16

Analysing the Difficulty of Learning Goal-Scoring Behaviour for Robot Soccer

Jeff Riley and Vic Ciesielski
RMIT University
Australia

1. Introduction

This work describes a method of analysing fitness landscapes and uses the method to analyse the difficulty of learning goal-scoring behaviour for robot soccer – a problem that is considered to be very difficult for evolutionary algorithms. Learning goal-scoring behaviour can be made easier or harder by varying the amount of expert knowledge provided to the evolutionary process. Expert knowledge can be varied by changing the innate player behaviours available, providing an *a priori* problem decomposition (that is, breaking the problem into a series of smaller, easier problems) or by providing a composite fitness function that effectively guides the search.

The concept of fitness landscapes, and the idea that the process of evolution could be studied by visualizing the distribution of fitness values across the population as a landscape, has been long-established in the field of evolutionary biology, having been first proposed by Sewell Wright (Wright, 1932). Later the landscape analogy was revived with the development of formal methods to handle optimization problems in complex physical systems (Frauenfelder et al., 1997). A major area of concern with fitness landscapes is that there is no generally accepted definition of what constitutes a fitness landscape. There is not much agreement in the field as to what a fitness landscape is and how it should be arranged - whether a neighbourhood relation is required to describe it, and much less agreement as to what the neighbourhood relation should be. This work addresses these shortcomings by describing a simple, "black-box" neigbourhood relation that defines the fitness landscape generated by an evolutionary search. The efficacy of the method is shown by applying an evolutionary technique to a difficult search problem (learning goal-scoring behaviour), and using *autocorrelation* and *information content* landscape measures to analyse features of the resultant fitness landscape to explain how the difficulty of the problem is changed by injecting human expertise. The analysis reveals that when only basic skills are available to the player the fitness landscape is very flat and contains only a few thin peaks. As more human expertise is injected, more gradient information becomes apparent on the landscape and genetic search is more successful.

Evolving soccer-playing skills for robot soccer players is a well-known difficult problem for evolutionary algorithms. A wide variety of approaches and technologies have been used in attempts to construct good robot soccer players. These include hand-coding, genetic

Source: Robotic Soccer, Book edited by: Pedro Lima, ISBN 978-3-902613-21-9, pp. 598, December 2007, Itech Education and Publishing, Vienna, Austria

algorithms, genetic programming, reinforcement learning, neural networks, behaviour-based and deliberative agents, and various combinations of those. In the early years of the RoboCupSoccer competition (Kitano et al., 1995) a few researchers attempted to fine tune, or learn incrementally, some low-level skills using AI machine learning techniques (Stone, 1998; Stone & Sutton, 2001), but nearly all entrants used hand-coded skills and strategies (Luke, 1998). Even today hand-coded players, or players with hand-coded skills, generally continue to outplay players whose skills have been entirely learned or developed automatically (Lima et al., 2005). For example, the 2005 RoboCupSoccer 2D simulation league winner used hand-coded strategies which employed a mixture of hand-coded skills and skills developed using machine learning techniques (Riedmiller et al., 2005). There has been only limited success when applying standard machine learning techniques to this problem. Much of the work to date has been characterised by researchers beginning their work with high expectations, then ratcheting down their expectations as the work progresses, and finally adjusting their goals (and the soccer playing behaviours and skills of the players being developed) to align with the progress being made.

The size of the robot soccer search space and the paucity of solutions that lead to good soccer-playing behaviour tend to suggest that, in the extreme case, the problem resembles a *needle-in-a-haystack* problem (Jones & Forrest, 1995a; Langdon & Poli, 1998a; Right et al., 2002), indicating a possible cause for the difficulty of the problem for evolutionary algorithms. Previous work in the area of evolutionary learning for robot soccer has focussed on the learning, and what parameters and controls need to be manipulated in order to produce acceptable soccer-playing behaviour, but there has been no systematic investigation of the difficulty of the problem. Understanding why problems are difficult for evolutionary algorithms is a critical step not only in solving the particular problem under investigation, but also in advancing the field and improving the potential usefulness of evolutionary algorithms. The goal of this work is to describe a method for analysis of the fitness landscape generated by the problem of learning goal-scoring behaviour for robot soccer, and to use the analysis to better understand the difficulty of the problem and how progress might be made.

2. Evolving Goal-Scoring Behaviour for Robot Soccer

For this work a messy-coded genetic algorithm (Holland, 1975; Goldberg et al., 1989) is used to evolve a single robot soccer player in the SimpleSoccer simulated soccer environment (Riley, 2003). The behaviour of the player is governed by a fuzzy inferencing system (Zadeh, 1965; Jang et al., 1997) with the ruleset for the fuzzy inferencing system being evolved by the genetic algorithm.

The player being evolved is endowed with a configurable subset of soccer-playing skills taken from a rich set of basic and hand-coded skills and default actions. The player executes one of the skills or performs the default action available to it in response to some external stimulus; the specific response being determined by the fuzzy ruleset and the fuzzy inferencing system. The external stimulus used as input to the fuzzy inference system is the visual information supplied by the soccer simulator.

A full description of the method used to evolve the soccer player is given in the previous chapter.

3. Fitness Landscapes

Wright's original diagram, reproduced in Fig. 1, is a two-dimensional depiction of the relative "adaptiveness" of all possible combinations of allelic states for genes on a particular chromosome. The diagram is effectively a contour map with the contour lines representing levels of adaptiveness. This original diagram included no labels for the axes, and no indication was given as to how the various gene combinations should be arranged on the landscape - no notion of "neighbourhood" was defined or discussed by Wright. In landscape terms, the neighbourhood relation defines which points, or individuals, are arranged as immediate neighbours on the landscape, and so is extremely important in defining the landscape.

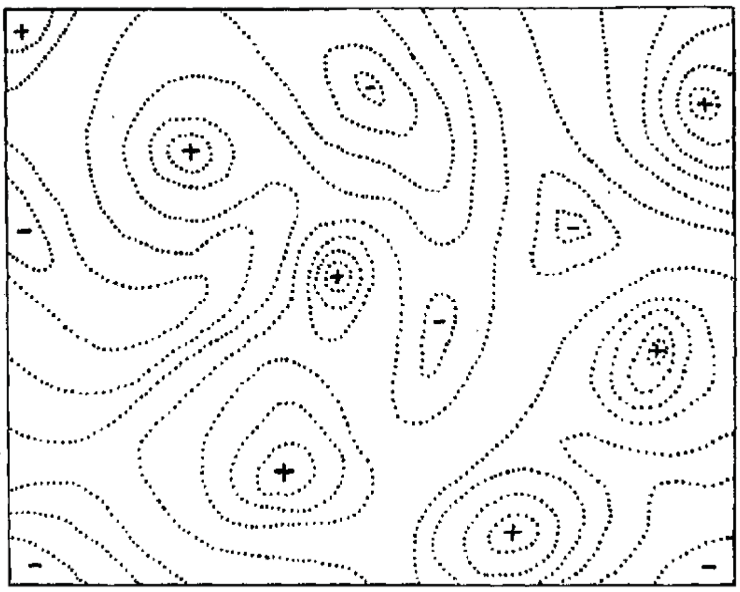

Fig. 1. Diagrammatic representation of the field of gene combinations in two dimensions instead of many thousands. Dotted lines represent contours with respect to adaptiveness. *Reproduced from (Wright, 1932)*

Much of the work involving fitness landscapes avoids a rigorous definition of the landscape under analysis (Jones, 1995a), and where it is mentioned or implied at all the landscape is usually assumed to be the single-bit mutation landscape: the landscape generated by arranging all single-bit mutations of a chromosome represented as a string of binary digits such that chromosomes that differ by only a single bit are neighbours, then plotting the fitness of each chromosome on a separate axis. On such landscapes, genetic operators such as crossover are assumed to take hypersteps over the fitness landscape described by mutation. There have been attempts to overcome this deficiency, and following is a description of some of major recent work in the field.

The NK fitness landscape model (Kauffman, 1989; Kauffman & Levin, 1987) was proposed as a means to study how epistasis - the dependency of fitness upon the interaction of the allelic state of multiple genes (Lush, 1935) - affects the ruggedness of the fitness landscape. An NK fitness landscape is defined by a fitness function which can be tuned in order to alter the ruggedness of the resultant fitness landscape. The fitness function is defined by two parameters: the number of genes (N), and the number of epistatic links, or interactions, between genes (K). In the most common implementations each gene has two possible alleles, thus the genotype can be represented by a bit string of length N. Each gene in a chromosome contributes to the fitness of the chromosome based on K+1 values: its own and those of the K genes to which it is linked. The three-dimensional fitness landscape is constructed by arranging chromosomes in two dimensions on the landscape such that bit strings that differ in the value of only one bit are neighbours, then using the fitness of the chromosomes as the third dimension. The NK landscape is widely used by the evolutionary computation community to generate epistatic landscapes as test functions for search and optimisation techniques (De Jong et al., 1997; Heckendorn et al., 1999; Skellett et al., 2005; Smith & Smith, 2001).

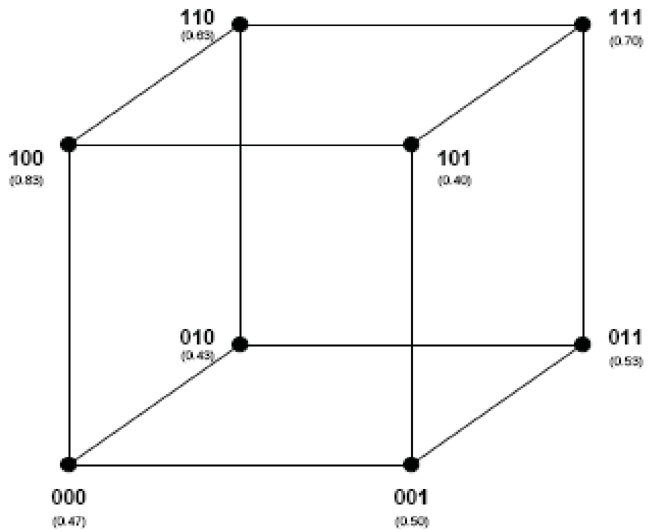

Fig. 2. An example of a fitness landscape for bit strings of length three, where the neighbourhood relation is defined by point mutation. Fitness values (randomly assigned) are shown in parentheses. *Reproduced from (Hordijk, 1996)*

Weinberger (1990a; 1991a) proposed a fitness landscape model in which the landscape is represented as a graph on which the vertices correspond to individuals and have associated fitness values, and traversing the edge of the graph corresponds to the action of a genetic operator (mutation, crossover etc.) and so taking a step on the landscape. Jones (1995a; 1995b), Culberson (1994), and later Hordijk (1996) describe similar fitness landscape models in which the landscape is represented as a graph (e.g. Fig. 2). Reidys and Stadler (2002) analyse a similar fitness landscape topography, focussing on the geometry of the moves

from one vertex to another and provide a mathematical treatment of the edge traversals and the resultant complex topographies.
Culberson's model described a landscape defined by a crossover operator rather than mutation, and in which the graph vertices represented a population of points. Jones presents a similar model, and expands it further to include the concept that each genetic operator defines its own separate landscape (Jones 1995a). In Jones' "one operator, one landscape" model, an evolutionary algorithm which implements the genetic operators selection, mutation and crossover makes transitions on three separate landscapes (Fig. 3). According to Jones' model the evolutionary algorithm takes steps on the mutation landscape, after which individuals are paired to form vertices on the crossover landscape and further steps are then taken on that landscape, and then the population is gathered into a vertex on the selection landscape for a further step there. The neighbourhood relation in Jones' model is therefore simply defined by the genetic operator for each landscape.

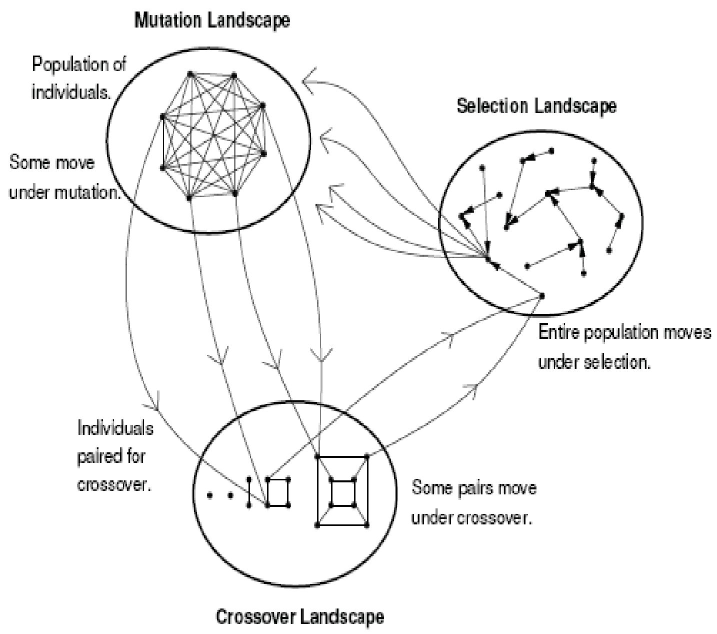

Fig. 3. A simplified view of an evolutionary algorithm operating on three landscapes.
Reproduced from (Jones 1995a)

More recently Moraglio and Poli (2004) presented a new topological framework for evolutionary algorithms and as part of that framework redefine the mutation and crossover operators to be more tightly linked to the fitness landscape. In the model proposed by Moraglio and Poli the genetic operators are defined by the fitness landscape upon which they operate – the genetic operators are a natural consequence of the neighbourhood relation and distance metric of the fitness landscape. For example, under what Moraglio and

Poli call topological uniform crossover, offspring are defined as those individuals that lie between the parents on the fitness landscape. Similarly, under topological ε-mutation, offspring are defined as those individuals that lie ε steps away from the parent on the fitness landscape.

In essence, the fitness landscape is a metaphor – a metaphor in which the landscape is often thought of as a 3-dimensional terrain, where the peaks (hills or mountains) represent areas of high fitness and the valleys between the peaks low fitness. A fitness landscape which depicts the fitness values of a population that varies greatly in fitness will likely display many local high peaks surrounded by deep valleys. Such a landscape is said to be rugged. Similarly, if many individuals in the population have a similar fitness, the landscape will vary little and is said to be flat. Search on some landscapes is notionally easier than search on others – search on a predominantly flat landscape is likely to be difficult, as is search on a rugged landscape with peaks and valleys randomly distributed.

In evolutionary biology the proliferation of an individual's genetic material is considered the ultimate objective of life, and the success or failure of a particular genotype, or its phenotypic expression, is most often measured by the number of progeny it produces – so for evolutionary biology, the fitness of an individual is a function of the number of progeny produced by that individual. For evolutionary optimisation the objective is usually less nebulous, and the success or failure of a particular genotype, or phenotype, is measured by a well-defined fitness function, and typically the number of progeny produced by an individual is a function of the fitness of that individual. Fitness landscapes are used in evolutionary biology and evolutionary optimisation to show the correlation between genotypes, or phenotypes, as a measure of success or failure and search difficulty (Jones & Forrest, 1995b; Kauffman, 1989; Kauffman & Levin, 1987; Langdon & Poli, 1998b; Vassilev, 1997).

4. Which Fitness Landscape?

As discussed in the previous section there are several possible definitions of, and representations for, fitness landscapes, and choosing the definition and representation which best describes the combination of the problem being studied and the algorithm being used to study it is extremely important.

A fitness landscape is most often defined by three basic attributes:

- a search space
- a relation that defines which points are neighbours in the search space
- a fitness function that assigns a fitness value to each point in the search space

A significant problem with this definition of fitness landscapes is the specification of the neighbourhood relation. The neighbourhood relation and its specification is extremely important because any discussion of landscapes invariably involves the terms "peaks" and "valleys", and no peak or valley can exist without the notion of neighbourhood – a peak is only a peak because it is higher than its neighbours.

Often the neighbourhood relation is defined in simple terms, such as the hamming distance for bit strings, or by defining all single bit mutations of a bit string as neighbours. This potentially ignores an important ingredient in the evolutionary processes: evolutionary algorithms are usually governed by some combination of several operators. A definition of the neighbourhood relation in terms of the actual mix of genetic operators (for example, selection, mutation and crossover) being used by the algorithm would seem to be more relevant and useful to an analysis of the performance of the algorithm. Since the search is for individuals of good fitness, the individuals which comprise the search space and their associated fitness values should ideally be arranged as a fitness landscape which has individuals of better fitness clustered together so that the landscape contains peaks and valleys representing the fitness extremes. The issue is how to represent the fitness landscape to achieve that, and a critical question is: what attributes of an individual should determine the locality of that individual on the fitness landscape?

If the neighbourhood relation is not defined correctly in terms of describing the actions of the algorithm being used to perform the search, then the fitness landscape is not the landscape searched by the search algorithm, and so not necessarily indicative of the difficulty of the search. On a physical landscape points are neighbours, or next to each other, because (for example) a person walking on that landscape can traverse the distance between those points in a single step. The corollary is that if a searcher walking on a landscape is not able to traverse the distance between two points in a single step, then those points are not next to each other and so, by definition, are not neighbours on the landscape defined by that searcher and the associated granularity of search (a single step). Simply plotting attributes of the members of the search space against fitness – for example chromosome length in the case of a genetic algorithm, or program length for genetic programming – while possibly useful for visualising the fitness distribution of the search space, may not be sufficient to describe the fitness landscape traversed by the search algorithm, since there may be no reason to expect that the search algorithm could traverse the distance between neighbouring points on that landscape with a single step.

As discussed earlier, a fitness landscape is a metaphor and an aid to the visualisation of the operation of a search algorithm, but for anything other than the actual landscape traversed by the search algorithm the metaphor is flawed and the visualisation is misleading. For example, the analogy of the search algorithm to a short-sighted explorer wandering over a (possibly rough) terrain (Langdon & Poli, 2002) is only valid for the actual fitness landscape searched by the algorithm, and anything else gives a misleading view of the complexity, or ruggedness, of the landscape. For the explorer analogy to be useful, the neighbourhood relation must reflect the notion that an individual's neighbour on the fitness landscape is any individual in the search space which can be reached in a single step of the explorer. In the case of an evolutionary algorithm, the explorer is the algorithm, and the single step is the combination of whichever genetic operators are implemented by that algorithm: selection, mutation and crossover. It should also be remembered that during the search the explorer may not be able to see all its theoretical neighbours. This is because, for an evolutionary algorithm using crossover, the individuals reachable in a single step from one parent depend upon the other individuals selected for mating, and while an individual can theoretically mate with any other individual in the search space, in reality the individuals available for selection are restricted by the size of the population. So to continue the analogy of the short-sighted explorer, not only is the explorer short-sighted, but also lacks peripheral vision.

While it should be remembered that the 3-dimensional fitness landscape is just a metaphor, since in most cases the surface being traversed by the searcher will be multi-dimensional and so complex that there will be no landscape that could actually be visualised, the analogy of the searcher on a 3-dimensional landscape is a useful aid to imagining how a search algorithm might use the available attributes of the search space.

It is important to understand that the (metaphoric) fitness landscape is defined by a number of attributes, which include the search algorithm, and not solely by the specification of the problem. It is not valid to visualise a fitness landscape without reference to a search algorithm, and then decide what algorithm will best search the landscape – the operation of the search algorithm defines the neighbourhood relation, and so the shape, of the landscape.

Previous work with fitness landscapes recognises the need to refer to the search algorithm when characterising the landscape (Hordijk, 1996; Jones & Forrest, 1995b; Reeves, 1999; Vassilev et al., 2000). Jones and others (Hordijk, 1996; Jones & Forrest, 1995a) further suggest that each operator employed by a search algorithm (e.g. selection, mutation and crossover for a genetic algorithm) should be viewed as operating on its own landscape. The notion that search operators define and act on separate landscapes may be useful for studying the effect of individual operators, but the combined effect of each move on each landscape needs to be considered. If the algorithm employs multiple operators, then the output of the algorithm is some combination of those operators, so it is not reasonable to consider a move on the "mutation landscape" without considering how that then affects a subsequent move on the "crossover landscape".

A further problem with the "one operator, one landscape" approach is determining what constitutes an operator. Jones (1995a) has, in the case of evolutionary algorithms, suggested that selection, mutation and crossover should each define and operate on separate landscapes, but none of those operators are necessarily atomic. Is an operator considered to define its own landscape simply because it has a well-known label? The breakdown of the evolutionary operation into the three operators of selection, mutation and crossover seems somewhat arbitrary.

5. Landscape Measures

Early work on the characterisation of landscapes involved analysing structural parameters such as the number and distribution of local minima (Edwards & Anderson, 1975; Sherrington & Kirkpatrick, 1975; Tanaka & Edwards, 1980). More recently, several methods for measuring and analysing landscapes for search algorithms have been proposed. The methods proposed can be categorised into two broad streams: statistical measures (Altenberg, 1995; Hordijk, 1996; Jones & Forrest, 1995b; Lipsitch, 1991; Manderick et al., 1991; Weinberger, 1990b; Weinberger, 1991b) and information measures (Borenstein & Poli, 2005a,b,c; Vassilev, 1997; Vassilev et al., 2000). Borenstein and Poli have extended the information measure to include a measure of the performance of the algorithm and so define a "performance landscape" which may prove useful but does not yet have sufficient history to gauge its efficacy. All the methods proposed have in common the notion that the points in the search space are arranged according to some neighbourhood relationship, and a measure of fitness, performance or information content associated with the points defines the ruggedness of the landscape.

The methods used to measure and analyse the structure of fitness landscapes in this work are the autocorrelation method suggested by Weingberger (1990b; 1991b), and the information content approach suggested by Vassilev et al. (Vassilev, 1997; Vassilev et al., 2000). These methods were chosen because they are different methods of measuring the structure of landscapes and, while there seems to be no generally accepted standard approach, both methods have gained some favour and are commonly cited as reasonable landscape characterisation methods (Hordijk, 1996; Merz & Freisleben, 1999; Smith et al., 2002; Stadler, 1996). Jones' *Fitness Distance Correlation* (Jones & Forrest, 1995b) is an interesting landscape measure but requires that the solution be known in order to calculate the metric, so is not applicable to this work.

5.1 Autocorrelation and Correlation Length

Correlation measures are the most commonly used means of characterising fitness landscapes and are seen as good indicators of ruggedness. A landscape is characterised as rugged when it contains multiple local maxima (minima). Typically, the more rugged the landscape the lower the average fitness correlation between neigbouring points on the landscape. Autocorrelation refers to the correlation of a time series with its own past and future values, and is also known as serial correlation, referring to the correlation between members of a series of numbers arranged in time. Autocorrelation is a correlation coefficient, but instead of measuring the correlation between two different variables, autocorrelation measures the correlation between two values of the same variable at times X_i and X_{i+k}, where k is the number of time steps, or lag, between the two values.

Autocorrelation definition (Hordijk, 1996; Weinberger, 1990b; Weinberger, 1991b):

Given measurements $Y_1, Y_2, \ldots, Y_N$, at time $X_1, X_2, \ldots, X_N$, where N is the number of measurements, and

$$\overline{Y} = \frac{1}{N}\sum_{i=1}^{N} Y_i, N > 0$$

the time lag autocorrelation function is defined as

$$r_k = \frac{\sum_{i=1}^{N-k}(Y_i - \overline{Y})(Y_{i+k} - \overline{Y})}{\sum (Y_i - \overline{Y})^2}, N > k \qquad (1)$$

Note that from Equation 1, if $|r_k| \approx 1.0$ there is much correlation between the points k steps apart in the series, whereas if $|r_k| \approx 0.0$ there is little correlation.

Weinberger proposed that a random walk be generated on the fitness landscape, where each step on the walk is taken between neighbouring points but the neighbour to which the step is taken is selected randomly, and the fitness values for each point visited during the random walk form a time series of numbers. The autocorrelation function can then be used as a measure of the ruggedness of the landscape described by the random walk.

The correlation length of a series of numbers is the largest distance, or time lag, between points for which some correlation exists. Hordijk (1996) defines the correlation length of a time series as one less than the first time lag for which the autocorrelation falls inside the region bounded by the two-standard-error bound (i.e. one less than the first time lag at which the autocorrelation becomes statistically equal to zero, making the correlation length the largest time lag for which the correlation between two points is still statistically significant). This is the method used for calculating the correlation length in this work. The two-standard-error bound *e* for a series of *N* points is defined as

$$e = \pm \frac{2}{\sqrt{N}}, N > 0$$

so the correlation length *l is* defined in this work as the first lag for which

$$|r_k| < \left| \frac{2}{\sqrt{N}} \right|, N > 0 \qquad (2)$$

5.2 Information Content

An alternative to the statistical autocorrelation measure proposed by Weinberger is Vassilev's information content method, based on both classic and algorithmic information theory (Chaitin, 1987; Shannon, 1948). Vassilev et al. propose three information measures that characterise the structure of a fitness landscape from a series of points generated by a random walk over the landscape (Vassilev, 1997; Vassilev et al., 2000) :

- Information Content – characterises the ruggedness of the landscape.
- Partial Information Content – measures the modality of the landscape.
- Information Stability – the sensitivity of the information content measures.

These measures are calculated by generating a random walk of length *n* on the fitness landscape, with the aim being to extract information by characterising the series of points as an ensemble of objects. To calculate the information content, a string $S(\varepsilon)$ is constructed representing a group of objects generated from a random walk over the fitness landscape, where

$$S(\varepsilon) = S_1, S_2, ..., S_n, S_i \in \{\bar{1}, 0, 1\}$$

and $S(\varepsilon)$ is enumerated according to the function

$$S_i = \psi_{f_t}(i, \varepsilon), t = 1..n$$

and

$$\psi_{f_t} = \begin{cases} \bar{1} & if \quad f_i - f_{i-1} < -\varepsilon \\ 0 & if \quad |f_i - f_{i-1}| \le \varepsilon \\ 1 & if \quad f_i - f_{i-1} > \varepsilon \end{cases}$$

Thus the string $S(\varepsilon)$ defines a sequence of objects where each object is represented by a substring S_iS_{i+1} being a sub-block of length two of the string $S(\varepsilon)$.

The parameter ε is a real number taken from the interval [0.0, 1.0] (in this case) which defines neutral fitness and determines the accuracy with which the string $S(\varepsilon)$ is defined. If the absolute fitness difference between neighbouring points is less than ε, the points are

considered to be of equal fitness. This means that as ε increases from 0.0 to the maximum possible fitness difference between points along the walk (1.0), the amount of fitness change (entropy), and the sensitivity of Ψ_{f_t}, decrease to zero.
The information content *H(ε)* is defined as the entropic measure of the group of sub-blocks of length two of string *S(ε)*, and is given by

$$H(\varepsilon) = -\sum_{p \neq q} P_{[pq]} \log_6 p_{[pq]} \tag{3}$$

$P_{[pq]}$ are frequencies of the possible blocks *pq* of elements from the set $\{\bar{1},0,1\}$ given by

$$P_{[pq]} = \frac{n_{[pq]}}{n}$$

where n[pq] is the number of occurrences of pq in *S(ε)* and n is the length of *S(ε)*.

The partial information content *M(ε)* is a measure of the modality (the number or frequency of local optima) of the landscape, and is calculated by filtering out elements of the string *S(ε)* which are not essential for measuring modality to create a new string *S′(ε)*, then measuring the length μ of the new string. The string *S′(ε)* is defined as

$$S'(\varepsilon) = S_{i_1}, S_{i_2}, \ldots, S_{i_k}, S_{i_j} \neq 0, S_{i_j} \neq S_{i_{j-1}}, j > 1$$

Thus the string *S′(ε)* has the form $\bar{1}\,1\,\bar{1}\,1\,\bar{1}\ldots$, representing the slopes of the path taken by the random walk over the landscape, and so is empty if *S(ε)* contains only 0s.
The partial information content *M(ε)* is given by

$$M(\varepsilon) = \frac{\mu}{n} \tag{4}$$

Note that when *M(ε) = 1.0*, the path taken by the random walk over the landscape is considered to be maximally multimodal, and when *M(ε)* = *0.0*, the path is flat.
The information stability *ε** is defined as the smallest value of ε for which the landscape becomes flat (i.e. for which *S′(ε)* is empty). Since ε governs the sensitivity of the information content and partial information content measures, *ε** is a measure of the difference in fitness between neighbouring points on the random walk.

6. Fitness Landscape Definition

As discussed in sections 3 and 4, there has been much previous work done with regard to fitness landscapes, and there is a variety of views and some disagreement. The previous sections of this work have presented a discussion of the issues. For the purposes of this work it is critical that the fitness landscape being analysed be formally defined, and that the definition be related to the search algorithm used to search the landscape.
Genetic algorithms are typically thought of as performing a search on a landscape described by single-bit mutation, where mutation performs a *local* search and the crossover operation is depicted as a *hyperstep* on the mutation landscape, allowing mutation to perform a local

search on a different, usually distant part of the landscape. For this work however, the fitness landscape is considered to be defined by the overall operation of the genetic algorithm.

The autocorrelation and information content measures used in this work to characterize the fitness landscape provide an analysis of a series of numbers: in the case of the autocorrelation measures, the analysis indicates how well correlated numbers in the series are. Of significance is not so much how the numbers in the series are collected or generated, but that the series be representative of the entity – in this case the fitness landscape – being measured. The random walk proposed by Weinberger is a good means to generate a representative series of points for the single-bit mutation landscape (provided that the walk is sufficiently long or that the landscape is statistically isotropic), but is not directly applicable to the landscape defined by the overall operation of the genetic algorithm.

Consider an observer watching a genetic algorithm searcher perform a *random walk* on a fitness landscape and assume that although the observer is able to discern the granularity of the search (the genetic algorithm's single steps), the means by which the GA determines where each step takes it is hidden from the observer.

The observer sees the searcher walking randomly over the landscape and considers points on the landscape one step apart to be neighbours. The definition of the neigbourhood relation is of no consequence to, and is not required by, the observer since the searcher is defining neigbouring points by performing the walk. If the random walk performed by the genetic algorithm searcher was sufficiently long, and the "altitude" (fitness) at each step recorded for the observer, the entire fitness landscape would be determined by observation. The landscape so determined would be the precise fitness landscape defined by the search algorithm.

A random walk of s steps is conducted as follows:

- An individual i_0 is randomly selected from the search space
- For each step s, $s = 1 .. maxsteps$
- i_{s-1} undergoes mutation with probability $P_{mutation}$
- Another individual i, $i \neq i_{s-1}$, is randomly selected from the search space
- Crossover is performed between i and i_{s-1} with probability $P_{crossover}$, resulting in two new individuals i'_1 and i'_2, both of which are neighbours of (a single step from) i_{s-1}
- Set $i_s = i'_1$ and step to i_s

This "black box" view of the genetic algorithm operation and consequential determination of the neigbourhood relation and fitness landscape doesn't change the actual operation of the genetic algorithm or the conceptual notion of the algorithm conducting a parallel search of different areas of the fitness landscape. What this view does is change the notion of the step size of the genetic algorithm from the result of a single genetic operator to the amalgamation of the genetic operators used by the algorithm, so the perception of the topography of the landscape is changed accordingly. This view of the fitness landscape satisfies the requirement that the landscape neigbourhood relation be defined by the search algorithm and is the definition used for the robot soccer problem addressed by this work.

7. Search Space and Fitness Landscape Analysis

Since the skills with which a player is endowed will undoubtedly affect the ability of the player to learn goal scoring behaviour, a key question is *"How does changing the skills available to the soccer player affect, or change, the fitness landscape?"* It would seem to be somewhat intuitive that changing the skills available to the players in some way alters the fitness landscape over which the search is conducted. In fact the fitness landscape is altered by changing the skills available to players, but it changes because the underlying search space is different for each set of skills: the chromosomes are interpreted differently for each skillset, and in fact may take on different values for each skillset depending upon the range of skills available in the skillset, so the individuals (players) that comprise the search space are different for each skillset. Since changing the skills available to the players defines a new search space it also defines a new fitness landscape.

The same is not true of changing the function used for the fitness evaluation. Different fitness functions alter the fitness landscape, not the underlying search space, because the individuals that comprise the search space remain the same. Furthermore, since only the means by which the individuals are evaluated is changed, the change to the fitness landscape is not in the neighbourhood relation but only in the fitness values of the individuals. For this reason, changing the fitness function has the potential to significantly change the fitness landscape, and so affect the effectiveness of the search process. Some fitness functions could make "mountains out of molehills" – teasing gradient information out of otherwise flat landscapes. Sometimes this is valid, but sometimes the definition of the fitness function effectively solves the problem for the search algorithm.

The difficulty of learning goal-scoring behaviour is evidenced by the size of the search spaces defined by the different skillsets of the players. These search spaces range in size from 1.55×10^{158} different chromosomes for the base skillset of {Turn, Kick, Dash} to 7.4×10^{161} for the complete skillset. The calculation of search space size is described in detail in (Riley, 2005).

7.1 Experiments Performed

A number of experiments were performed in order to examine how the fitness landscape, and the performance of the resulting search over it, is affected by varying the player skillset (refer to the full list of available skills shown in Table 1 of the previous chapter), the default action (previous chapter, Table 2) and the fitness function (discussed below). In addition to the evolutionary search, five random walks (as described in section 6) were conducted for each experiment, each walk starting at a randomly selected point on the fitness landscape and continuing for a duration of 100,000 steps. The statistics gathered during the walks are also analysed.

7.2 Fitness Functions Evaluated

Two fitness functions were compared: a composite fitness function and a simple, goals-only fitness function. For both fitness functions implemented the fitness values range from 0.0 to 1.0, with 1.0 being the worst fitness possible, and optimal fitness values approaching 0.0.

The simple, goals-only fitness function rewards a player for goals scored only – the greater the number of goals scored the greater the reward. The goals-only fitness function was implemented as:

$$f = \begin{cases} 1.0 & , goals = 0 \\ \dfrac{1.0}{2.0 \times goals} & , goals > 0 \end{cases} \tag{5}$$

where *goals* is the number of goals scored by the player.

The composite fitness function rewards, in order of importance:

- the number of goals scored in a trial
- minimising the distance of the ball from the goal

This combination was chosen to reward players primarily for goals scored, while players that do not score goals are rewarded on the basis of how close they are able to move the ball to the goal on the assumption that a player which kicks the ball close to the goal is more likely to produce offspring capable of scoring goals. This decomposes the original problem of evolving goal-scoring behaviour into the two less difficult problems:

- evolve ball-kicking behaviour that minimises the distance between the ball and goal, and
- evolve goal-scoring behaviour from the now increased base level of skill and knowledge

The composite fitness function was implemented as:

$$f = \begin{cases} \begin{cases} 1.0 & , kicks = 0 \\ 0.5 + \dfrac{dist}{2.0 \times fieldLen} & , kicks > 0 \end{cases} & , goals = 0 \\ \dfrac{1.0}{2.0 \times goals} & , goals > 0 \end{cases} \tag{6}$$

where *goals* is the number of goals scored by the player
kicks is the number of times the player kicked the ball
dist is the minimum distance of the ball to the goal
fieldLen is the length of the field

Note that both the simple and composite fitness functions represent the problem as a minimisation problem, so a lower fitness value is a better result than a higher fitness value.

7.3 Genetic Algorithm Parameters

The genetic algorithm parameters used in the experiments were chosen empirically after some experimentation. The population size and number of generations were chosen to provide reasonable population diversity and search space coverage. The GA parameters are shown in Table 1. Each player was allowed a fixed time in which to score as many goals as possible.

Skill	Description
Maximum Chromosome Length	64 genes
Population Size	500
Maximum Generations	500
Selection Method	Elitist: *a percentage of the best players is retained, with the remainder selected using the Roulette Wheel method.*
Elitist Retention Percentage	2.5
Crossover Method	Single Point Cut and Splice
Crossover Probability	0.95
Mutation Rate	10%
Mutation Probability	0.15

Table 1. GA parameters

7.4 Results and Analysis

For each experiment the following information is presented for analysis and comparison:

- Data from the evolutionary search:
  - Line graphs showing the population average fitness and individual best fitness for generations 1 to 500. Note that a *lower* fitness is a better fitness.
  - A bar chart showing the cumulative fitness distribution for all individuals evaluated during the 500 generations, showing the percentage of all individuals evaluated which failed to kick the ball (denoted by "nm" on the graph x-axis), moved the ball closer to the goal (graduated), kicked one goal, two goals, three goals etc.
- Data from the random walks:
  - A correlogram showing the autocorrelation data for time lags 1 to 100 for the five random walks and the two-standard error bounds.
  - A graph showing the information content data for $0 <= \varepsilon <= 1.0$ for the five walks.
  - A table showing:
    - the mean fitness and fitness variance of the points visited in the random walks.
    - the fitness distribution for all individuals evaluated during the random walks.
    - the average autocorrelation for the first time lag (i.e. for immediate neighbours).
    - the average correlation length.
    - the average information content $H(\varepsilon)$ and average partial information content $M(\varepsilon)$ for selected ε.
    - the average information stability ε^*.

7.4.1 Experiment 1: Composite Fitness; All Skills; Default Action = Hunt Action 1

Experiment 1 is the problem for which the search algorithm has been given the most help in the form of initial player skills, default action, and a composite fitness function to guide the search. The data shown on the graph of population average fitness (Fig. 4) tend to indicate that the population as a whole ceases to improve after 30 to 40 generations though, as evidenced by the graph of best fitness values, individuals of good fitness continue to be found beyond that point. The percentage of the population exhibiting ball-kicking or goal-scoring behaviour is reasonably high, as shown by the frequency distribution (Fig. 5).

From the data presented in Table 2 it is apparent that although the mean fitness of the 100,000 individuals evaluated during the random walk is close to 1.0 (indicating no ball movement) and the variance small, the random walk did find individuals which exhibited goal-scoring behaviour, and in fact found more than 100 individuals which scored multiple goals.

The autocorrelation data shown in Fig. 6 and the correlation length given in Table 2 indicate that the fitness landscape for this problem (as described by the random walk) offers a reasonable amount of gradient information that the search algorithm can use to guide the search. With an autocorrelation of ~0.32 for points on the random walk a single step apart and a fairly steep descent for points further apart, the correlation between next and near neighbours on this fitness landscape is not so high that a search algorithm is led unerringly to a solution, but with a good correlation and a long correlation length the problem, in this form, should be readily solved by a search algorithm able to take advantage of the landscape features.

Random Walk Statistics (Average over 5 walks)								
Mean Fitness	0.98283							
Fitness Variance	0.00669							
Individual Fitness	**No Movement**	**Ball Movement**	**Goals**					
			1	2	3	4	5	>5
Number of Individuals	95974	3462	434	99	26	4	1	0
Autocorrelation ***r(1)***	0.31716							
Correlation Length	41							
ε	**Information Content** $H(\varepsilon)$				**Partial Information Content** $M(\varepsilon)$			
0.0	0.12447				0.03266			
0.2	0.09770				0.03020			
0.4	0.06179				0.01269			
0.6	0.00754				0.00098			
0.8	0.00229				0.00025			
0.885	0.00000				0.00000			

Table 2. Experiment 1: Random walk statistics

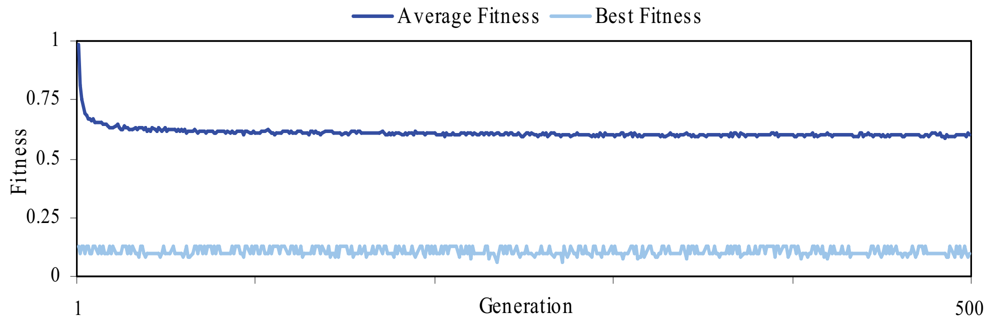

Fig. 4. Experiment 1: Average and best fitness

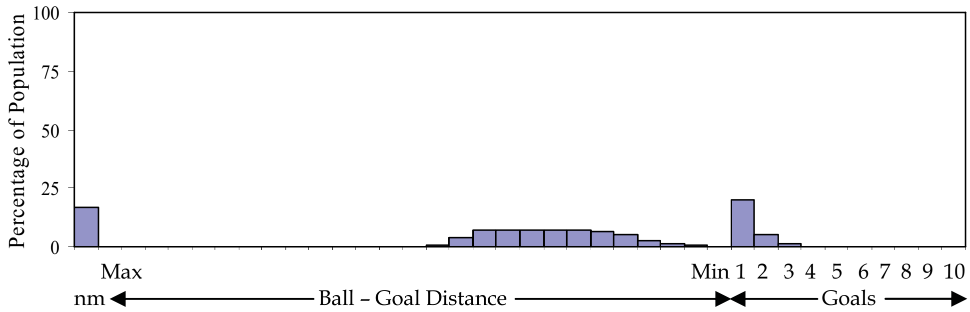

Fig. 5. Experiment 1: Ball movement and goals scored

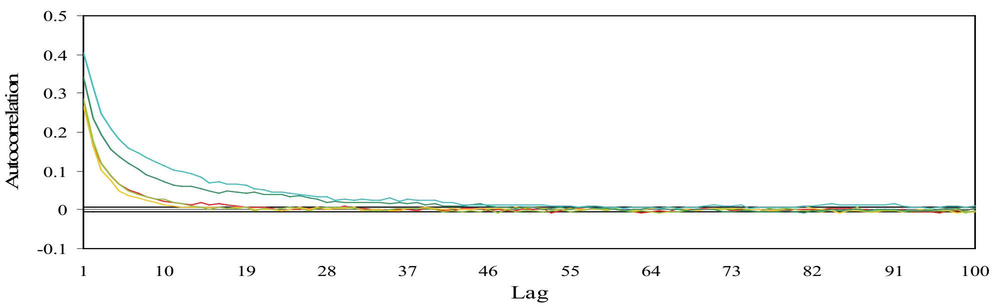

Fig. 6. Experiment 1: Autocorrelation

The information content graph shown in Fig. 7 supports the autocorrelation data for this experiment. Information stability is quite high at 0.885, indicating a high difference in fitness among neighbouring points, so pointing to some good gradient information being present in the landscape. *H(0.0)* is not particularly large, indicating that the diversity of shapes on the landscape is not high. Similarly *M(0.0)* is relatively small, indicating that the degree of modality of the landscape is low.