heading away from the light was far more likely than heading towards it, most paths leading away from the centre of the space. Good agents evolved eyes in their backs or wide angled vision to cope with this. Without sensor evolution phototropism was largely futile, agents evolved single modality behaviours, avoidance accruing by far the greatest fitness and so evolving most often. The case of the evolution of rearward pointing vision is better evidence for the utilisation of extradimensional bypasses. The higher fitness peak enabled by the evolved sensor profile could not have been attained without sensor evolution.

4.3 Domains of Plasticity and Evolvability.

The original hypothesis: that additional domains of plasticity increase evolvability, has not been clearly supported by the experimental data. The data does show that increased plasticity leads to an increased number of attractors in behaviour space. The data also suggests that attempting to reap that reward is a difficult business.

The increased number of attractors in behaviour space is evidenced by the richness of possible strategies in sparse networks with evolved sensor profiles when compared to fixed feed forward agents. These attractors are enabled by both network architecture and sensor plasticity as demonstrated by reversing manoeuvres or rear-ward facing vision. It would appear that sensor plasticity also enables evolutionary simplification of the problem space through rationalisation and may enable extra dimensional bypasses between fitness peaks.

Yet the inclusion of all these extra possibilities within the search space leads to a greater number of populations converging on poorer strategies. Not all the new attractors are helpful to the task in hand. Poorer strategies with above latent fitness are found more often in the early stages of a run. If no significant improvements are found quickly these solutions dominate the population. Thereafter even prolonged neutral search by mutation does not improve overall fitness.

The synchronisation of co-evolutionary partners slows evolution in the early stages of a run and makes a population vulnerable to converging on a strategy with just above latent fitness. If populations are fortunate enough to get beyond this, there is evidence that both sensors and controllers become structurally coupled and achieve higher overall fitness.

Additional domains of plasticity increase the range of possible behaviours open to exploration. In the sort of limited trial with small populations undertaken here this generates as many difficulties as it reaps rewards; evolution is slower and less reliable although better agents can be found.

4.4 Detour or not Detour?

Comparisons of observable behaviour in the support worlds suggest good agents employ two simultaneous strategies: a simple photophilia and a more complex wall following/avoidance technique. The photophilia involves keeping a light at the periphery of vision, the agents can then spiral towards the source. Agents will perform clean phototaxis in a single light environment; progress in a straight line untill near obstacles in dark environments. Upon detection of an obstruction the agent will follow its perimeter untill a source is re-sighted. If no source is sighted wall following will continue. The best agents pan left and right to facilitate wall following with the discontinuous obstructions. Some agents also show sleek reversing behaviour if caught in a tight spot.

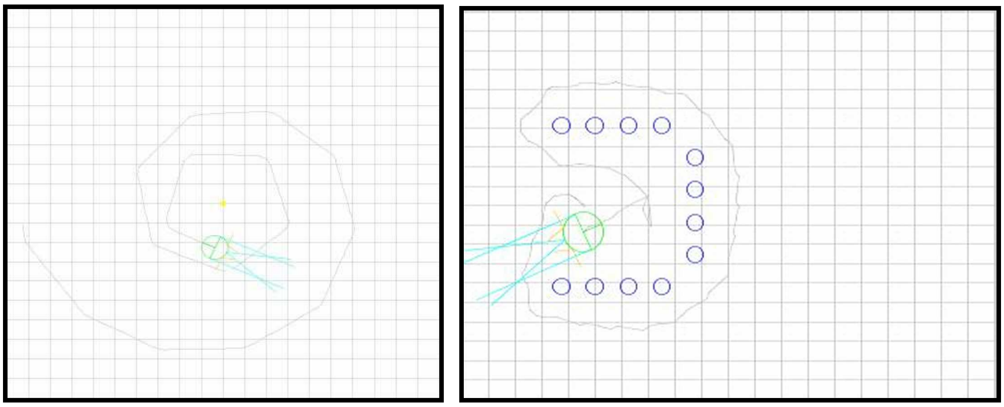
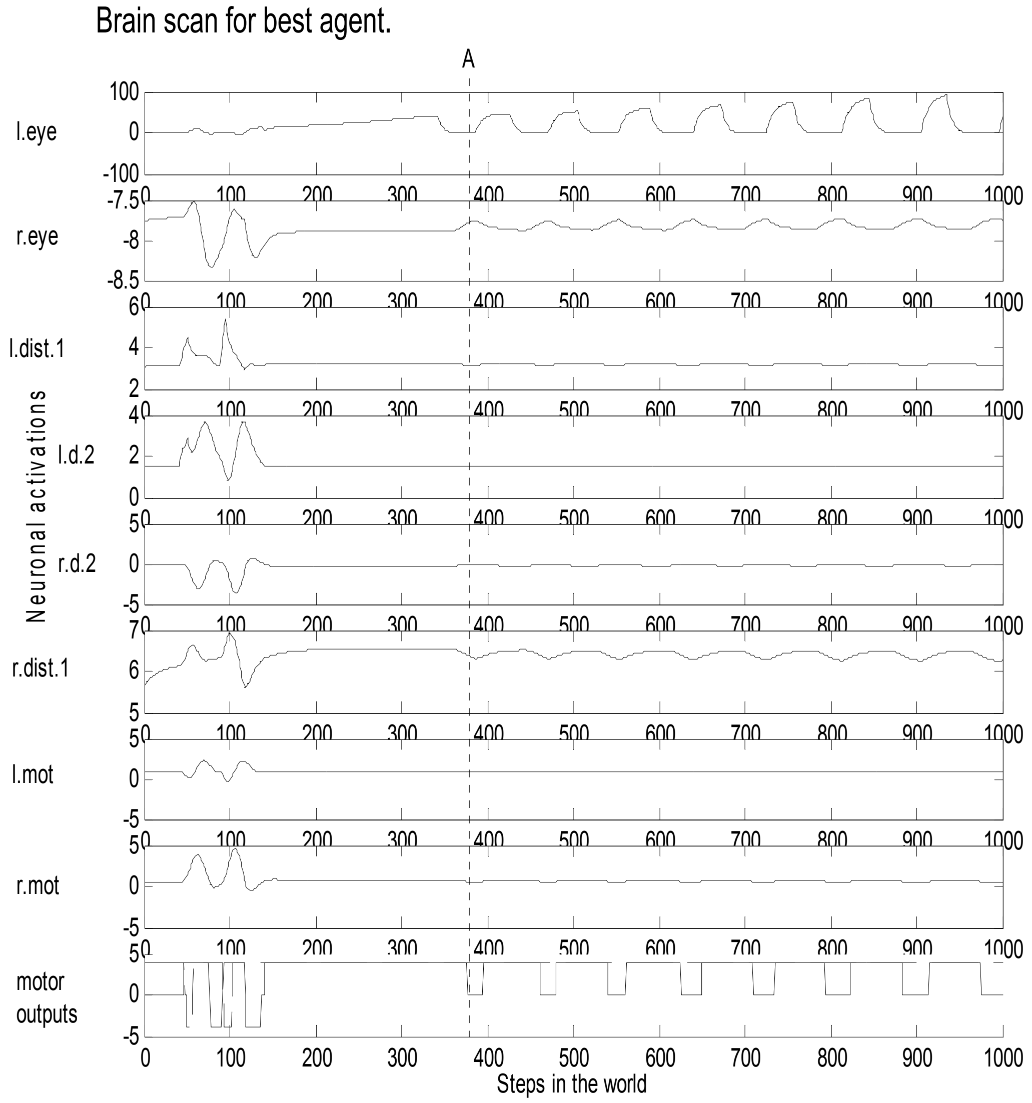


Figure 4.4.1: An agent performing phototropism and avoidance.

Examination of neuronal activations (figure 4.4.2 overleaf) in a typical agent explains these behaviours. Position A on the graphs and folowing diagrams (figure 4.4.3) serves to locate the traces in the diagram.

Trace one, 'leye', shows the rythmic peaks as the source is kept at the periphery of the left eye. Trace two shows the corresponding lesser stimulation from the right eye. Lesioning the eye neurons shows that the right eye is chiefly responsible for effective taxis with the left eye providing a modulatory effect. Increasing 'reye' membrane potential fires the right motor causing straight line travel for a period untill this potential slowly decays. Once decayed, slow turning begins until the light is re-sighted. Decay in 'leye' compensates for the discontinuity in source stimulation, lesioning causes the agent's path to wander. If 'reye' is lesioned then 'rmot' is insufficiently stimulated and looping occurs.

Traces three and four illustrate the perturbation of the left distance sensor pair which causes the avoidance behaviour and wall following. Increasing potential in the left distance sensor inhibits the right motor neuron causing the agent to turn away from the obstruction. Lesioning shows that allthough the inner sensory neuron 'ldist2' assists in good avoidance the outer sensory neuron 'ldist1' is critical. Lesioning 'ldist1' causes the agent to loop with insufficient stimulation of 'rmot'. Lesioning 'ldist2' causes the agent to crash into the fence end-post on about 1 in 10 trials. Lesioning shows that 'rdist1' and 'rdist2' have a slight modulatory effect but that good overall behaviour is maintained without them.



The response in the motor neuron 'rmot' is far more sensitive to 'ldist1' inhibition than to 'reye' excitation. This causes clean subsumption of the tropism by the more important avoidance task. Less optimal agents often suffer from tropic interference in their avoidance manouveres and get stuck.

So, 'lmot' has a high latency perturbed only by strong excitationn of the distance sensors, the left motor runs full forwards. Turning, the key to success in both behaviours, is effected through 'rmot'. This motor neuron is kept excited at the threshold of forwards motion making for sensitive control through 'reye' excitation and 'ldist1' inhibition. Functional modularity at the behavioural level is not repeated at the neuronal level, loss of either 'reye' or 'ldist1' causes failure, both neurons contribute to the near-threshold latency.

A fascinating role is played by 'leye', this neuron recieves high stimulation during phototaxis but has little direct effect on motor control. Instead the decaying potential serves as a memory that a source was recently sighted which in turn compensates for the discontinuous source. Remembering the light allows for a smoother aproach trajectory.

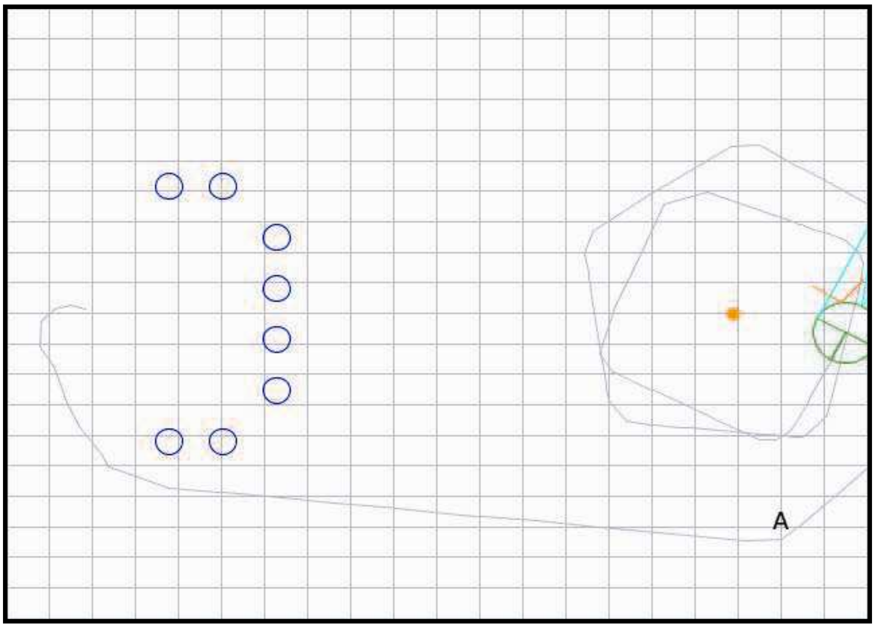
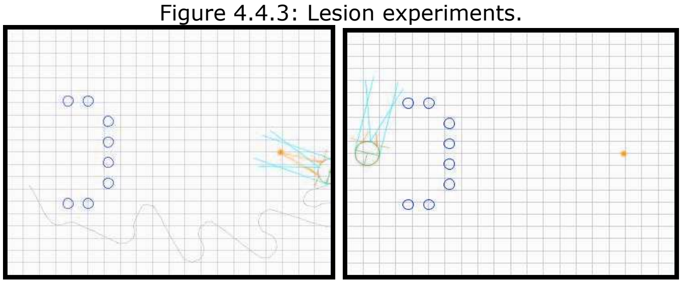
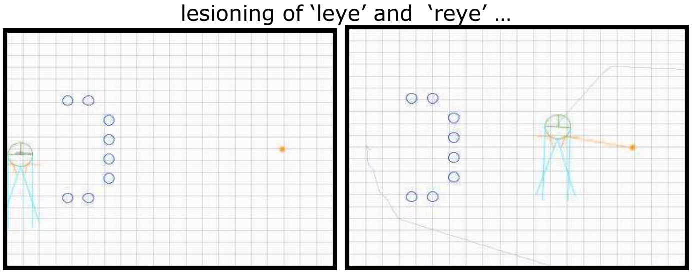


Figure 4.4.2: one run of ‘best’ detour behaviour to match brain scan opposite

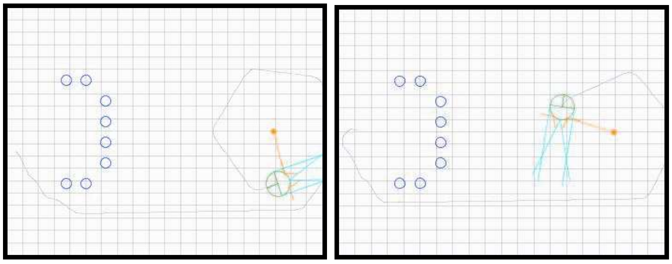
Figure 4.4.3: Lesion experiments.



lesioning of ‘leye’ and ‘reye’ …



…’ldist1’ and ‘ldist2’…



…’rdist1’ and ‘rdist2’.

Further examples of paths and their neuronal traces can be found in appendix II.

Arbib’s frogs allegedly integrate their perceptual information into a single coherent motor schema. In Arbib’s experiments animals were shown to begin a straight-line trajectory to avoid the fence (figure 4.4.4, left image overleaf).

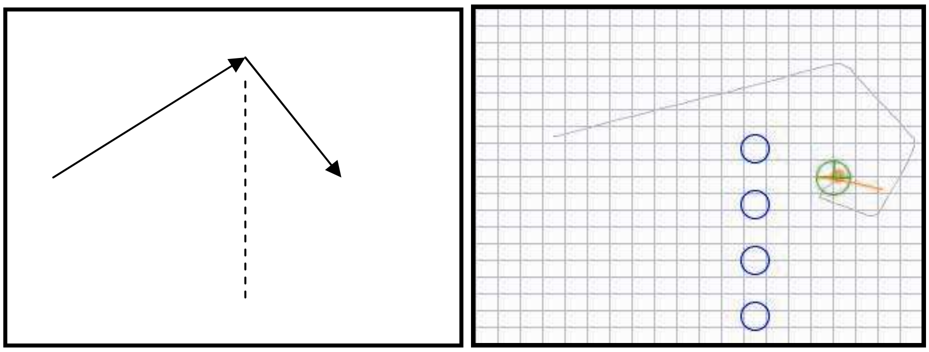


Figure 4.4.4: The frog's detour path, all too easy for spiralling photo-philes.

Unfortunately this form of indirect path is endemic in spiralling phototropes. The fenced worlds used were designed to prevent this sort of task evasion and demand object avoidance skills. Therefore, no simple comparison is possible. If the light was removed from the simulation while the agents were engaged in detour they did not go looking where they thought it might be. In personal communication[1] Arbib has confirmed that the same is true in the frog; the frog is 're-stimulated' by the fly once it reaches the edge of the fence. In all fairness, noticing the absence of the source and thus abandoning the path *would* be the 'intelligent' thing to do. So, is this detour behaviour?

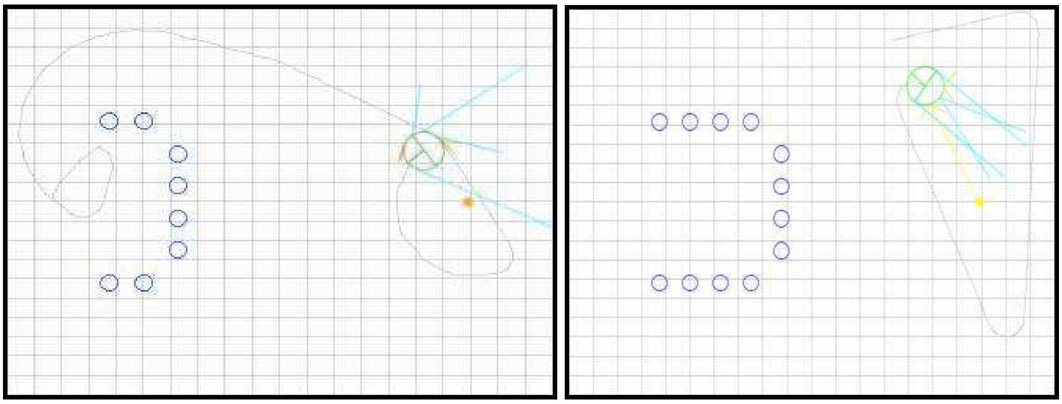


Figure 4.4.5. more complex detour environments used in the experiment.

It would seem that instead of integrated path planning like a frog these agents perform phototaxis subsumed by avoidance behaviour. But, because this behaviour is modulated by an internal 'memory' of the sighted source we would suggest that this is a form of detour behaviour. Just as frogs detour with less grace than rats (Arbib, in personal communication) so our agents detour with less grace than frogs, but detour they do.

---

[1] In conversation about avoidance in frogs, rats and evolved dynamic recurrent neural networks at *Biologically Inspired Robotics - the Legacy of W. Grey Walter,* a workshop at HP Labs in Bristol, 16th-18th August 2002.

5 Conclusions:

5.1 The Hypothesis.

The driving hypothesis of this work was:

> *that* additional domains of plasticity increase evolvability.

A domain of plasticity was understood as a related set of state space variables which could be ceded to evolutionary selection and optimisation; for example sensor morphology or network structure. Evolvability was understood in terms of the speed and ease of evolution on the one hand and the richness and variety of possible strategies or attractors in behaviour space on the other.

The experimental evidence supports that the number of possible attractors in behaviour space increases with increased plasticity. Yet within the compass of these experiments this ultimately frustrated evolutionary progress; evolution was both slower and more difficult in most cases. Strictly, the hypothesis as written should be rejected.

5.2 Criticism.

We regret not having the space to fully expand and clarify the philosophical position underlying this work. It is hoped that embodiment be understood as something more than merely being physical and that some sense of embodiment as a ground for plasticity be allowed. The former point is found widely within the literature (see Ziemke 2001) but the latter is closer to the author's unfinished thoughts (see Cowan 2001).

Artificial evolution is a difficult process to master. There are many parameters to set, many of which are sensitively dependent on both one another and the fitness landscape involved. Add to this three classes of neural network and sensor morphology and one can find oneself in a parametric nightmare. In all more than three hundred full evolutionary runs were completed, many of which debugged and parameterised the algorithms. Due to the noisy nature of the experiments ten times this number of runs might not have yielded enough data. It would have been prudent to spend more time on each of the fifteen experiments completed, to be more methodical about controlling fewer variables.

Beyond simple parameterisation the results of the experiments were unclear and mixed in their interpretation. We feel the analysis best fits the available data and makes a reasonable statement of the situation. Yet some of these difficulties would be ameliorated in larger scale evolutionary runs, especially employing those algorithms which encourage speciation and look to long term species adaptation (e.g. Harvey 2001). Perhaps this exploration deserves more time than was available here.

In the light of this, further work involving yet more domains of plasticity seems unadvisable. One of the tenets of a synthetic approach to science would be to add further complexity only when the existing system is fully understood (Beer 2002). Instead, the existing ground should be covered in better detail.

Behaviours were characterised as attractors within the agent-environment state space. The application of dynamical systems theory techniques to a fuller examination of the state space would yield clearer support for conclusions about the frequency and types of these attractors. Although some attempt was made to classify observed behavioural strategies, this was subjective. As the complexity of the network behaviour increased, the difficulty of borderline cases made such observational classification unworkable, the granularity of such observations being crude. Field notes gave a good insight into what was going on and supported the intuitive analysis of evolutionary statistics, nonetheless, it would be good to get better mathematical support from an analysis of the state spaces.

Detour behaviour could be further refined, although this would be taking the project away from evolutionary exploration into behavioural engineering. Path integration could be encouraged by requiring the agents to minimise energy usage. Exploration of the significance of 'leye' as an internal representation could be used to fuel the ongoing philosophical debate; in this project we have held our tongues!

5.3 Last Words.

Despite rejecting the working hypothesis, we believe these experiments have been successful. The exploration of the effect of additional plasticity on evolvability has demonstrated the complexity of the issue. Many more possibilities are enabled within behaviour space but search becomes more difficult. However, the highest scoring agents were found in fully evolved populations suggesting that increased plasticity *can* improve things. Cooperative co-evolution of controller and sensors led to observable structural congruence in the case of handed behaviours and illustrated a selective pressure for simpler and more robust systems through sensor rationalisation. Both these features were seen to support fitter agents. The working hypothesis is rejected, but only just so.

Primitive detour behaviour was evolved in numerous populations of fully connected dynamic-recurrent networks with both fixed and evolved sensors. Spatial network encodings supported such agents to a lesser extent. The best detour agents subsumed phototropism under wall following and performed well under all circumstances. They were supported in their evolution by extended plasticity, grounded in sensor evolution.

Appendix I:

Bibliography.

Arbib, M., (1987) *Depth and Detours: An Essay on Visually Guided Behaviour.*
In Arbib and Hanson (eds.) *Vision, Brain and Cooperative Computation.* Cambridge, Mass: Bradford Books/M. I. T. Press, (pp129-163).

Arbib, M., (1982) *Rana Computatrix, An Evolving Model of Visuo-motor Coordination in Frog and Toad.*
In Hayes, Michie and Pao (eds.) *Machine Intelligence* 10, Chichester: Ellis Horwood, (pp501-517).

Balakrishnan, K., and Honavar, V., (1996) *On Sensor Evolution in Robotics*.
In Koza, Goldberg, Fogel, Riolo (eds.) *Proceedings of Genetic Programming 96*, Cambridge, Mass: M. I. T. Press, (pp455-460).

Barnett, L., (2001) *Netcrawling – Optimal Evolutionary Search with Neutral Networks,*
In *Proceedings of the 2001 Congress of Evolutionary Computation.* Piscataway, NJ: IEEE Press, (pp30-37).

Beer, R., D. (2002) *Frictionless Brains.*
To appear in *Proceedings of Biologically Inspired Robots - the Legacy of W. Grey-Walter,* (2002). Hewlett Packard Technical Report. In Press.

Beer, R., D. (1996) *Towards the Evolution of Dynamical Neural Networks for Minimal Cognitive Behaviour.*
In Maes, Mataric, Meyer, Pollack and Wilson (eds.) *From Animals to Animats 4: Proceedings of the Fourth International Conference on Simulation of Adaptive Behaviour.* Cambridge, Mass: M. I. T. Press, (pp. 421-429).

Beer, R., D., Gallagher, J. C. (1992) *Evolving Dynamic Neural Networks for Adaptive Behaviour.*
In *Adaptive Behaviour* 1, (pp91-122).

Bongard, J. C. & C. Paul (2001) *Making Evolution an Offer It Can't Refuse: Morphology and the Extradimensional Bypass.*
In Keleman & Sosik (eds.), *Proceedings of the Sixth European Conference on Artificial Life*. Berlin: Springer-Verlag, (pp. 401-412).

Bongard, J. C. & R. Pfeifer (2001) *Repeated Structure and Dissociation of Genotypic and Phenotypic Complexity in Artificial Ontogeny*.
In Spector, Goodman, Wu, Langdon, Voigt, Gen, Sen, Dorigo, Pezeshk, Garzon, Burke (eds.) *Proceedings of the Genetic and Evolutionary Computations Conference 2001,* San Francisco CA: Morgan Kaufman, (pp829-836).

Brooks, R. A. (1990) *Intelligence without Reason.*
In *Artificial Intelligence* 47. (1991), (pp139–159).

Clark, A. (1997) *Being There, Putting Brain, Body and World Together Again.*
Cambridge, Mass: M. I. T. Press.

Cliff, D. and Miller, G., T., (1995). *Tracking the Red Queen: Measurements of Adaptive Progress in Co-evolutionary Simulations.*
In Moran, Moreno, Merelo, and Chacon (eds.) *Advances in Artificial Life: Proc. Third European Conference Artificial Life*, Berlin: Springer-Verlag, (pp.200-218).

Collins, R. and Jefferson, D., (1991) *Selection in Massively Parallel Genetic Algorithm.*
In Belew, Booker, (eds.) *Proceedings of the Fourth International Conference on Genetic Algorithms.* (pp249-256).

Conrad, M., (1990) *The Geometry of Evolution.*
In *Biosystems* 24 (pp 61-81).

Cowan, D. J., (2001) *Embodiment and Cognition.*
Unpublished Masters Degree term paper. Available on request: dan@danslan.freeserve.co.uk

Dellaert, F. and Beer, R.D. (1996) *A Developmental Model for the Evolution of Complete Autonomous Agents*.
In Maes, Mataric, Meyer, Pollack and Wilson (eds.), *From Animals to Animats 4: Proceedings of the Fourth International Conference on Simulation of Adaptive Behaviour.* Cambridge, Mass: M. I. T. Press, (pp393-401).

Di Paolo, E. A., (2000) *Homeostatic Adaptation to Inversion of the Visual Field and Other Sensory Motor Disruptions.*
In Meyer, Berthoz, Floreano, Roitblat and Wilson (eds.) *From Animals to Animats: Proceedings of the Sixth International Conference on Simulation of Adaptive Behaviour*. Cambridge, Mass: M. I. T. Press.

Eggenberger, P., (1996) *Cell Interactions for Development in Evolutionary Robotics.*
In Maes, Mataric, Meyer, Pollack and Wilson (eds*.), From Animals to Animats 4: Proceedings of the Fourth International Conference on Simulation of Adaptive Behaviour,* Cambridge, Mass: M. I. T. Press, (pp. 440-448).

Floreano, D., and Mondada, F. (1996a) *Evolution of Homing Behaviour in a Real Mobile Robot*
IEEE Transactions on Systems, Man and Cybernetics-Part B, 26. Piscataway, NJ: IEEE Press, (pp396-407).

Floreano, D., and Mondada, F. (1996b) *Evolution of Plastic Neurocontrollers.*
In Maes, Mataric, Meyer, Pollack and Wilson (eds.) *From Animals to Animats 4: Proceedings of the Fourth International Conference on Simulation of Adaptive Behaviour.* Cambridge, Mass: M. I. T. Press, (pp. 403-410).

Floreano, D., and Mondada, F. (1994) *Automatic creation of an autonomous agent: Genetic evolution of a neural-network driven robot.*
In Cliff, Husbands, Meyer, and Wilson (eds.) *From Animals to Animats: Proceedings of the Third International Conference on Simulation of Adaptive Behaviour*. Cambridge, Mass: M. I. T. Press, (p421-430).

Grey Walter, W., (1950) *An Imitation of Life*
In *Scientific American* 182 (5) (pp42-45).

Gruau, F., (1994) *Automatic definition of modular neural networks*,
In *Adaptive Behaviour* 3 (2), (pp151-183).

Harvey, I., (2001) *Artificial Evolution: A Continuing SAGA.*
In Gomi (ed.), *Evolutionary Robotics: From Intelligent Robots to Artificial Life*, *Proceedings of Eighth International Symposium on Evolutionary Robotics*. Berlin: Springer-Verlag.

Harvey, I., Husbands, P., and Cliff, D., (1994) *Seeing the light: artificial evolution, real vision.*
In Cliff, Husbands, Meyer and Wilson (eds.), *From Animals to Animats 3: Proceedings of the Third International Conference on the Simulation of Adaptive Behaviour*. Boston MA: M. I. T. Press/Bradford Books, (pp392-401).

Holland, J. (1975) *Adaptation in Natural and Artificial Systems.*
University of Michigan Press.

Huber, S. A., Mallot, H. A. and Bulthoff, H. H., (1996) *Evolution of the Sensorymotor Control in an Autonomous Agent.*
In Maes, Mataric, Meyer, Pollack and Wilson (eds.), *From Animals to Animats 4: Proceedings of the Fourth International Conference on Simulation of Adaptive Behaviour*. Cambridge, Mass: M. I. T. Press, (pp449-457).

Husbands, P., Harvey, I., Cliff, D., Thompson, A., and Jakobi, N., (1996) *Evolutionary Robotics: The Sussex Approach.*
In *Robotics and Autonomous Systems,* 20, (pp205-224).

Husbands, P., (1998) *Better Living Through Chemistry.*
In *Connection Science* 10 (3&4). (pp185-210).

Jakobi, N., (1998) *Running Across the Reality Gap: Octopod Locomotion Evolved in a Minimal Simulation*.
In Husbands and Meyer (eds.) *EvoRobot-98: Proceedings of First European Workshop on Evolutionary Robotics.* Berlin: Springer-Verlag, (pp39-58).

Lipson, H. and Pollack, J. P., (2000) *Automatic Design and Manufacture of Robotic Life-forms.*
Nature 406, (pp974-978).

Mayley, G., (1996) *Evolutionary Cost of Learning*.
In Maes, Mataric, Meyer, Pollack and Wilson (eds.), *From Animals to Animats 4: Proceedings of the Fourth International Conference on Simulation of Adaptive Behaviour.* Cambridge, Mass: M. I. T. Press, (pp. 458-467).

Mark, A., Polani, D., and Uthmann, T., (1998) *A Framework for sensor Evolution in a Population of Braitenberg Vehicle-like agents.*
In Adami, Belew, Kitano and Taylor (eds.) *Procedings of Artificial Life VI.* Cambridge, MA: MIT Press, (pp428-432).

Maturana, H. R. and Varela, F. J. (1987) *The Tree of Knowledge- The Biological Roots of Human Understanding*.
Boston: Shambala Press.

Merleau-Ponty, M. (1962) *Phenomenology of Perception*.
London and New York: Routledge Press.

Miglino, O., Denaro, D. and Tascini, G., (1998) *Detour Behaviour in Evolving Robots: Are Internal Representations Necessary?*
In Husbands and Meyer (eds.) *EvoRobot98: Proceedings of 1st European Workshop on Evolutionary Robotics.* Berlin: Springer-Verlag.

Noble, J. and Watson, R., A. (2001) *Pareto Co-evolution: Using Performance against Co-evolved opponents in a Game as Dimensions for Pareto Selection.*
In Spector, Goodman, Wu, Langdon, Voigt, Gen, Sen, Dorigo, Pezeshk, Garzon, Burke (eds.) *Proceedings of the Genetic and Evolutionary Computations Conference.* San Francisco CA: Morgan Kaufman, (pp493-500).

Nolfi, S., (1997) *Evolving non-Trivial Behaviours on Real Robots: A Garbage Collecting Robot.*
In *Robotics and Autonomous Systems* 22 (3-4), (pp187-198).

Pfeifer, R., (1999) *Dynamics, morphology, and materials in the emergence of cognition.*
Proceedings. KI-99, Lecture Notes in Computer Science, Berlin: Springer-Verlag, (pp27-44).

Pfeifer, R., and Scheier, C. (1998) *Embodied Cognitive Science: A novel approach to the study of intelligence in natural and artificial systems.*
In T. Gomi (ed.). *Evolutionary Robotics, Vol. II: From intelligent robots to artificial life*. Ontario, Canada: Applied AI Books, (pp1-95).

Quick, T. Dautenhahn, K., and Nehaniv, C. L., (1999) *On Bots and Bacteria, Ontology Independent Embodiment*.
In Floreano, Nicoud and Mondada (eds.) *Proceedings of the Fifth European Conference on Artificial Life.*
Berlin: Springer-Verlag.

Rosenblatt, F. (1962) *Principals of Neurodynamics.*
New York: Spartan Books.

Salomon, R., (1996) *Increasing Adaptivity through Evolution Strategies.*
In Maes, Mataric, Meyer, Pollack and Wilson (Eds.), *From Animals to Animats 4: Proceedings of the Fourth International Conference on Simulation of Adaptive Behaviour.* Cambridge, Mass: M. I. T. Press, (pp. 411-420).

Sharkey, N. and Ziemke, T., (1999) *Life, Mind and Robots: The Ins and Outs of Embodied Cognition.*
In Wermter and Sun (eds.) *Hybrid Neural Systems*. Berlin: Springer-Verlag, (pp313-332).

Simms, K., (1994) *Evolving 3D Morphology for Behaviour by Competition.*
In Brooks and Maes (eds.) *Proceedings of the Fourth International Conference on Artificial Life* .Cambridge, Mass: M. I. T. Press, (pp28-30).

Teller, A. (1994) *The Evolution of Mental Models*.
In Kinnear (ed.) *Advances in Genetic Programming*, Cambridge, Mass: M. I. T. Press, (pp199-219).

Von Uexkull, J. (1928) *Theoretiche Biologie.*
Berlin, Springer-Verlag.

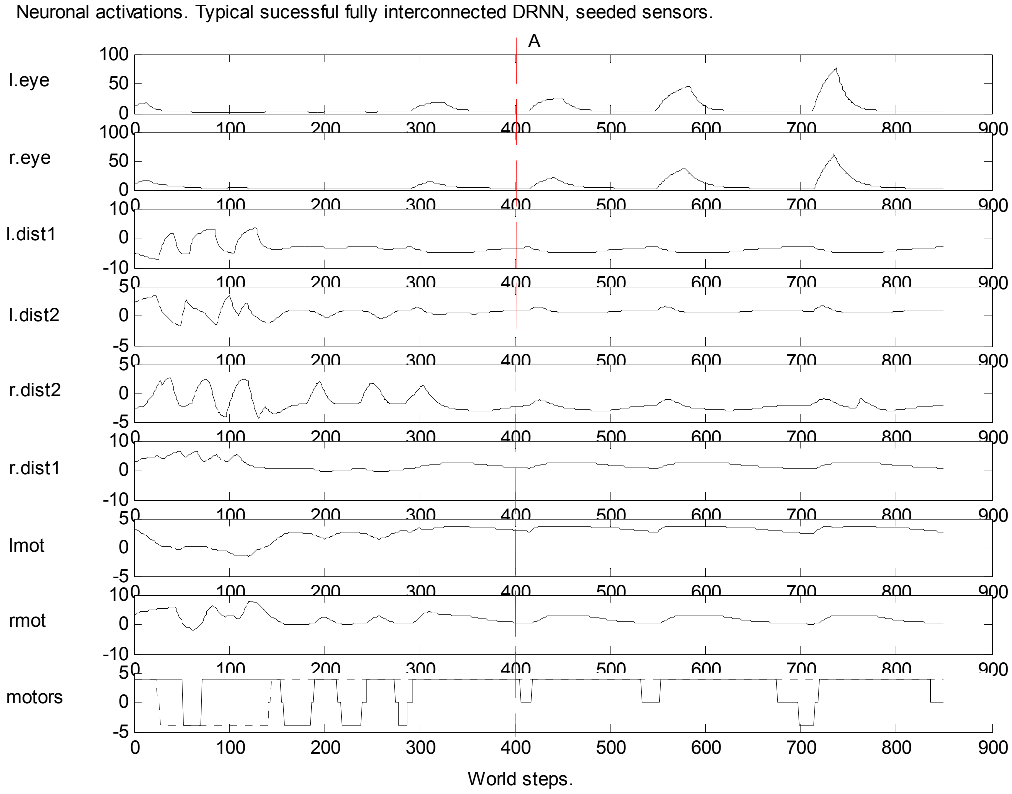
Webb, B. (1995) *Using Robots to Model Animals: A Cricket Test.*
In *Robotics and Autonomous* Systems 16, (pp117-134).

Wittgenstein, L., (1949) *Philosophical Investigations.*
(trans.) Anscombe, E., (1953), Oxford: Blackwell.

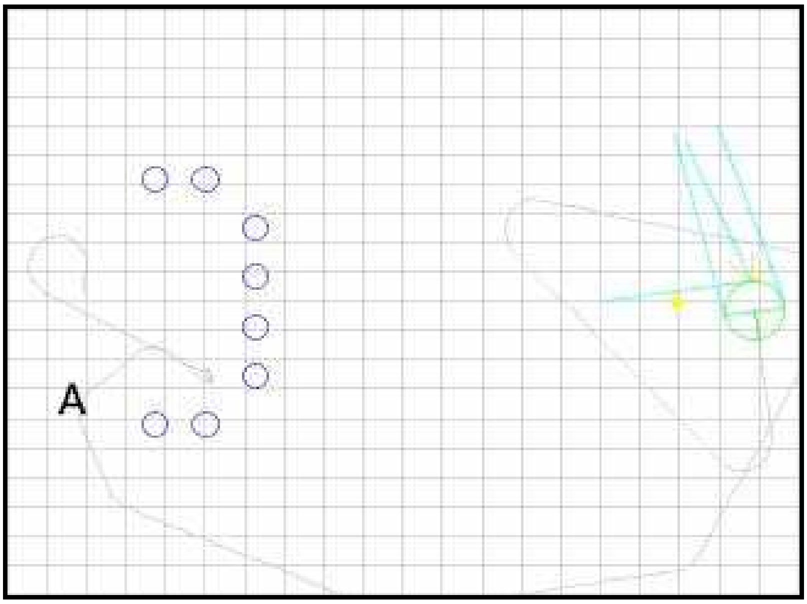
Ziemke (2001) *Are Robots Embodied*?
In *First International Workshop on Epigenetic Robotics: Modelling Cognitive Development in Robotic Systems*. Lund University Cognitive Studies, (vol. 85).

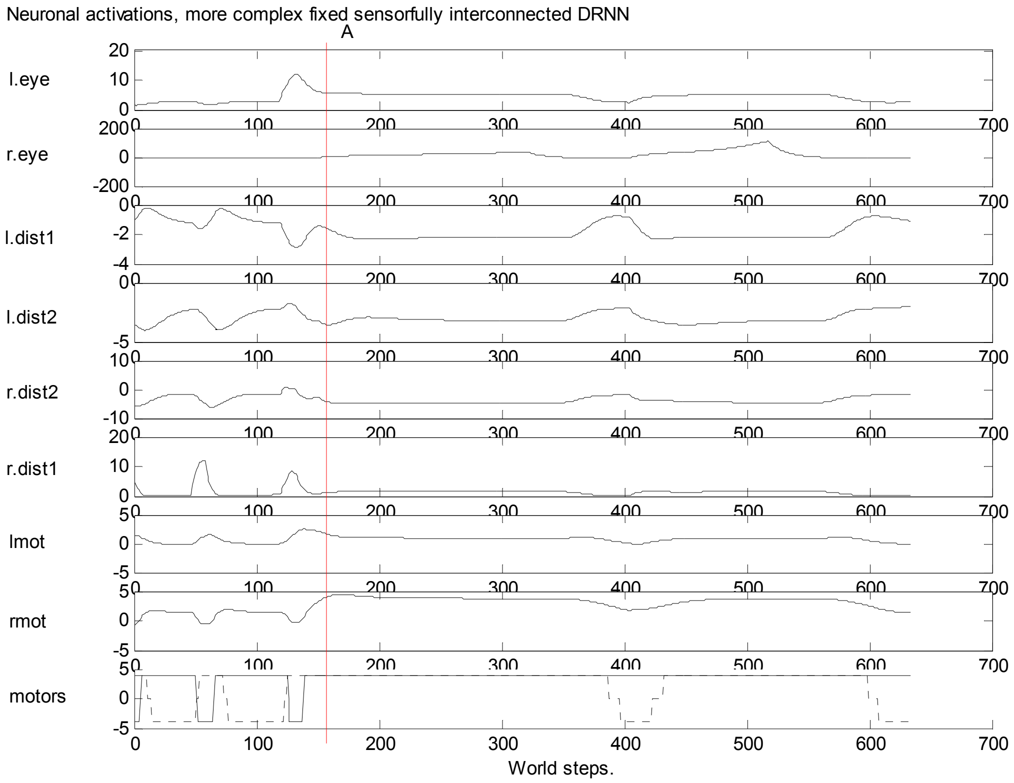
Appendix II:

Further Detour Analysis.



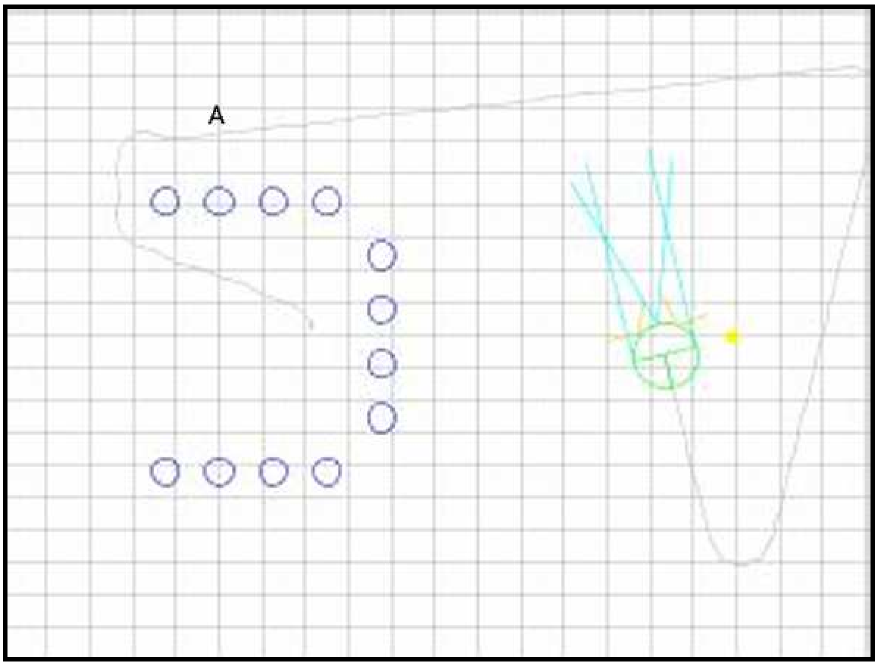
Both eyes have evolved together. L.mot has a latency which drives the motor forwards but which is strongly inhibited by l.dist1 and l.dist2. R.mot is excited by l.eye and r.eye and decays very slowly. So, proximity of an obstacle drives both motors into reverse with same-side inhibition the strength of which subsumes phototropism. Phototropism is achieved through spiralling. Rmot's slow decay causes overshooting with a corresponding sharp turn when rmot activation falls below the motor threshold.





Neuronal activations, more complex fixed sensorfully interconnected DRNN

Essentially the same behaviour grounded in a very much more complex brain. Neuronal connectivity ensures greater damping of any one neuron's contribution to overall behaviour. R.dist1 has a significant inhibitory effect on rmot. Leye strongly excites both lmot and rmot which both decay very slowly. Turning is initiated when lmot decays below the motor threshold, bringing the source back to gently excite r.eye.



Appendix III:

Implementation.

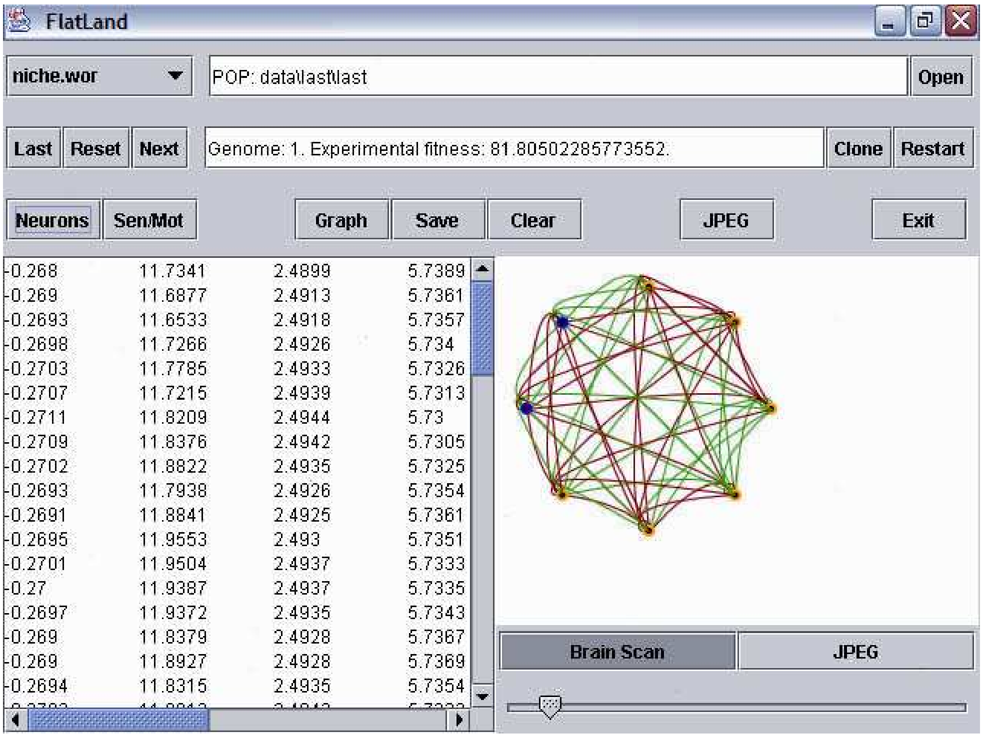
The algorithms necessary for this project come from a package written by this author (Dan Cowan) and Tony Poppleton.

Developed during studies at the University of Sussex to support the rapid deployment of experiments into artificial life, it was the aim of the authors to minimise the coding time involved in new artificial life projects by providing extensive libraries of the most common algorithms. This is still very much a work in progress.

The package comprises of a set of class definitions grouped into sub-packages or libraries including neural networks, 2D geometry and physics, genetic algorithms and various a-life support utilities. Each package contains further sub packages such as dynamic-recurrent and feed forward networks or single distributed grid or co-evolutionary genetic algorithms.

The algorithms were developed using Java 1.4.0 to support maximum platform independence and utilise Java's interface technologies to secure generality of algorithm and extensibility. Anyone wishing to add a new genetic algorithm class to the package must merely fulfil the genetic algorithm interface requirements.

The world, agents, sensors and the programs which drive the algorithms such as the GUI below were written by this author alone and are specific to this project.



niche.wor

POP: data\last\last

Genome: 1. Experimental fitness: 81.80502285773552.

```
-0.268     11.7341    2.4899    5.7389
-0.269     11.6877    2.4913    5.7361
-0.2693    11.6533    2.4918    5.7357
-0.2698    11.7266    2.4926    5.734
-0.2703    11.7785    2.4933    5.7326
-0.2707    11.7215    2.4939    5.7313
-0.2711    11.8209    2.4944    5.73
-0.2709    11.8376    2.4942    5.7305
-0.2702    11.8822    2.4935    5.7325
-0.2693    11.7938    2.4926    5.7354
-0.2691    11.8841    2.4925    5.7361
-0.2695    11.9553    2.493     5.7351
-0.2701    11.9504    2.4937    5.7333
-0.27      11.9387    2.4937    5.7335
-0.2697    11.9372    2.4935    5.7343
-0.269     11.8379    2.4928    5.7367
-0.269     11.8927    2.4928    5.7369
-0.2694    11.8315    2.4935    5.7354
```

The GUI developed for viewing agent behaviour and generating telemetry in the form of data files, graphs and jpegs of the agent's world, body and brain states.

Appendix IV contains project specific code while appendix V contains the package code and a brief explanation of structure and use.

Appendix IV:

Project Specific Source code.

Almost 150 files of source code follow. This appendix contains the project specific code including the package `simulator2D` and the two main driver programs: `Evolve` and `Viewer`.

`Evolve` is a command line based application which evolves a specified population to a specified fitness test. The command line options are listed below:

```
USAGE: java Evolve [args]
    -g   n    Sets the number of generations to n (currently 500)
    -t   n    Sets the number of trials to n (currently 5)
    -pop n    Sets the population to n (currently 100)
    -ff  n    Sets fitness test(1.Avoid 2. Cylinder 3.Barrier) (ff=3)
    -i   n    Sets the maximum number of steps per trial to n (i=100)
    -n   n    Sets the level of noise to n (currently 0.25)
    -w   n    Sets the world to n (currently worlds\barrierTest.wor)
    -s   n    Loads a sensor file, n (currently null)
    -p   n    Loads a population, n (currently null)
    -nt  n    Network (1.ff  2.drnn 3.spatial 4.sectored) (nt=4)
    -nn  n    Sets number of network nodes to n (currently 16)
    -nc  n    Sets network connectivity to n (currently 0.5)
    -nm  n    Sets the net mutation rate to n (currently 1.0)
    -sm  n    Sets the sensor mutation rate to n (currently 0.05)
    -amb n    Sets ambient sensors to n (currently 0)
    -dir n    Sets directional light sensors to n (currently 2)
    -ray n    Sets single ray light sensors to n (currently 0)
    -son n    Sets sonar sensors to n (currently 2)
    -ec -!ec  Flips elite child flag (currently true)
    -er -!er  Flips elite replace flag (currently true)
    -sa -!sa  Flips the save all populations flag (currently false)
    -st -!st  Flips the save date stamped flag (currently true)
    -?        Prints usage help.
```

`Viewer` is the Swing based GUI application in appendix III, for viewing evolved populations and generating telemetry for analysis. Called from the command line it defaults to view the last evolved population but can be pointed at any data directory containing the files saved by the `Evolve` program during a run. Populations can be trialled in different worlds while data files of sensor/motor/neuronal values are written to disk, jpegs of the agent's path, brain and sensor profile can be also be recorded. Data is written in a Matlab or Access friendly style.

Appendix V:

Alife Package Source code.

The package alife breaks down into a number of related sub-packages, files are printed in sub-package order, starting with those of: `alife.evolution.*`

`alife`				
	`evolution`			genetic interfaces.
		`encoding`		genotype to phenotype base encodings.
		`ga`		genetic algorithms.
		`ra`		reproduction algorithm container classes.
			`mutation`	mutation operators.
			`combination`	combination operators.
	`geometry2D`			radian based geometry classes.
		`gui`		graphical extensions of geometry classes.
	`neurlnets`			neural network interfaces.
		`drnn`		dynamic-recurrent neural network.
		`ffnn`		perceptron neural network.
	`utils`			utilities for file management, time management, additional maths, graphing, and statistics.

Each object set is governed through interfaces which determine the ways in which an object may be used. For example, all neural networks must have `fire()` and `update()` methods, calling `fire()` on any neural network will have the same functional result.

The reproduction algorithm determines mutation and crossover operators for each chromosome on a genome. The fitness function object contains the encodings for translating the genomes within a population into their phenotypic form and the means of their evaluation.

A genetic algorithm object is initialised with a population size, number of trials, generations and other necessary flags. The method:

```
ga.iterate(ff, ra)
```

runs the algorithm for one generation using the fitness function and reproduction algorithm passed in the call. The code:

```
while (ga.isRunning)
      ga.iterate(ff, ra);
```

drives the algorithm for the number of generations specified in the constructor.