Evolvability, Neutrality and Search Difficulty

Tom Smith[*1,2], **Phil Husbands**[1,3] **and Michael O'Shea**[1,2]
[1]Centre for Computational Neuroscience and Robotics (CCNR)
[2]School of Biological Sciences
[3]School of Cognitive and Computing Sciences
University of Sussex, Brighton, UK
[*]toms@cogs.susx.ac.uk

Abstract

Analysis of problem search spaces is crucial if we are to use artificial evolutionary techniques to produce good solutions to difficult problems. In this paper, we apply the concept of evolvability in order to highlight the differences between two non-trivial search spaces, for which significant differences in the time required to evolve good solutions has previously been shown. We define a set of evolvability metrics based on the distribution of solution offspring fitnesses, and show that the metrics do predict the difficulty of finding good solutions in a class of tunably rugged and neutral landscapes. In applying the metrics to the search space defined by a robotics visual shape discrimination task, we find no evidence that evolvability changes during neutral epochs. However, the evolvability measures do show differences between the search spaces defined by two different robot control architectures. In particular we see a decrease in the number of non-deleterious mutations for one architecture, allowing the evolving population to contain a larger number and variety of good genotypes on which the evolutionary process can work.

Keywords: Evolvability, Neutral Evolution, Search Space, Genetic Algorithm, Evolutionary Robotics

1 Introduction

It is often argued that the success of biological evolution is due in large part to the discovery of highly evolvable genetic systems (see e.g. Dawkins, 1989; Partridge and Barton, 2000; Kirschner and Gerhart, 1998; Wagner and Altenberg, 1996). In this paper we investigate the evolvability of two different genetic systems in an artificial evolutionary setting, in order to explain the differences in the evolutionary time required to find good solutions. The conclusions we draw regarding the nature of the two search spaces, in particular the importance of non-deleterious neutral mutations, may also apply to biological evolution.

Evolving good solutions to difficult problems takes time, especially when the evaluation of individual solution fitnesses is time-consuming. Any reduction in the time required, either in the number of solutions needed to be evaluated or in the time taken for each evaluation, is thus of great benefit. One potential avenue for artificial evolution research is in the development and analysis of solution representations on which evolution can work to produce good solutions in

few evaluations. The analysis of the search space underlying such good representations is clearly of prime importance.

In recent work, we have investigated a range of architectures, or genetic systems, to be used as controllers in the problem domain of evolutionary robotics (Nolfi and Floreano, 2000; Husbands and Meyer, 1998), and introduced the "GasNet" (Husbands, 1998; Husbands et al., 1998; Smith and Philippides, 2000). Previous work has demonstrated significant reduction in the number of evaluations required to evolve successful GasNet controllers over a range of robotics tasks, when compared with more standard control architectures. In this paper, we investigate the complex search spaces underlying two control architectures ("GasNet" and "NoGas", described further in section 4), in order to explain the speed of evolution differences observed. We must emphasise the non-trivial nature of the genotype-to-fitness mapping: the initial genotype is translated to an intermediate neural network phenotype, which is then evaluated as a controller for a robot completing some sensorimotor task. It is by no means clear that conclusions drawn from work on more theoretical landscapes will apply in such spaces.

We argue that the concept of evolvability, or the ability of an individual or population to evolve, has advantages over more standard measures of search space properties, such as landscape ruggedness and local modality, when considering landscapes with significant degrees of neutrality (Kimura, 1983). In particular, we define a set of evolvability metrics based on the distribution of offspring, which we show can discriminate between theoretical search spaces of differing neutrality. We show that the search space defined by the robotics task does indeed show significant neutrality, and that the evolutionary process contains long neutral epochs, punctuated by short bursts of increase in fitness. However, we find no evidence that evolvability changes during these neutral epochs, so cannot confirm theoretical arguments that evolvability will increase during evolution, even during neutral epochs.

Applying the evolvability metrics to the two search spaces defined by the GasNet and NoGas control architectures does highlight differences between the two spaces, markedly in the number of deleterious mutations. The GasNet architecture defines a search space in which fewer mutations at higher fitness are deleterious; thus a wider variety of good genotypes will exist in the population at any one time. However, it is unclear whether this effect is the only cause of the faster evolutionary search; the analysis into the two search spaces is an ongoing project.

The paper proceeds as follows. Section 2 introduces the idea of the search space as a fitness landscape, and describes the important concept of landscape neutrality. Section 3 relates these notions to the idea of evolvability, introduces the offspring transmission function, and derives a set of evolvability metrics based on the function. Section 4 outlines the evolutionary robotics approach, introduces the complex genotype-to-phenotype mapping used in this paper, and describes previous results showing evolutionary speed differences between two robotics controller architectures. Section 5 goes on to describe how we apply the evolvability analysis to the two evolvability search spaces, and section 6 details the results of analysing a single evolutionary run in detail. Section 7 applies the evolvability metrics to the two different control architecture search spaces, and the paper concludes with discussion.

2 Search spaces and neutrality

This section introduces two of the main concepts used in the paper. The *fitness landscape* (section 2.1), first introduced by Wright (1932), describes the search space as a multi-dimensional landscape defined by the genotype-to-fitness mapping through which evolution moves. The classical idea of searching this landscape for good genotypes focuses on the difficulty of climbing up to the globally optimal fitness solution, and avoiding locally optimal solutions. Here we argue that in difficult search problems, much of the time spent optimising may be spent in non-adaptive *neutral* evolution (section 2.2). Thus measures aimed at predicting the difficulty of search, must take account of the nature of neutrality in the space. Section 3 goes on to outline how the concept of evolvability, in particular the transmission function, can be used to develop such a measure.

2.1 Fitness landscapes

Wright (1932) introduced the *fitness landscape* as a non-mathematical aid to visualise the action during evolution of selection and variation (in this paper we will use the term evolution to refer to both natural biological evolution, and the artificial evolution class of stochastic search processes which operate through some form of "generate-and-test" algorithm, e.g. genetic algorithms (Holland, 1992), genetic programming (Koza, 1992), evolutionary strategies (Rechenberg, 1973) and evolutionary programming (Fogel et al., 1966)). The description views the space in which evolution takes place as a landscape, with one dimension per genotype loci and an extra dimension, or height, representing the phenotype, or fitness, of that particular genotype[1]. The search space defined by a two-loci representation can thus be viewed as a three-dimensional fitness landscape (figure 1), with each point corresponding to a single genotype and fitness. Applying a mutation operator to a particular genotype A typically produces a cluster of offspring genotypes lying close to A in the landscape, while recombination of two different genotypes A, B typically produces offspring genotypes lying somewhere between A and B in the landscape. Evolution can thus be viewed as the movement of the population, represented by a set of points (genotypes), towards higher (fitter) areas of the landscape.

This view of the search space leads naturally to the identification of the major problems with which evolution will have to cope; ruggedness and modality (Kallel et al., 2000). Highly *epistatic* problems where fitness is dependent on multiple inter-gene interactions will produce a rugged landscape, in which the direction to good solutions is obscured by local noise. Similarly, a high degree of *modality*, i.e. large numbers of local optima, will be seen as large numbers of hill-tops in the landscape with no neighbours of higher fitness. The majority of measures derived to predict the difficulty of searching in a given space are based around these problems of ruggedness and modality (see e.g. Weinberger, 1990; Hordijk, 1996; Jones and Forrest, 1995; Naudts and Kallel, 2000).

A more exact picture, especially when dealing with solutions represented by discrete-valued genotypes, is the connected graph (Stadler, 1996). Solution vertices, or nodes, are connected

[1] Wright defined two forms of fitness landscapes. The first version, used in this work, defines each point on the landscape as representing a single genotype with height corresponding to genotype fitness. The second version has each landscape point representing an entire population, with the values along each dimension representing the allele frequency over the population, and the height corresponding to the mean population fitness. The two approaches may show markedly different properties (Coyne et al., 1997).

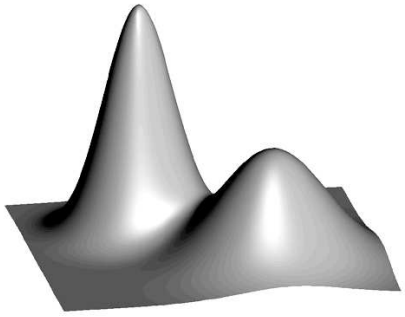


Figure 1: A two-dimensional model fitness landscape, with one globally-optimal and one locally-optimal peak. From a starting point, typically chosen at random, the search process tries to find good solutions. The process typically creates a new set of solutions through the application of genetic operators to the current solution(s), evaluating whether the new set is better than the current solutions. Evolving populations will tend to get stuck at the locally-optimal peak due to its large basin of attraction, and from there will only find the global optimum with difficulty.

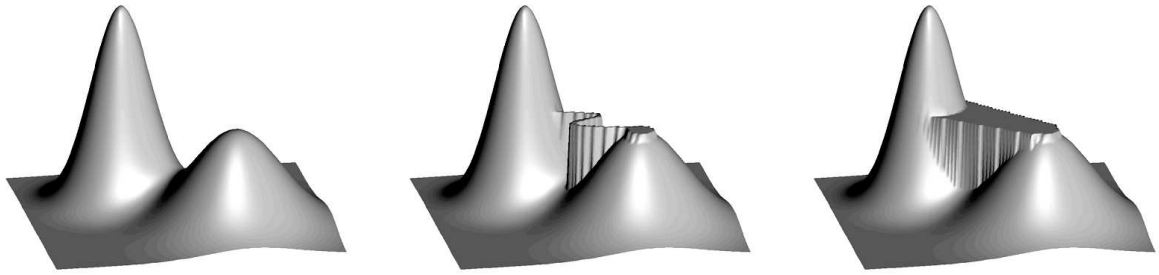
directly through the action of the genetic operators. The graph may show the space in a very different way to the fitness landscape: mutation operators acting on more than one loci, and other operators such as recombination, may not 'see' fitness landscape hill-tops as local optima at all. However, local optima can clearly exist in the graph, occurring as graph nodes from which all connected nodes are of lower fitness. This definition may produce local optima with respect to genetic operators other than simply mutation, for example some solutions may be local optima with respect to recombination operators.

The graph definition of the search space highlights the dangers in the simple visualisable picture afforded to us by the fitness landscape description: our intuitive view may not apply in higher dimensional spaces. For example, local optima may not exist in a large class of high-dimensional spaces, a view expressed by Fisher the same year that Wright developed the fitness landscape concept (Provine, 1986, p. 274), although it should be stressed that many problems clearly do show local optimality, e.g. the travelling salesman problem (Lawler et al., 1985). The next section introduces the idea of search space *neutrality*, one possible way in which some high-dimension spaces may differ radically from our intuitive viewpoint.

2.2 Fitness landscape neutrality

In the neutral theory, it is argued that evolving populations may spend relatively large periods of time undergoing non-adaptive neutral mutation (Kimura, 1983), staying at a constant height in the fitness landscape. The evolutionary timescale may be dominated by long periods of neutral epochs (van Nimwegen et al., 1999), interspersed with short periods of rapid fitness increase, i.e. *punctuated equilibrium* (Eldredge and Gould, 1972; Gould and Eldredge, 1977; Elena et al., 1996). During these neutral epochs, the population will move in the space through random drift (note this is a separate process to Wright's idea of genetic drift due to finite population size). Despite the undirected nature of the population movement, neutrality can be of use in escaping from local optima: figure 2 shows three model landscapes illustrating the possible advantages of neutrality.

Neutral mutation in a fitness landscape will occur as random drift between solutions of equal fitness connected by mutation; such neutral evolution cannot be distinguished from a population stuck in a local optimum by looking at fitness. Instead, the underlying dynamics of the population must be investigated. Two key features have been predicted that distinguish the behaviour



(a) Unconnected peaks (b) Single neutral network (c) Broad neutral plateau

Figure 2: Three two-dimensional model fitness landscapes showing the possible advantage of neutrality in a simple landscape with one globally-optimal and one locally-optimal peak. **(a)** shows the two peaks as unconnected; populations evolving to the locally optimal peak will have difficulty moving to the global optimum. **(b)** shows the two peaks connected by a single neutral network; a population on the sub-optimal peak will eventually find the neutral pathway. **(c)** shows the two peaks connected by a broad plateau; the population will move easily from the sub-optimal peak to the global optimum.

of the population during these periods of neutral drift from the behaviour of a population stuck in a local optimum. The first key feature is that of *movement*, i.e. is the population moving significantly in genotype space. The second key feature is that of *constant innovation*, i.e. the number of previously unencountered phenotypes seen over time is constant. An immediate corollary of this property is that any phenotype should be accessible from a large enough neutral network; neutral drift can eventually find a higher fitness genotype and jump up to a higher neutral network.

These key features have been analysed in many theoretical landscapes. Barnett (1998) introduces the NKp landscape, a tunably neutral variant on Kauffman's NK systems (Kauffman, 1993), and through calculation of population diffusion coefficients shows highly neutral landscapes produce much more population movement during periods of no fitness change when compared with such periods in landscapes of zero neutrality. An important result from the NKp landscape work is that changing the amount of neutrality in the landscape has no effect on the ruggedness correlation function; correlation lengths do not predict the change in evolutionary dynamics seen for landscapes of different neutrality (Barnett, 1998). Newman and Engelhardt (1998) investigate a similar tunably neutral variant of the NK system, finding that increased neutrality allowed high fitness solutions to be found more easily through search. This is backed up by Shackleton et al. (2000) who find that adding neutrality through redundancy can help improve the level of fitness found through artificial evolution.

Neutrality has also been shown in real-world problem landscapes: In experiments on evolving tone recognition circuits, populations were seen to move in genotype space during periods where fitness did not increase (Harvey and Thompson, 1996). Experiments on evolution of digital circuits under two conditions - allowing neutral mutations and not allowing such changes - have also shown the importance of neutrality to the search process. Experiments where neutral changes were allowed consistently produced two-bit multiplier solutions of higher fitness than experiments without neutral mutation (Vassilev and Miller, 2000). Also, Thompson (2001) reports on the presence of neutral plateaus in evolving speech recognition circuits, and shows

through exhaustive mutation of genotypes at the start of the plateau that no transitions exist to a higher fitness level; the neutral evolution phase is necessary.

The degree of neutrality in a system is clearly a factor in the population dynamics during evolution. However, many measures aimed at predicting the difficulty of search in a particular space do not account for this factor, e.g. the work on NKp landscapes showing that the autocorrelation function does not change with the level of neutrality (Barnett, 1998). In the next section, we describe how the concept of evolvability can be used to derive measures aimed at predicting the difficulty of searching in a space, taking account of the level of neutrality in the space.

3 Evolvability and the transmission function

Evolvability is loosely defined as the capacity to evolve, alternatively the ability of an individual or population to generate fit variants (Altenberg, 1994; Marrow, 1999; Wagner and Altenberg, 1996). Thus evolvability is more closely allied with the *potential* for fitness than with fitness itself; two equal fitness individuals or populations can have very different evolvabilities (Turney, 1999). Typically, researchers use some definition of evolvability based on the offspring of current individuals or populations: in this paper we follow Cavalli-Sforza and Feldman (1976) and Altenberg (1994) in using the *transmission function* of all possible offspring from a parent to define a set of metrics of evolvability, see section 3.1 for further details.

It is often argued that there may be long-term trends for evolvability to increase during evolution (see e.g. Wilke, 2001; Turney, 1999). However, as evolvability is more directly related to fitness potential than fitness itself, long-term change cannot be due to straight fitness selection. Thus any trend towards change in evolvability can only be understood through some second order selection mechanism, by which evolution tends to select solutions that have a more evolvable genetic system (Dawkins, 1989; Kirschner and Gerhart, 1998).

Researchers in both biology and evolutionary computation typically link evolvability with the properties of the local search space. For example, Burch and Chao (2000) shows that RNA virus evolvability can be understood in terms of the mutational neighbourhood, while many evolutionary computation researchers (see e.g. Ebner et al., 2001; Marrow, 1999) argue that changing the properties of the search space (through such mechanisms as adding neutrality) can affect evolvability as evidenced by the speed of evolution. The interest in evolvability for evolutionary computation practitioners is thus tied closely to work on the ruggedness and modality of the search space, argued to primarily influence the ease of finding good solutions in the space (Weinberger, 1990; Hordijk, 1996; Jones and Forrest, 1995; Naudts and Kallel, 2000).

Recent work has emphasised that in addition to landscape ruggedness and modality, search space neutrality may have impact on the population dynamics of evolution (section 2.2). This factor may not be predicted by standard measures based on the landscape ruggedness and local modality, but may be measurable through change in evolvability. For example, recent artificial evolution research has shown that evolvability can change during neutral epochs; populations tend to move to "flatter" areas of the fitness landscape where fewer mutations are deleterious (Wilke et al., 2001; Wilke, 2001). This can clearly have an impact on the speed of search, but may not be picked up by the standard landscape ruggedness and modality measures. In this paper, we define and apply measures based on the evolvability of solutions, in order to investigate whether we can account for the observed speed of evolution differences between two different

robotic controller architectures. The search spaces defined by these two genetic systems show a high degree of neutrality, and no differences are seen using standard measures of ruggedness and modality (Smith et al., 2001a). Further details of the genotype-to-fitness mapping are given in section 4.

Other biological research in evolvability is also of relevance to evolutionary computation, e.g. the work on adaptation to change in environment through such mechanisms as alleles providing increased mutation rates (Taddei et al., 1997; Sniegowski et al., 1997). However, in this paper we focus on evolvability in terms of the properties of the solutions' local search space. The next section outlines the offspring transmission function, and defines a simple set of evolvability metrics.

3.1 The transmission function

In this paper, we follow the definition of evolvability as the ability of individuals and populations to produce fit variants, specifically the ability to both produce fitter variants, and to not produce less fit variants. This definition is intimately tied in with research on the *transmission function* T (Altenberg, 1994; Cavalli-Sforza and Feldman, 1976), and the population offspring probability distribution function ϕ from all possible applications of the genetic operators to the parent(s):

$$\phi(g, f) = \int\int\int\int \psi(h, k, h', k')\, T(g, f : h, k, h', k')\, dh\, dk\, dh'\, dk' \tag{1}$$

or the probability ϕ (with parental selection function ψ) of obtaining offspring genotype g and phenotype f, over all parents of genotypes h, h' and phenotypes k, k'. The transmission function T is the probability density function of obtaining g, f given h, k, h', k' (Cavalli-Sforza and Feldman, 1976).

In the absence of recombination, only a single parent h, k is required to produce offspring through mutation:

$$\phi(g, f) = \int_{-\infty}^{\infty} \psi(h, k)\, T(g, f : h, k)\, dh\, dk \tag{2}$$

or the probability of obtaining offspring g, f over all parents h, k with selection ψ. In this paper, we focus on the offspring of a set of single genotypes (saved during the course of evolutionary runs), so do not integrate over the set of all possible parents. Similarly, the selection function can be omitted as we pre-select the parent. Since we are interested only in the offspring phenotypes f, and not the offspring genotypes g, we can refer to the transmission function $T(f : h, k)$ as short-hand for the probability density function of offspring fitnesses from a single parent h, k.

The transmission function thus encompasses both the operators and the representation; instead of referring to good and bad genetic operators or good and bad representations, we can talk about the effectiveness of the transmission function. Thus the evolvability of an individual or population, i.e. their ability to generate fit variants, is simply a property of the individual or population transmission function. The next section derives measures for the evolvability of an individual solution in terms of this transmission function for continuous variables.

3.2 Evolvability metrics

The evolvability of a solution genotype h and fitness k is directly tied to the probability of that solution not producing offspring of lower fitness. Thus we derive our first metric of evolvability E_a:

$$E_a = \frac{\int_k^\infty T(f:h,k)\,df}{\int_{-\infty}^\infty T(f:h,k)\,df} \tag{3}$$

or the probability that the offspring fitness f is greater or equal to the current fitness k, i.e. the mutation is non-deleterious. Since the transmission function $T(f:h,k)$ is a probability density function, the infinite integral sums to unity, so we have:

$$E_a = \int_k^\infty T(f:h,k)\,df \tag{4}$$

Low fitness solutions may have a larger E_a than high fitness solutions, simply due to the increased number of better mutations. The second evolvability metric E_b uses only the offspring fitnesses:

$$E_b = \int_{-\infty}^\infty f\,T(f:h,k)\,df \tag{5}$$

or the expected offspring fitness from genotype h. Note, this value is fitness dependent, so should not be compared across genotypes without reference to their original fitness. A further problem with both E_a and E_b is their dependence on the entire set of offspring fitnesses; the fraction of offspring that are significantly fitter than the parent may be extremely small. The third measure reflects this dimension of evolvability, looking only at the top C'th percentile of the offspring fitnesses:

$$E_c = \frac{100}{C}\int_{F_c}^\infty f\,T(f:h,k)\,df \tag{6}$$

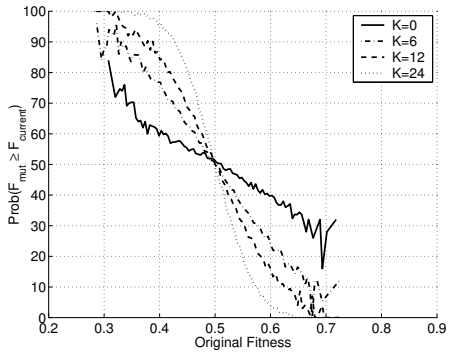
$$\text{with } \mathrm{F_c} \text{ defined by } \int_{F_c}^\infty T(f:h,k)\,df = \frac{C}{100} \tag{7}$$

or the expected fitness of only the top C'th percentile of fitnesses. A similar measure E_d (not shown) calculates the expected fitness of the bottom C'th percentile of offspring.

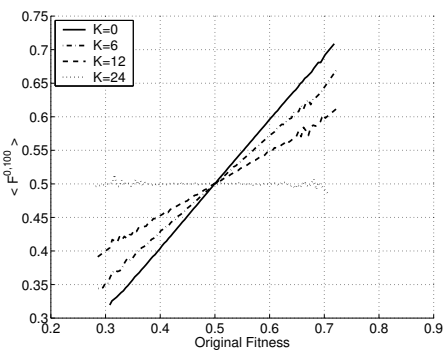
We define the population evolvability metrics as the evolvability calculated over the sum of the individual transmission functions (see Smith et al., 2001b, for further details). It is also straightforward to extend the analysis to the discrete set approximation of the transmission function T (again see Smith et al., 2001b, for details). In the next section we apply the defined evolvability metrics to a tunably rugged and neutral variant of the NK landscapes (Kauffman, 1993; Newman and Engelhardt, 1998), as a prelude to the real focus of the paper: applying the evolvability metrics to the robotic controller architecture search spaces described in section 4.

3.3 Evolvability of NK landscapes

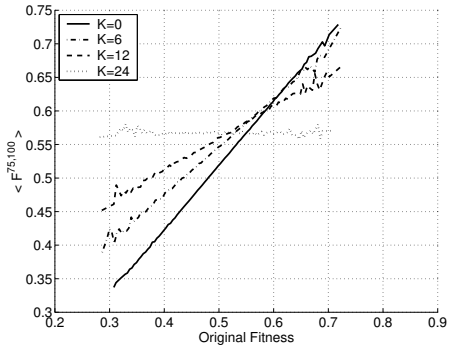
The four metrics derived above have been used to correctly predict difficulty of search in the theoretical NK landscapes through random sampling of the space (Smith et al., 2001b), and agree with analytically derived values for the metrics over the NK landscapes, see figure 3.



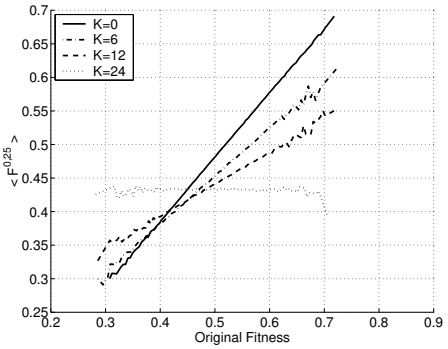
(a) Probability of a non-deleterious mutation, E_a



(b) Expected fitness over all mutations, E_b



(c) Expected fitness over top quartile of mutations, E_c

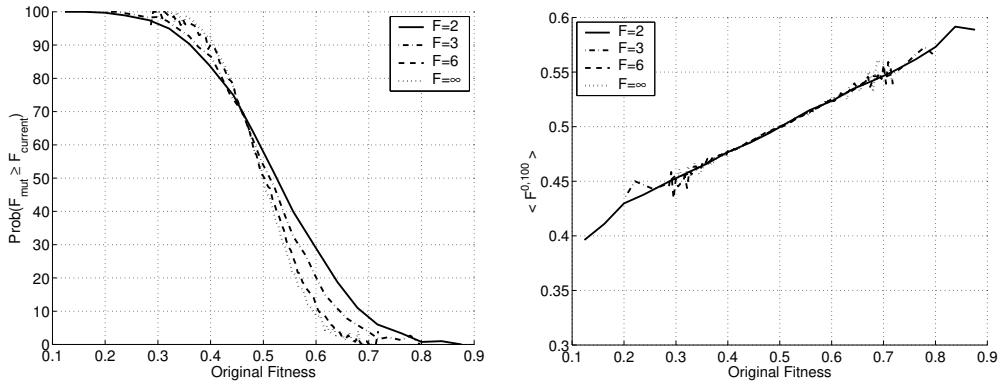


(d) Expected fitness over bottom quartile of mutations, E_d

Figure 3: The metrics of evolvability derived in section 3.1 applied to the NK landscapes (Kauffman, 1993). See text for details.

The graphs show the four evolvability metrics calculated on a set of individual genotypes collected through randomly sampling 1000 genotypes from 100 landscapes ($N = 25$, $K = \{0, 6, 12, 18, 24\}$). The graphs correctly predict the changing evolvability at different fitness heights in the landscape: Figure 3(a) shows that the probability of finding a non-deleterious mutation at high fitnesses ($F > 0.5$), is highest for the $K = 0$ landscape. However, for low fitnesses ($F < 0.5$), the maximally rugged $K = N - 1 = 24$ landscape provides the greatest probability of non-deleterious mutations. Similarly, the expected mutation fitness at high fitnesses ($F > 0.5$), is highest for the $K = 0$ landscape, but highest for the rugged $K = 24$ landscape at low fitnesses ($F < 0.5$), see figure 3(b). Figures 3(c) and 3(d) show a similar story: the expected fitness for the top and bottom quartiles of all mutations are highest for the $K = 0$ landscape only at high starting fitnesses.

Thus the evolvability metrics provide more information than measures of correlational structure in the landscapes, giving detail on how easy it will be to find solutions of varying fitness. In particular, they correctly predict that the time required to find good solutions for varying K will depend on how good a solution we need: for low fitnesses a maximally rugged landscape is best, but as our required fitness increases, a smaller degree of epistasis results in faster search. Note, this is of course affected by random solutions already having expected fitnesses of 0.5, but holds if we are starting from low fitness solutions.



(a) Probability of non-deleterious mutation, E_a

(b) Expected fitness over all mutations, E_b

Figure 4: The metrics of evolvability derived in section 3.1 applied to a tunably neutral version of the NK landscapes (Newman and Engelhardt, 1998). See text for details.

Figure 4 shows the evolvability metrics for a set of neutral "terraced" NK landscapes (Newman and Engelhardt, 1998), in which the fitnesses for each loci are chosen from a discrete set of values lying in the range $[0, 1]$, rather than the continuous range used in the standard NK landscape. The number of discrete values, or terraces, F is thus a measure of the neutrality in the system; low F corresponds to landscapes of high neutrality, while $F = \infty$ corresponds to the classical NK landscape. The probability of obtaining a non-deleterious mutation (figure 4(a)), and the expected mutation fitness (figure 4(b)), are plotted for $F \in \{2, 3, 11, \infty\}, N = 24, K = 18$. The varying level of neutrality does not affect the expected fitness of all mutations, but does affect the probability of obtaining a non-deleterious mutation. This change in probability is non-linear with fitness, similar to the change seen for high K in figure 3(a). At low fitnesses, the highly neutral landscapes have smaller probability of non-deleterious mutation. However, at higher fitnesses the neutrality in the landscape is more useful, and the more neutral landscapes show higher probability of non-deleterious mutations; at a fitness of 0.6 the $F = 2$ landscape shows nearly 30% of such mutations compared with less than 10% for the $F = \infty$ landscape. This is likely to have a significant effect on the time required to find solutions of high fitness, as seen by Newman and Engelhardt (1998). Again, the graphs show close agreement with the derived analytical versions (Smith et al., 2001b).

In the robotics search spaces we go on to analyse in sections 5 and 7, there is a high degree of neutrality as evidenced by the long neutral epochs during which evolution proceeds through non-adaptive mutation. Thus being able to determine the difficulty of finding good solutions in spaces with neutrality, as shown above, is of prime importance. The next section describes the

applied evolutionary robotics genotype-fitness mapping used in the work presented here.

4 An evolutionary robotics search space

One of the new styles of Artificial Intelligence to have emerged recently is evolutionary robotics (Cliff et al., 1993; Nolfi and Floreano, 2000; Floreano and Mondada, 1994; Husbands and Meyer, 1998). The evolutionary process involves evaluating, over many generations, whole populations of robot control systems specified by artificial genotypes. These are interbred using a Darwinian scheme in which the fittest individuals are most likely to produce offspring. Fitness is measured in terms of how good a robot's behaviour is according to some evaluation criterion.

In previous work, we have investigated the use of different neural network architectures, focusing on developing control structures that produce successful solutions in fewer evaluations using artificial evolution (Husbands et al., 1998). In experiments on a variety of robotics tasks, we have shown that a particular style of network, the "GasNet" (section 4.2), is particularly amenable to evolutionary search.

Our primary reason for applying the evolvability analysis developed in the earlier sections is to explain the differences in speed of evolution for the GasNet and NoGas neural networks in terms of their underlying search spaces. The second aim of the paper is to investigate how a search space defined by an extremely complex genotype-to-fitness mapping differs from landscapes such as the NK systems, which are derived primarily for theoretical analysis. In particular, it is not at all clear whether properties of more theoretical landscapes will be observed for the mapping used here, in which the initial genotype translates to an intermediate neural network phenotype, with the final fitness measuring how well this network performs over time in controlling a robot engaged in solving a visual shape discrimination task.

4.1 Evolutionary robotics control architectures

Artificial neural networks have been successfully used in a large number of evolutionary robotics experiments (for an overview see Nolfi and Floreano, 2000). Typically, external sensory data is used for the network input, and the network output is used to control the robot motors. Other styles of control architectures have also been used for evolutionary robotics experiments, notably genetic programming (Koza, 1992) and classifier systems (Holland, 1992).

4.2 The GasNet and NoGas architectures

The networks used in the experiments described later are discrete time step dynamical systems built from units connected together by links that can be excitatory (with a weight of +1) or inhibitory (with a weight of -1). The output, O_i^n, of node i at time step n is a function of the sum of its inputs, as described by equation 8. This defines the basic "NoGas" architecture:

$$O_i^n = \tanh\left[k_i^n\left(\sum_{j\in C_i} w_{ji}O_j^{n-1} + I_i^n\right) + b_i\right] \tag{8}$$

In the "GasNet" control system, in addition to this underlying network in which positive and negative 'signals' flow between units, an abstract process loosely analogous to the diffusion of gaseous modulators is at play (Philippides et al., 2000). Some units can emit 'gases' which diffuse and are capable of modulating the behaviour of other units by changing their transfer functions in ways described in detail later. This form of modulation allows a kind of plasticity in the network in which the intrinsic properties of units are changing as the network operates. The networks function in a 2D plane; their geometric layout is a crucial element in the way in which the 'gases' diffuse and affect the properties of network nodes.

Where C_i is the set of nodes with connections to node i, I_i^n is the external (sensory) input to node i at time n, and b_i is a genetically set bias. Each node has a genetically set default transfer function parameter k_i^0. As will seen later, the value k_i^n for each node can be changed by diffusing gases as the network runs. Thus the actual shape of the node's transfer function is altered via the gas modulation mechanism.

4.2.1 Gas diffusion in the networks

It is genetically determined whether or not a node will emit one of two 'gases' (gas 1 and gas 2), and under what circumstances emission will occur (either when the 'electrical' activation of the node exceeds a threshold, or the concentration of a (genetically determined) gas in the vicinity of the node exceeds a threshold). The electrical threshold used in the experiments described later was 0.5, the gas concentration threshold 0.1.

A very abstract model of gas diffusion is used. For an emitting node, the concentration of gas at distance d from the node is given by equation 9. Here, r is the genetically determined radius of influence of the node, so that concentration falls to zero for $d > r$. This is loosely analogous to the length constant of the natural diffusion of NO, related to its rate of decay through chemical interaction. $T(t)$ is a linear function that models the build up and decay of concentration after the node has started/stopped emitting (equation 10 and 11). The slope of this function is individually genetically determined for each emitting node, C_0 is a global constant.

$$C(d,t) = \begin{cases} C_0 \times e^{\frac{-2d}{r}} \times T(t) & d < r \\ 0 & \text{else} \end{cases} \tag{9}$$

$$T(t) = \begin{cases} H(\frac{(t-t_e)}{s}) & \text{emitting} \\ H(H(\frac{(t_s-t_e)}{s}) - H(\frac{(t-t_s)}{s})) & \text{not emitting} \end{cases} \tag{10}$$

$$H(x) = \begin{cases} 0 & x \leq 0 \\ x & 0 < x < 1 \\ 1 & \text{else} \end{cases} \tag{11}$$

where t_e is the time at which emission was last turned on, t_s is the time at which emission was last turned off, and s (controlling the slope of the function) is genetically determined for each node.

In other words, the 'gas' concentration varies spatially as a Gaussian centred on the emitting node. The height of the Gaussian at any point within the circle of influence of the node is

linearly increased or decreased depending on whether the node is emitting or not. Note $T(t)$ saturates at a maximum of 1 and a minimum of 0. The total concentration at any point in the network is found by summing the concentrations from all emitting nodes.

4.2.2 Modulation by the gases

The transfer parameter value for the ith node at time step n, k_i^n (see equation 8), is changed (or *modulated*) by the presence of gases at the site of the node. Gas 1 increases the value of k_i^n in a concentration dependent way, while gas 2 decreases its value. This modulation is described by equations 12 to 14 and happens on every time step as the network runs. This provides a form of plasticity very different from that found in most traditional artificial neural networks.

$$k_i^n = \mathbf{P}[index_i^n], \quad \mathbf{P} = \{-4.0, -2.0, -1.0, -0.5, -0.25, 0.0, 0.25, 0.5, 1.0, 2.0, 4.0\} \tag{12}$$

where,

$$index_i^n = f\left(index_i^0 + \frac{C_1^n}{C_0 \times K}(N - index_i^0) - \frac{C_2^n}{C_0 \times K} index_i^0\right) \tag{13}$$

$$f(x) = \begin{cases} 0 & x \leq 0 \\ \lfloor x \rfloor & 0 < x < N \\ N & \text{else} \end{cases} \tag{14}$$

where $\mathbf{P}[i]$ refers to the ith element of set $\mathbf{P}$, $index_i^n$ is node i's index into the set $\mathbf{P}$ of possible discrete values k_i^n can assume, N is the number of elements in $\mathbf{P}$, $index_i^0$ is the genetically set default value for $index_i$, C_1^n is the concentration of gas 1 at node i on time step n, C_2^n is the concentration of gas 2 at node i on time step n, and C_0 and K are global constants (both set to 1 in this study). So, $index_i^n$ increases in direct proportion to the concentration of gas 1, and decreases linearly with respect to the concentration of gas 2. In this way the value of k_i^n is changed over time by the presence of gases at the node's site (the concentrations are governed by equation 9).

In a variety of robotics tasks (Husbands, 1998; Husbands et al., 1998; Smith and Philippides, 2000), GasNet controllers evolve significantly faster than networks without the gas signalling mechanism. The next section describes the task used in the work presented here, while section 4.6 details previous results for the speed of evolution.

4.3 The robotics task

The evolutionary task at hand here is a visual shape discrimination task; starting from an arbitrary position and orientation in a black-walled arena, the robot must navigate under extremely variable lighting conditions to one shape (a white triangle) while ignoring a second shape (a white square). Both the robot control network, an arbitrarily recurrent neural network incorporating artificial diffusing neuromodulators, and the robot sensor input morphology, i.e. the number and position of input pixels used in the visual array, were under evolutionary control. Fitness over a single trial was taken as the fraction of the starting distance moved towards the

triangle by the end of the trial period, and the evaluated fitness was returned as the weighted sum of 16 trials of the controller from different initial conditions:

$$F = \frac{1}{136} \sum_{i=1}^{i=16} i \frac{D_i^F}{D_i^S} \tag{15}$$

where D_i^F is the distance to the triangle at the end of the ith trial, and D_i^S the distance to the triangle at the start of the trial, and the i trials are sorted in descending order of $\frac{D^F}{D^S}$. Thus good trials, in which the controller moves some way towards the triangle, receive a smaller weighting than bad trials, encouraging robust behaviour on all 16 trials.

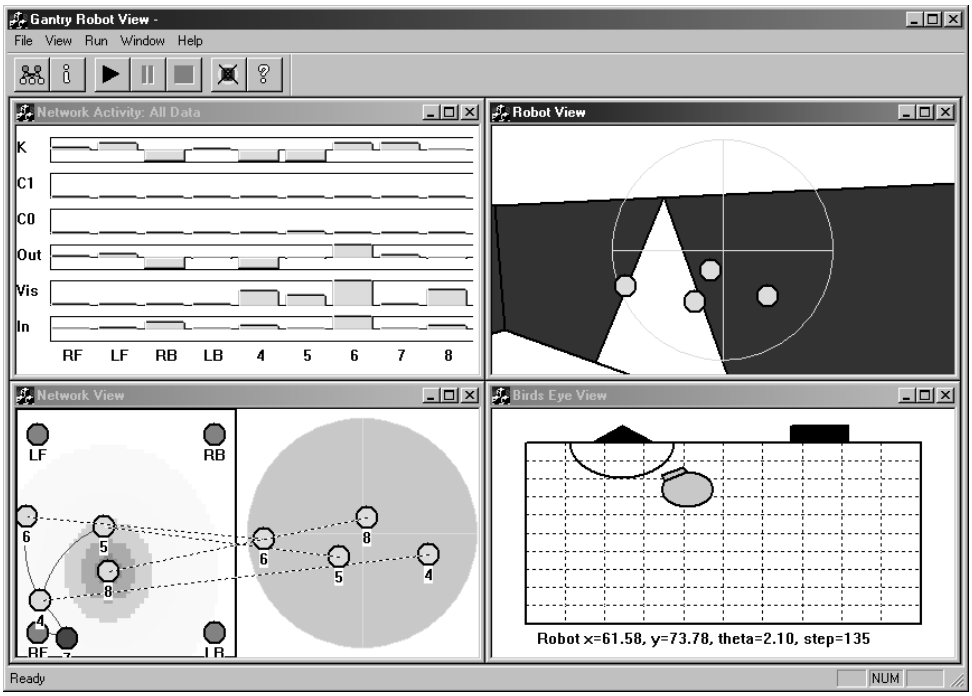


Figure 5: Screen shot of the simulated arena and robot. The bottom-right view shows the robot position in the arena with the triangle and square. Fitness is evaluated on how close the robot approaches the triangle. The top-right view shows what the robot 'sees', along with the pixel positions selected by evolution for visual input. The top-left view shows the current activity of all nodes in the neural network. The bottom-left view shows the robot control neural network: the visual input positions in the camera are shown on the right, with the nodes they connect to placed in the network plane on the left. The motor output nodes *RF*, *LF*, *RB* and *LB* are shown in the four corners of the network plane, and high gas concentrations are shown by shading, e.g. surrounding node 8. See text for further details of the task and network controllers.

Evaluations are carried out in a *minimal simulation* (Jakobi, 1998), with large amounts of noise added to sensor and motor readings, so that controllers will transfer to robots operating in the real environment. Figure 5 shows a screen shot of a simulated evaluation. As in many problems requiring controllers to provide sensor-to-motor mappings over time, fitnesses are extremely time consuming to evaluate (in the work presented here, evaluating a sample of 10^6 fitnesses takes around 24 hours on a Pentium II 700MHz machine) and inherently extremely noisy. Success in the task was taken as an evaluated fitness of 1.0 over thirty successive generations of the genetic

```
- Initialise population of 100 solutions on 10x10 grid.
- Evaluate each solution fitness.
- Repeat until success criterion met, or MaxGenerations reached:
  - Repeat 100 times for 1 generation:
    - Select solution at random.
    - Create mating pool of solution plus 8 nearest grid neighbours.
    - Pick parent P through rank-based roulette wheel selection on mating pool.
    - Create offspring O through mutation of P, and evaluate fitness.
    - Place O in 10x10 grid, replacing mating pool solution picked through
          inverse rank-based roulette wheel selection.
```

Figure 6: The genetic algorithm pseudo-code.

algorithm.

4.4 The genetic algorithm

A distributed asynchronous updating genetic algorithm was used, with a population of 100 solutions arranged on a 10×10 grid. Fitness was awarded on the fraction of the distance moved towards the triangle over a series of 16 runs with different initial conditions, see equation 15. Parents were chosen through rank-based roulette-wheel selection on the mating pool consisting of the 8 nearest neighbours to a randomly chosen grid-point. The child solution was a mutated copy of the parent (see section 4.5 for details of the mutation operator) and placed back in the mating pool using inverse rank-based roulette-wheel selection. One generation was specified as 100 such breeding events. Figure 6 shows the pseudo-code for the genetic algorithm.

4.5 The solution representation and mutation operator

The neural network robot controllers were encoded as variable length strings of integers, with each integer allowed to lie in the range $[0, 99]$. Each node in the network was coded for by nineteen parameters, controlling such properties as node connections, sensor input, and node bias. For the NoGas networks, certain parameters coding for the gas diffusion details were simply ignored (note that this does not affect the mutation rate; as described below the rate is specified per genotype loci). In all experiments, the GA population were initially seeded with networks containing ten neurons. For further details see Husbands et al. (1998); Smith and Philippides (2000).

Three mutation operators were applied to solutions with probability μ during evolution (for most experiments detailed here, $\mu = 4\%$). First, each integer in the string had a $\mu\%$ probability of mutation in a Gaussian distribution around its current value (20% of these mutations completely randomised the integer). Second, there was a $\mu\%$ chance per *genotype* of adding one neuron to the network, i.e. increasing the genotype length by 19. Third, there was a $\mu\%$ chance per genotype of deleting one randomly chosen neuron from the network, i.e. decreasing the genotype length by 19. It should be noted that the value of $\mu = 4\%$ used in these experiments is a much larger level of mutation than typically used in artificial evolution optimisation (and certainly much larger than in biological evolution). However, lower levels of mutation produce little change

in the mutated offspring networks when compared to the parents: table 1 gives some evidence that even higher mutation rates are useful.

4.6 Previous results: GasNets are faster

Over a large sample of evolutionary runs with GasNet and NoGas conditions, GasNet networks allowed to use the gaseous signalling mechanism reached success significantly faster than the NoGas networks (remember from section 4.3 that success is defined as 100% fitness over 30 consecutive generations). This speed difference was seen in several different evolutionary robotics scenarios (e.g. Husbands, 1998; Smith and Philippides, 2000) and over several different mutation rates, e.g. see table 1.

Condition	$\mu = 0.01$	$\mu = 0.02$	$\mu = 0.04$	$\mu = 0.08$	$\mu = 0.16$
GasNet	7,354	3,436	675	449	671
NoGas	>10,000	9,510	1,228	1,656	831

Table 1: Median number of generations required for success for the GasNet and NoGas architectures, shown for several different mutation rates μ.

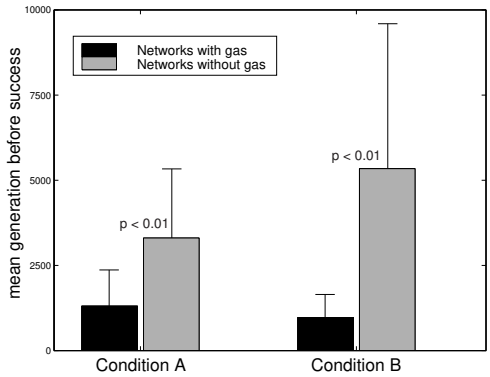


Figure 7: Mean number of generation required for success for GasNets and NoGas networks, under two different network connectivity schemes. Error bars show one standard deviation. Note that the mean differences are bigger than the median differences reported in table 1, as the distribution of times taken to reach high fitness was not normally distributed.

It should be emphasised that there is a significant difference in the times required to evolve successful controllers for the task. Over many mutation rates, the GasNets are several times faster than the NoGas versions. Figure 7 shows the differences in mean number of generations required to evolve successful controllers. It is this difference in time required for the evolution of successful controllers that we want to explain in terms of the underlying search spaces. The next section describes the evolvability analysis used in this paper, applying the derived evolvability metrics to the two evolutionary robotics search spaces.

5 Evolvability analysis of a search space

In previous work (Smith et al., 2001a), we have shown that several standard analyses (correlation structure (Weinberger, 1990), local modality (Weinberger, 1991), and information structure (Vassilev et al., 2000)), derived to predict the difficulty of finding good solutions in a given space, fail to predict the evolutionary speed differences described in section 4.6. There are three primary reasons for these failures. First, the fitness distribution of the space is extremely skewed, with the vast majority of genotypes having extremely low fitness: statistics calculated through random sample techniques fail to explore the interesting parts of the space. Second, the large amount of noise inherent in the fitness evaluation tends to obscure measures of the space ruggedness and local modality. Third, most of the evolutionary runs show extremely long periods during which fitness does not apparently increase. Barnett (1998) has shown the unreliability of correlation structure measures when applied to spaces with neutrality; if these periods of no apparent fitness increase are indeed neutral epochs then we might expect many standard measures to fail when predicting search difficulty.

In spaces with a high degree of neutrality, it may well be that analysis of the evolvability of solutions will predict the difficulty of finding successful solutions. This is backed up by the evidence presented on the evolvability of tunably neutral NK landscapes, shown in section 3.3, which tallies with the time required to evolve good solutions (Newman and Engelhardt, 1998). Thus we frame the hypothesis that evolvability analysis of the two search spaces we are interested in, namely the GasNet and NoGas conditions, may well explain the differences in time required for evolution of good solutions.

The rest of this paper focuses on applying the evolvability metrics derived in section 3.1 to the two evolutionary robotics spaces at hand. The three questions we address are: First, do the periods during which fitness does not apparently increase correspond to neutral epochs, i.e. do the spaces show a high degree of neutrality? Second, if these periods are neutral epochs, how does evolvability change during over the evolutionary run, both during periods during which fitness increases and during these neutral epochs? Third, do the evolvability metrics predict differences between the GasNet and NoGas search spaces, which lead to the observed evolutionary speed differences?

During the course of twenty evolutionary runs, ten with the gas mechanism active and ten without, we have collected a large sample of genotype populations. The following sections apply the evolvability metrics to the saved individuals and populations. Section 6 analyses a single evolutionary run in detail, showing that the run can be characterised by a long neutral epoch, with two rapid periods of fitness increase. We also investigate how evolvability changes over the run, and see if the pattern is repeated over a large sample of runs. Finally, section 7 analyses the evolvability of Gas and NoGas samples, to see if differences in the spaces are seen.

6 Analysis of a single evolutionary run

Figure 8 shows the population best and mean evaluated fitnesses over generations for an evolutionary run chosen at random from the sample of GasNet runs. Both fitnesses climb quickly from an initial near-zero random performance, then stay fairly constant over the next 450 generations apart from the high levels of evaluation noise. The fitness increase seen at generation

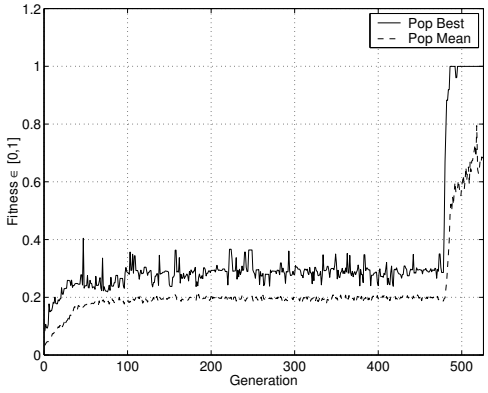


Figure 8: Best and mean fitness of the population over generations. See text for details of the fitness function.

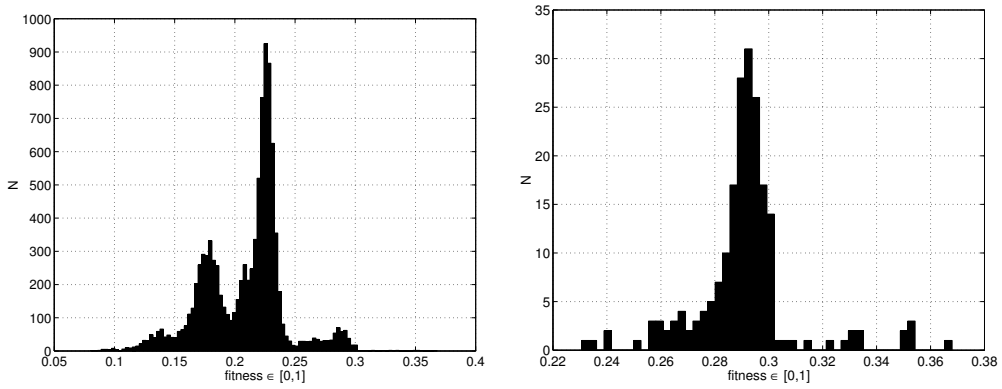
477 leads extremely quickly to the best individual solutions reaching 100% fitness.

First, we show that the periods between generations 100 and 477 can be regarded as a neutral epoch (section 6.1), and investigate the population behaviour during this epoch, showing that the population moves significantly through the space (section 6.2). Section 6.3 applies the evolvability metrics derived in section 3.1, and section 6.4 compares the evolvability across the neutral epoch predicted by the metrics, with results obtained from repeating evolution with different starting populations.

6.1 Results: A neutral epoch?

The best and mean fitnesses of the population (figure 8) between generations 50 and 477 appear to show no significant increase; this section investigates whether this apparently neutral phase is really a neutral epoch. We investigate the variance in evaluated fitness shown by a single genotype, and relate this to the variance seen in the population.

Figure 9 shows the distribution of fitnesses from multiple evaluations of a single genotype, with all fitnesses plotted (figure 9(a)), and only the best fitness every 50 evaluations plotted (figure 9(b)). Making the approximation that half the population has a similar high level of fitness (supported by the relatively high mean fitness over the population) allows us to make the link between the best fitness every 50 evaluations of the single genotype to the fitness of the best individual in a single generation. Note that changing this approximation and plotting only the best fitnesses over some other number of evaluations has little effect on the analysis, beyond changing the lower limit of the neutral band. The fitnesses seen for a single genotype gives us a possible range for the band of neutral fitness; figure 10 shows this range applied to the evolution fitness graph (the grey band). Only one evaluation before the large increase in fitness at generation 478 lies above this range (the 0.40 fitness seen at generation 46), with all others lying in the band. This is an important result: A single genotype evaluated multiple times can give rise to the variation in fitness seen before generation 478. If a single genotype can give rise to such variation, we can consider the range as a neutral band of fitness; the population is in a non-adaptive phase of evolution. However, figure 10 also shows the rolling mean over the last 30 generations of the fittest individual in the population (calculating the rolling mean over other numbers of generations show similar results), implying that there may be more than a single



(a) All evaluations

(b) Best of 50 evaluations

Figure 9: The fitness distribution histogram of a single genotype evaluated $10,000$ times. **(a)** All evaluations plotted, and **(b)** Only the best fitness every 50 evaluations plotted. Fitness ranges from $[0.23, 0.37]$ and was used to calculate the range of the neutral band in figure 10. Possible fitness $\in [0, 1]$.

neutral network within this band; the first lower fitness band lies roughly between generations 50 and 100, the second between generations 100 and 477. No further analysis is done on possible multiple neutral networks in the band; this would be an interesting area of further study.

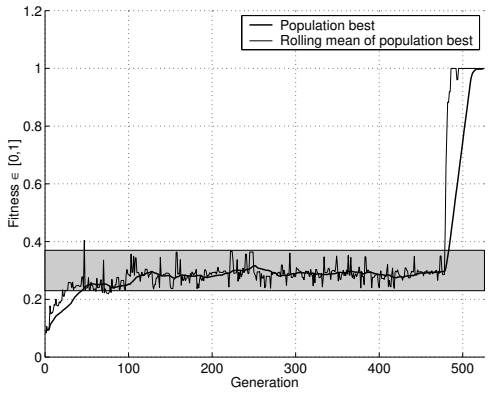


Figure 10: Best fitness of the population plotted with rolling mean over the last 30 generations, showing at least two distinct neutral plateaus, with fitnesses roughly of 0.25 and 0.3. The neutral band (grey block) was calculated from the range of fitnesses found over the mutations performed on a single genotype (figure 9).

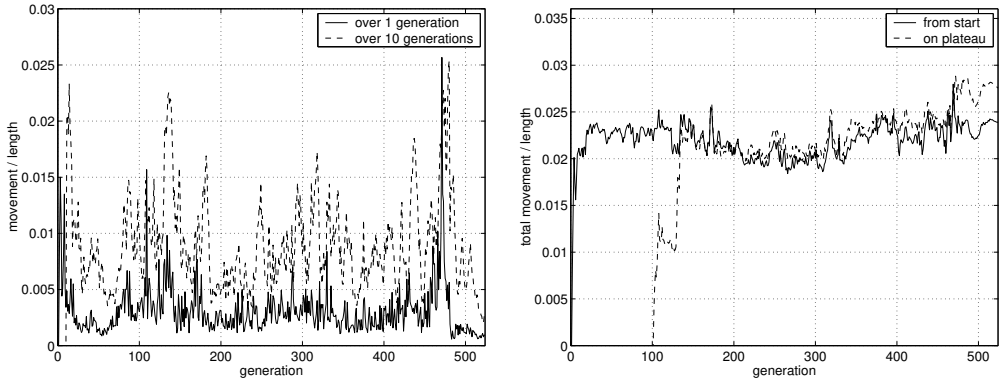
Thus we identify the main neutral phase of evolution as lying between generations 100 and 477. The next section explores the dynamics of the evolutionary population during this period; is it simply stuck in a local optimum; or moving in genotype space?

6.2 Results: Neutral epoch population movement

The key identifying feature for the population diffusing along a neutral network is the movement of the population in genotype space. This is typically measured through the population

diffusion coefficient, calculated through the distance moved by the population centroid each generation. However, in this scenario we have variable length genotypes (section 4.5). The approach used here is to calculate the centroid movement separately for each genotype length present in the population; results shown are for length 171 genotypes which are present on nearly every generation. Results for other lengths were similar.

Figure 11 shows the distance moved by the population centroid. Figure 11(a) shows the distance on each generation between the current 171 length genotype centroid and the centroids from both the previous generation and 10 generations earlier, scaled by the genotype length. The population is clearly moving from generation to generation during the neutral phase, and the ten-generation movement shows it is not just moving back and forth. Note that the peak occurring at generation 471 does not coincide with the increase in fitness, but occurs just before.



(a) Centroid movement per generation

(b) Total centroid movement

Figure 11: The Euclidean distance moved by the population centroid: **(a)** The distance between the current generation centroid, and centroids on both the previous generation and 10 generations earlier, and **(b)** The distance between the current generation centroid, and the centroids on both the start generation and the start of the neutral epoch. Calculated for length 171 genotypes only, and scaled by the genotype length.

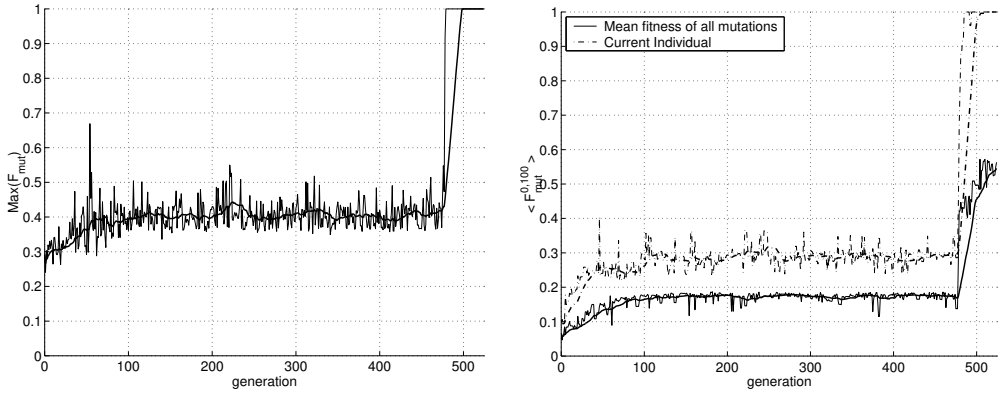
Figure 11(b) shows the distance moved by the centroid from the start of the entire run, and from the start of the neutral plateau (taken to be generation 100 in section 6.1). The plateau seen in the graph around a divergence of 0.025 is due to the space not being infinite in size; as the population moves away from the initial population, it can only move a finite distance related to the average distance between two randomly chosen unit vectors. This distance for N-dimensional vectors is $\sqrt{N}/3$, so scaled by the length the distance is $(3\sqrt{N})^{-1}$, which equals 0.025 for the length 171 genotype vectors used here. Thus the population is moving significantly during the neutral phase. Results from the other length genotypes are extremely similar. Note that the high mutation rates described in section 4.5, and the possibility of adding and deleting whole sections of the genotype, make this movement higher than in other studies of population movement during artificial evolution (see e.g. Barnett, 1998).

Having established that the period from generations 100 to 477 can be regarded as a neutral epoch during which the population moves significantly, we now apply the evolvability metrics defined in section 3.2 to the best-of-generation individuals from each generation across the evolutionary run. We focus on the behaviour of evolvability during the neutral epoch, and

during the two brief periods during which fitness increases.

6.3 Results: Evolvability over the evolutionary run

For the best individual of each generation, the transmission functions were approximated through recording fitnesses from 100,000 applications of the mutation operator. Figure 12 shows the highest and expected mutation fitnesses across generations (rolling means over 20 generations have been added to aid clarity). The graphs closely follow the best individual fitness, rising sharply during the initial period of fitness increase, then staying roughly constant once the neutral epoch is reached around generation 100 (although there is a single high mutation fitness found just after generation 60).



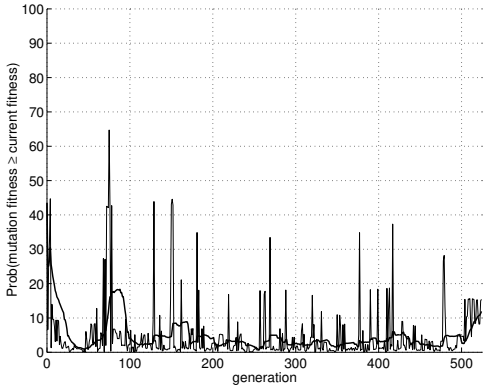
(a) Highest mutation fitness

(b) Expected fitness over all mutations, E_b

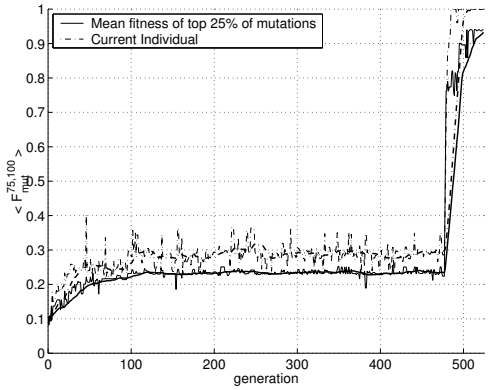
Figure 12: 100,000 mutations were applied to each of the best-of-generation individuals, approximating each individual transmission function distribution of offspring fitnesses. **(a)** Shows the highest mutation fitness found, and **(b)** shows the expected mutation fitnesses (current individual fitnesses are shown for comparison). Neither graph shows a clear trend during the neutral epoch over generations 100-477, but both increase dramatically during the periods during which fitness increases. Rolling means over 30 generations have been added to aid clarity.

Two evolvability metrics derived in section 3.1 are shown in figure 13; the probability of obtaining a better mutation, and the expected fitness over the top quartile of mutations. Figure 14 shows the expected fitness over the bottom quartile of mutations.

We see that the probability of obtaining a better fitness (figure 13(a)) does not show any obvious trends over either the neutral or fitness-increasing epochs, although the spike around generation 60 coincides with the high fitness seen in figure 12. The expected fitness value over the upper quartile of the transmission function (figure 13(b)) does not change over the neutral epoch, but increases during the fitness-increasing epochs. However, if anything, the expected fitness over the bottom quartile of mutations shows a slight decrease over the neutral epoch, and decreases dramatically during the increase in fitness around generation 477. This behaviour is discussed further in section 8. In the next section, we confirm that the population is not moving in any useful manner over the neutral epoch, by reseeding five populations from across the neutral epoch back into the genetic algorithm, and recording how many generations were required to

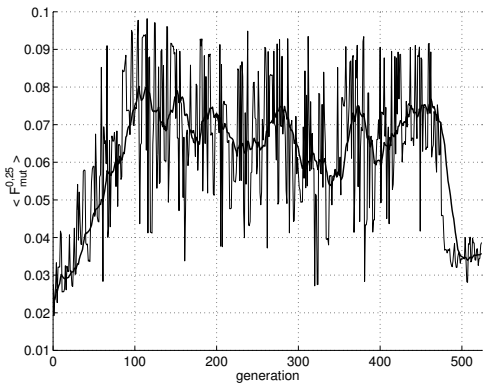


(a) Probability of mutation fitter than current fitness, E_a

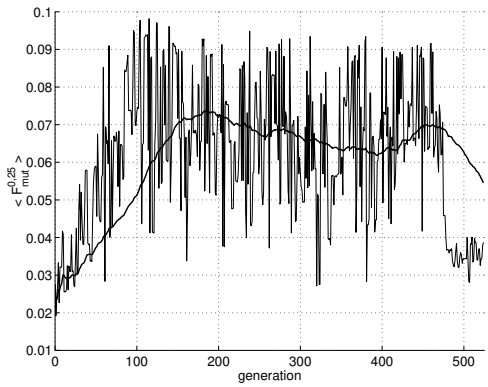


(b) Expected fitness of top quartile of mutations, E_c

Figure 13: Evolvability metrics plotted over generations. **(a)** The probability of a fitter mutation than the current individual, and **(b)** The expected value over the top quartile of transmission function fitnesses. Rolling mean over 30 generations have been added to aid clarity.



(a) Short term trend in expected fitness of bottom quartile of mutations, E_d



(b) Long term trend in expected fitness of bottom quartile of mutations, E_d

Figure 14: The expected fitness of the bottom quartile of mutations (E_d) plotted over generations. Rolling means have been added to show the short- and long- term trends: **(a)** Rolling mean over last 20 generations shown, and **(b)** Rolling mean over the last 100 generations shown.

evolve successful controllers.

6.4 Does evolvability predict speed of evolution?

The evolvability results from section 6.3 predict that there is no change in potential for evolution of high fitness during the neutral epoch between generations 100-477. In particular, the population shows no evidence of having moved towards areas with either increased probability of producing better mutations, or decreased probability of producing bad mutations. The second

experiment empirically tests this prediction, repeating the evolution from different points along the neutral epoch. Five populations, from generations $\{100, 200, 300, 400, 477\}$, were used as the initial populations for the genetic algorithm, and the evolutionary process repeated five times for each population. Table 2 shows the number of generations required for 100% success on each of the evolutionary runs, while figure 15 shows the median number of generations taken to reach certain fitnesses. Statistical analysis shows no significant differences between the five sets of runs, supporting the hypothesis that there is no change in the evolutionary potential, or evolvability, of the population across the neutral epoch.

	Pop 100	Pop 200	Pop 300	Pop 400	Pop 477
Mean	2,008	2,096	1,901	1,680	3,024
Median	1,522	1,464	932	1,093	1,597
Maximum	4,713	7,696	>10,000	5,707	>10,000
Minimum	353	365	107	353	290

Table 2: Statistics on the number of generations required before the GA reaches 100% success, starting from the 5 populations saved on generations $\{100, 200, 300, 400, 477\}$. Note: Runs not reaching success in $10,000$ generations were counted as $10,000$ for averaging purposes. No significant differences were seen between the populations (Kruskal-Wallis analysis).

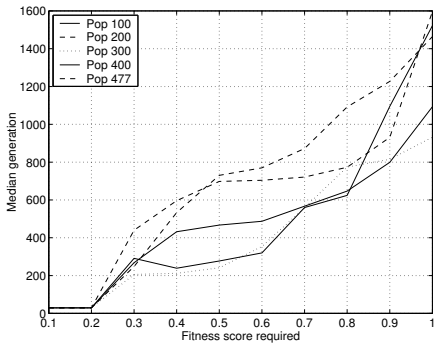


Figure 15: Median from 5 runs of the number of generations required before the GA reaches a given fitness level, starting from the 5 populations saved on generations $\{100, 200, 300, 400, 477\}$. No significant differences were seen between the populations (Kruskal-Wallis analysis).

6.5 Summary of evolutionary run analysis

Over a sample of twenty evolutionary runs, we see the same general trends as seen for this single run. First, there is no evidence for change in evolvability across neutral epochs (although it is possible that detailed mathematical time-series analysis would show some small trend), either in the probability of obtaining fitter mutations, or in the best or expected mutation fitness, or in the expected fitness over the top or bottom quartiles of the mutations. Thus we see no trend towards fewer bad mutations or more good mutations. Second, evolvability does change significantly during epochs of fitness increase, when measured in terms of the highest and expected fitnesses of the transmission function. In general, evolvability increases during these periods, although some evidence is seen for an *increase* in bad mutations during fitness-increase, corresponding to the best individual solution reaching a fitter area of space, but with many more deleterious mutations nearby. However, we should mention that on three of the twenty runs, we saw possible indications of evolvability increasing over a neutral epoch, as measured through expected fitness and probability of obtaining higher fitness. This is discussed further in section

8. The next section goes on to investigate the evolvability of two different search spaces defined by two neural network architectures.

7 GasNet and NoGas evolvability

In this section, we apply the evolvability analysis defined in section 3.2 to the GasNet and NoGas neural network architectures, to try to explain the observed differences in speed of evolution, in terms of the underlying search spaces.

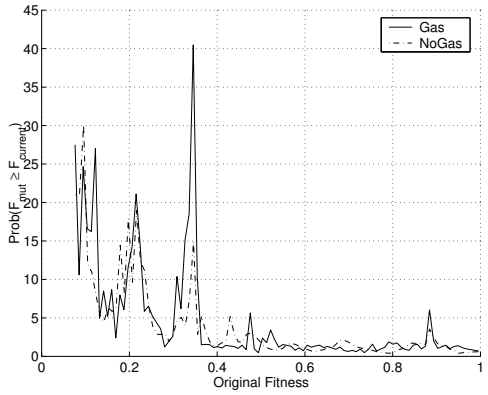
A sample of twenty evolutionary runs, ten GasNet and ten NoGas, were used to provide an online sample of the two search spaces (online in the sense that the samples were collected during a search process, rather than collected through random sampling). The best individual of each generation on every run was saved (roughly $80,000$ solutions in all), and the transmission function distribution of offspring fitnesses calculated through $1,000$ applications of the mutation operator. Figure 16 shows the four evolvability metrics calculated over the transmission functions for the entire online sample, with evolvability calculated for the population at each sample fitness, and plotted against the sample fitness (section 3.2 describes how the population evolvability is defined as the evolvability over the sum of the individual transmission functions). Several points emerge:

First, the probability of obtaining a fitter mutation (figure 16(a)) decreases from an initial value of around 15%, eventually reaching a constant value of roughly 2% at a fitness level just above 0.4. These values will be extremely affected by the noise inherent in fitness evaluation, and the constant value of 2% is similar to the probability that an equal fitness solution has a higher fitness than the current evaluation (already evaluated as the best of generation). No obvious difference is seen between the two conditions.

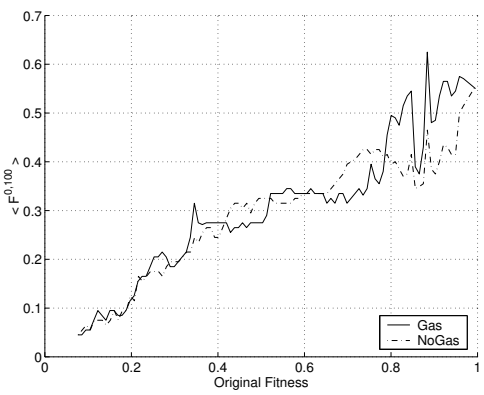
Second, the expected fitness over all mutations increases roughly linearly as a function of the current fitness (figure 16(b)). Again no obvious differences are seen between the two spaces, although there is a slight suggestion that the expected fitness in the GasNet space increases slightly faster above fitnesses of 0.7.

Third, the expected fitness of the top 50% of mutations (figure 16(c)) increases roughly linearly with fitness. Again no differences are seen between the two spaces.

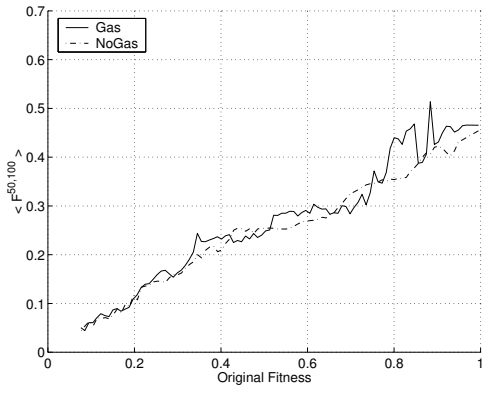
Fourth, the expected fitness of the bottom 50% of mutations (figure 16(d)) shows a clear difference between the two spaces. In general the expected fitness increase with fitness, but the NoGas space shows a decrease at a fitness of roughly 0.5. This difference is also seen in the expected value over the bottom quartile of mutations, and for the middle 50% and 33% of mutations. Although the fitness differences are relatively small, there is a consistent trend: at higher fitnesses, the GasNet search space shows fewer bad mutations than the NoGas space. This fits with previous results showing that the evolutionary speed differences only emerge at high fitnesses, see figure 17. It should be emphasised that the number of online samples collected and tested (over $80,000$ solutions) make this result extremely unlikely through chance. All statistical analyses show significant differences, as would be expected for such a large sample size.



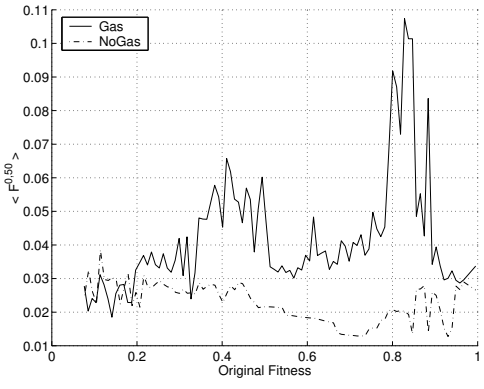
(a) Probability of mutation fitter than current fitness, E_a



(b) Expected fitness over all mutations, E_b



(c) Expected fitness over top 50% of mutations, E_c



(d) Expected fitness over bottom 50% of mutations, E_d

Figure 16: The metrics of evolvability derived in section 3.1 applied to the search spaces defined by the GasNet and NoGas neural network architectures. Twenty evolutionary runs were used to provide an online sample of the spaces, and the transmission distribution of 1-step mutation offspring approximated through 1,000 applications of the mutation operators to the collected sample. These distributions were then summed up for all sample solutions of equal fitness (more accurately, all samples lying within a certain fitness range), and evolvability calculated over each summed distribution. Thus each point represents an evolvability metric calculated over a sample of solutions at that fitness in the search space. Rolling means over the 20 neighbouring fitness ranges have been added to aid clarity. See text for further details.

7.1 Summary of GasNet and NoGas evolvability analysis

We have applied a set of evolvability metrics to samples of solutions from the search spaces defined by two different neural network architectures, GasNet and NoGas. Clear differences in the evolvability of the two network representations are seen in the higher expected fitnesses for the worst mutations, see figure 16(d). Further analysis (figure not shown) shows that this higher expected fitness is due to a larger fraction of neutral mutations and a smaller fraction of

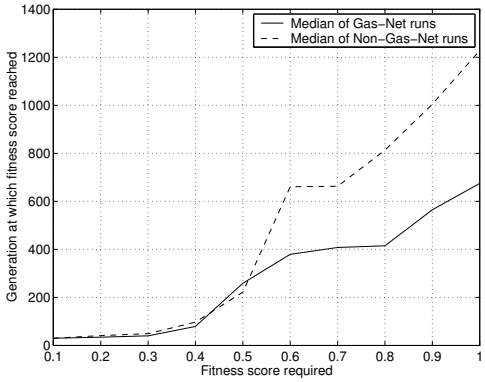


Figure 17: Median generations before given fitness level reached, for GasNet and NoGas conditions.

extremely deleterious mutations from the GasNet solutions, rather than an increased number of intermediate mutations. By contrast, the probability of mutations which increase fitness, and the overall expected fitness of mutations were extremely similar for the two search spaces (the differences in the expected fitness of the worst mutations has only a relatively small impact on the overall expected fitness).

Although the differences are small, the larger number of neutral mutations in the GasNet space, especially at higher fitnesses, will result in a larger number and variety of good solutions in the population at any one time. The probability of a deleterious mutation being subsequently reversed is small[2], thus at any one time, evolution in the GasNet space has a greater probability of finding the good mutations: both the increased number of good solutions, and the increased variety will aid evolution.

8 Discussion

The promise of artificial evolution lies in producing good solutions to difficult problems. However, if we are to apply such techniques to evolving solutions for problems with time consuming fitness evaluation, we must develop suitably evolvable solution architectures. In recent work we have developed the "GasNet", and shown in a series of evolutionary robotics experiments that a significant decrease in the numbers of evaluations is required to evolve successful GasNet controllers (Husbands, 1998; Husbands et al., 1998; Smith and Philippides, 2000). In this paper we have applied the concept of evolvability in order to analyse the search spaces underlying two different genetic systems, the GasNet and NoGas robot control architectures.

We have shown how a set of evolvability metrics defined in terms of the transmission function, or the distribution of solution offspring fitnesses, can predict the difficulty of finding good solutions in a theoretical class of tunably rugged and neutral landscapes (Kauffman, 1993; Newman and Engelhardt, 1998). In particular, we have shown that the metrics can predict the difficulty of finding solutions of a given fitness, not possible with single measures of landscape "difficulty" such as the correlation length.

[2]We can apply Muller's ratchet argument: the probability of a low fitness solution being selected as a parent is small, and the probability that the resulting offspring will have high fitness is also small.

In a set of experiments we then applied the evolvability metrics to the two search spaces defined by the GasNet and NoGas control architectures, in order to investigate the behaviour of evolvability during neutral fitness epochs (identified through analysis of the population fitness and movement through genotype space), and periods where fitness increased. The majority of runs showed no evidence for any general trend towards increase in evolvability across neutral epochs, although three of the twenty runs analysed showed some such increase. By contrast, there was certainly a general trend for increase in evolvability during periods in which fitness increased. However, this was not the whole story: occasionally evolvability was seen to *decrease* during periods of fitness increase, e.g. see figure 14. Such significant decreases in evolvability during a rise in fitness represent the discovery of a portal mutation to a neutral network of higher fitness, but where the error threshold for deleterious mutations is more severe (Nowak and Schuster, 1989). Using an evolutionary approach in which the elite solutions are not necessarily propagated to the next generation might well result in losing these high fitness genotypes.

Finally, we applied the metrics of evolvability to the GasNet and NoGas search spaces in order to highlight the differences leading to the significantly faster evolutionary search with the GasNet architecture. Possibly surprisingly, no differences were seen for the probability of mutations increasing fitness, or for the expected fitness over all mutations, or for the expected fitness over the best mutations. However, significant differences were seen in the expected fitnesses over the worst mutations: at higher fitnesses, the GasNet solutions produced less deleterious and more neutral mutations than the NoGas solutions.

It should be emphasised that this result is not contradicted by the GasNet and NoGas evolutionary runs showing the same population mean fitness for a given best fitness (as is indeed the case), for two reasons. First, as described in section 7, the differences in the expected fitness of the worst mutations has only a relatively small impact on the overall expected fitness, so the population fitnesses will not be expected to be very different, and the effect may well be masked by the inherent evaluation noise. Second, the population may be spread over larger or smaller volumes of the search space at different times, and may be centred on more than one dominant genotype. By contrast, the distribution of offspring investigated in this paper is by definition the distribution of 1-step mutants from the parent, so the evolvability metrics describe a constant volume of space defined by the genetic operators.

It is unclear whether the differences in the numbers of non-deleterious mutations between the GasNet and NoGas search spaces provide the whole picture. The significant differences in the search spaces are small, and may not account fully for the observed differences in the time required for evolving good solutions. Although the increased number and variety of good genotypes in the population will aid evolution, it may be that there is another factor affecting evolutionary speed that remains unidentified. Further research will focus on whether the increase in variety of good genotypes fully accounts for the speed of evolution.

Acknowledgements

The authors would like to thank Andy Philippides, Anil Seth, Inman Harvey, Lionel Barnett, John Welch and all the members of the Centre for Computational Neuroscience and Robotics (http://www.cogs.sussex.ac.uk/ccnr/) for constructive discussion. We would also like to thank the Sussex High Performance Computing Initiative (http://www.hpc.sussex.ac.uk/) for computing support. TS is

funded by a British Telecom sponsored Biotechnology and Biology Science Research Council Case award.

References

Altenberg, L. (1994). The evolution of evolvability in genetic programming. In Kinnear Jr, K., editor, *Advances in Genetic Programming*, chapter 3, pages 47–74. MIT Press.

Barnett, L. (1998). Ruggedness and neutrality: The NKp family of fitness landscapes. In Adami, C., Belew, R., Kitano, H., and Taylor, C., editors, *Artificial Life VI: Proceedings of the Sixth International Conference on Artificial Life*, pages 18–27. MIT Press / Bradford Books.

Burch, C. and Chao, L. (2000). Evolvability of an RNA virus is determined by its mutational neighbourhood. *Nature*, 406:625–628.

Cavalli-Sforza, L. and Feldman, M. (1976). Evolution of continuous variation: Direct approach through joint distribution of genotypes and phenotypes. *Proceedings of the National Academy of Sciences, USA*, 73:1689–1692.

Cliff, D. T., Harvey, I., and Husbands, P. (1993). Explorations in evolutionary robotics. *Adaptive Behaviour*, 2(1):71–104.

Coyne, J., Barton, N., and Turelli, M. (1997). Perspective: A critique of Sewall Wright's shifting balance theory of evolution. *Evolution*, 51(3):643–671.

Dawkins, R. (1989). The evolution of evolvability. In Langton, C., editor, *Artificial Life: Proceedings of the Interdisciplinary Workshop on the Synthesis and Simulation of Living Systems*, volume VI of *Santa Fe Institute Studies in the Sciences of Complexity*, pages 201–220. Addison-Wesley, Redwood, CA.

Ebner, M., Langguth, P., Albert, J., Shackleton, M., and Shipman, R. (2001). On neutral networks and evolvability. In *Proceedings of the 2001 Congress on Evolutionary Computation: CEC2001*, pages 1–8. IEEE, Seoul, Korea.

Eldredge, N. and Gould, S. (1972). Punctuated equilibria: An alternative to phyletic gradualism. In Schopf, T., editor, *Models in Paleobiology*, pages 82–115. Freeman, San Francisco.

Elena, S., Cooper, V., and Lenski, R. (1996). Punctuated evolution caused by selection of rare beneficial mutations. *Science*, 272:1802–1804.

Floreano, D. and Mondada, F. (1994). Automatic creation of an autonomous agent: Genetic evolution of a neural-network driven robot. In Cliff, D., Husbands, P., Meyer, J.-A., and Wilson, S., editors, *From Animals to Animats 3: Proceedings of the Third International Conference on Simulation of Adaptive Behaviour, SAB94*. MIT Press / Bradford Books.

Fogel, L., Owens, A., and Walsh, M. (1966). *Artificial Intelligence through Simulated Evolution*. John Wiley, New York.

Gould, S. and Eldredge, N. (1977). Punctuated equilibria: The tempo and mode of evolution reconsidered. *Paleobiology*, 3:115–151.

Harvey, I. and Thompson, A. (1996). Through the labyrinth evolution finds a way: A silicon ridge. In Higuchi, T., editor, *Proceedings of the First International Conference on Evolvable Systems: From Biology to Hardware (ICES'96)*, pages 406–422. Springer-Verlag.

Holland, J. (1992). *Adaptation in Natural and Artificial Systems*. MIT Press / Bradford Books, 2nd edition.

Hordijk, W. (1996). A measure of landscapes. *Evolutionary Computation*, 4(4):335–360.

Husbands, P. (1998). Evolving robot behaviours with diffusing gas networks. In Husbands and Meyer (1998), pages 71–86.

Husbands, P. and Meyer, J.-A., editors (1998). *Evolutionary Robotics: First European Workshop, EvoRobot98*. Springer-Verlag, Berlin.

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

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

Jones, T. and Forrest, S. (1995). Fitness distance correlation as a measure of problem difficulty for genetic algorithms. In Eshelmann, L., editor, *Proceedings of the Sixth International Conference on Genetic Algorithms (ICGA95)*, pages 184–192. Morgan Kaufmann, CA.

Kallel, L., Naudts, B., and Reeves, C. (2000). Properties of fitness functions and search landscapes.

Kauffman, S. (1993). *The Origins of Order: Self-Organization and Selection in Evolution*. Oxford University Press.

Kimura, M. (1983). *The Neutral Theory of Molecular Evolution*. Cambridge University Press.

Kirschner, M. and Gerhart, J. (1998). Evolvability. *Proceedings of the National Academy of Sciences, USA*, 95:8420–8427.

Koza, J. (1992). *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press / Bradford Books.

Lawler, E., Lenstra, J., Rinnooy Kan, A., and Schmoys, D. (1985). *The Travelling Salesman Problem*. John Wiley, New York.

Marrow, P. (1999). Evolvability: Evolution, computation, biology. In Wu, A., editor, *Proceedings of the 1999 Genetic and Evolutionary Computation Conference Workshop Program (GECCO-99 Workshop on Evolvability)*, pages 30–33.

Naudts, B. and Kallel, L. (2000). A comparison of predictive measures of problem difficulty in evolutionary algorithms. *IEEE Transactions on Evolutionary Computation*, 4(1):1–15.

Newman, M. and Engelhardt, R. (1998). Effects of selective neutrality on the evolution of molecular species. *Proceedings of the Royal Society of London, B*, 265:1333–1338.

Nolfi, S. and Floreano, D. (2000). *Evolutionary Robotics: The Biology, Intelligence and Technology of Self-Organizing Machines*. MIT Press / Bradford Books.

Nowak, M. and Schuster, P. (1989). Error thresholds of replication in finite populations - mutation frequencies and the onset of Muller's ratchet. *Journal of Theoretical Biology*, 137:375–395.

Partridge, L. and Barton, N. (2000). Evolving evolvability. *Nature*, 407:457–458.

Philippides, A., Husbands, P., and O'Shea, M. (2000). Four-dimensional neuronal signaling by nitric oxide: A computational analysis. *Journal of Neuroscience*, 20(3):1199–1207.

Provine, W. (1986). *Sewall Wright and Evolutionary Biology*. Chicago Press.

Rechenberg, I. (1973). *Evolutionsstrategie*. Friedrich Frommann Verlag, Stuttgart.

Shackleton, M., Shipman, R., and Ebner, M. (2000). An investigation of redundant genotype-phenotype mappings and their role in evolutionary search. In *Proceedings of the 2000 Congress on Evolutionary Computation: CEC2000*, pages 493–500. IEEE, San Diego, USA.

Smith, T., Husbands, P., and O'Shea, M. (2001a). *Not* measuring evolvability: Initial exploration of an evolutionary robotics search space. In *Proceedings of the 2001 Congress on Evolutionary Computation: CEC2001*, pages 9–16. IEEE, Seoul, Korea.

Smith, T., Husbands, P., and O'Shea, M. (2001b). An evolvability measure of landscapes. In preparation.

Smith, T. and Philippides, A. (2000). Nitric oxide signalling in real and artificial neural networks. *BT Technology Journal*, 18(4):140–149.

Sniegowski, P., Gerrish, P., and Lenski, R. (1997). Evolution of high mutation rates in experimental populations of *E. coli*. *Nature*, 387:703–705.

Stadler, P. (1996). Landscapes and their correlation functions. *Journal of Mathematical Chemistry*, 20:1–45.

Taddei, F., Radman, M., Maynard-Smith, J., Toupance, B., Gouyon, P., and Godelle, B. (1997). Role of mutator alleles in adaptive evolution. *Nature*, 387:700–702.

Thompson, A. (2001). Neutrality in evolutionary hardware experiments. In preparation.

Turney, P. (1999). Increasing evolvability considered as a large-scale trend in evolution. In Wu, A., editor, *Proceedings of the 1999 Genetic and Evolutionary Computation Conference Workshop Program (GECCO-99 Workshop on Evolvability)*, pages 43–46.

van Nimwegen, E., Crutchfield, J., and Huynen, M. (1999). Neutral evolution of mutational robustness. *Proceedings of the National Academy of Sciences, USA*, 96:9716–9720.

Vassilev, V., Fogarty, T., and Miller, J. (2000). Information characteristics and the structure of landscapes. *Evolutionary Computation*, 8(1):31–60.

Vassilev, V. and Miller, J. (2000). The advantages of landscape neutrality in digital circuit evolution. In Miller, J., Thompson, A., Thomson, P., and T., F., editors, *Proceedings of the Third International Conference on Evolvable Systems: From Biology to Hardware (ICES'2000)*, volume 1801 of *Lecture Notes in Computer Science*, pages 252–263. Springer-Verlag.

Wagner, G. and Altenberg, L. (1996). Complex adaptations and the evolution of evolvability. *Evolution*, 50(3):967–976.

Weinberger, E. (1990). Correlated and uncorrelated fitness landscapes and how to tell the difference. *Biological Cybernetics*, 63:325–336.

Weinberger, E. (1991). Local properties of Kauffman's N-K model: A tunably rugged energy landscape. *Physical Review A*, 44(10):6399–6413.

Wilke, C. (2001). Adaptive evolution on neutral networks. *Bulletin of Mathematical Biology*. Submitted.

Wilke, C., Wang, J., Ofria, C., Lenski, R., and Adami, C. (2001). Evolution of digital organisms lead to survival of the flattest. *Nature*. Submitted.

Wright, S. (1932). The role of mutation, inbreeding, crossbreeding and selection in evolution. In Jones, D., editor, *Proceedings of the Sixth International Congress on Genetics*, volume 1, pages 356–366. Reprinted in Ridley, M. (1997). *Evolution*. pp32-40. Oxford University Press.