

How redundancy and neutrality may affect evolution on NK fitness landscapes

Marisol B. Correia, *Member, IEEE*, and Carlos M. Fonseca

Abstract—An experimental study was performed to determine whether it is neutrality itself or the larger neighborhoods associated with neutral representations that allow good results to be achieved on NK fitness landscape problems. Markov chains were used to model a stochastic hill climber on NK fitness landscapes, using three types of representation: a neutral network representation, a redundant representation without neutrality which exhibits the same neighborhood of the neutral representation and a non-redundant representation.

I. INTRODUCTION

The neutral theory of molecular evolution proposed by Kimura [8] considers that it is not selection, but the random fixation of neutral mutations that is the main source of evolution. This scientist observed that, in nature, the rate of evolution in terms of nucleotide substitutions seems to give a value so high that many of the mutations involved must be neutral ones (a mutation is neutral if its application to a genotype does not result in a change of the corresponding phenotype). As large parts of the genotype have no actual effect on the phenotype, i.e., they are redundant, evolution can use them as a store for genetic information that was necessary to survive in the past and is important for developing new properties of the individual that could be advantageous in the future. So, he concluded that a considerable fraction of mutations will be neutral and only a reduced number of them will be reflected at phenotypic level.

In the literature, several redundant representations have been proposed for evolutionary algorithms. The main reason for the development of redundant representations in evolutionary computation has been the desire to achieve increased performance, but the redundant representations proposed so far use large amounts of redundancy and complex mappings [4], [12], [13]. The advantages of using redundant representations are still not evident, and the existing practical results have still not been accepted by all [9]. Others, such as [11], have identified some properties of redundant representations, such as synonymity, locality and connectivity, which are believed to influence the quality of such redundant representations. In [5], practical evidence was provided showing that the connectivity between phenotypes can increase, even when using synonymously redundant representations, in contrast with what was asserted in [11].

In molecular evolution, neutrality plays a positive role in supporting adaptive selection through random drift. Schuster

[14] and Schuster *et al.* [15] simulated the evolution of RNA molecules *in vitro*, which can be visualized as a hill-climbing process on a fitness landscape. They concluded that the presence of neutral networks prevent populations from being caught in evolutionary traps, and allows them to eventually reach the global optimum through a composite dynamics of adaptive walks and random drift.

Neutral networks are connected networks of genotypes which map to the same phenotype. Two genotypes are considered to be connected (or to be neighbors) if they differ by a single point mutation. Theoretically, when the genotype-phenotype mapping exhibits neutrality, it is possible to continue the search by drifting along the neutral networks in genotype space. When none of the neighbors is fitter than the current genotype, there is, at least, one neighboring genotype that maps to the same phenotype, allowing the search to proceed without loss of fitness. As neutral networks exist throughout genotypic space, it is possible to transverse it with less of a chance of becoming stuck at local optima, and it may not be necessary to pass through regions of lower fitness in order to reach regions of higher fitness.

This paper presents the results of an experimental study which used Markov chains to model a stochastic hill-climber on NK fitness landscapes. Three different types of representation are used: the neutral network representation proposed in [5], which exhibits neutrality, a redundant representation without neutrality, but with the same neighborhood of the corresponding neutral case and the non-redundant representation. The purpose is to investigate whether it is in fact the neutrality provided by the representation that allows good results to be achieved on NK fitness landscape problems, or whether the same results can be explained by the larger neighborhood provided by both representations.

The remainder of the paper is structured as follows: Section II presents the neutral and the non-neutral representations used in the study. The NK fitness landscape problem is explained in section III, as well as the difference between the notion of local optimum when using a neutral versus a non-neutral representation. Section IV shows how to calculate the probability of a stochastic hill climber reaching the global optimum of NK fitness landscapes, using a Markov chain model. Section V presents some experimental results. The paper concludes with a brief discussion of the results and some directions for further work.

Marisol B. Correia is with CSI & ESGHT, Algarve University, Faro, Portugal (e-mail: mcorreia@ualg.pt).

Carlos M. Fonseca is with CSI / FCT, Algarve University, Faro, Portugal (e-mail: cmfonsec@ualg.pt).

II. NEUTRAL NETWORK REPRESENTATION VERSUS NON-NEUTRAL NETWORK REPRESENTATION

The family of neutral network representations inspired by error-control codes proposed in [5] exhibits various levels of neutrality, connectivity, synonymity and locality. To understand that representation, a brief explanation of some of the concepts used is presented.

A. Neutral network representation inspired on error-control codes

The codewords of a linear (ℓ, k) Hamming code C [2], defined over the *Galois Field* $GF(2)$ ¹, where the codewords v are obtained from the message word u using a generator matrix of dimension $k \times \ell$, can be defined as:

$$v^T = u^T [I_k \dot{;} P_{k \times (\ell-k)}] \quad (1)$$

To decode the word received, it is necessary to use the *parity-check matrix* H which consists of the P submatrix transposed:

$$H = [(P_{k \times (\ell-k)})^T \dot{;} I_{\ell-k}] \quad (2)$$

To verify if the received word is error-free, the *syndrome* s has to be determined:

$$s = Hv \quad (3)$$

When a word is added to the codewords of a linear code C , new classes of words are created, which can be seen as “translations” of the original codewords. In group theory [6], these classes are termed *cosets* of the linear code C . Each coset C_i is generated choosing a word w that is added to all the codewords v of a linear code C to create a set of all words of the form $v \oplus w$, where \oplus corresponds to *Exclusive-or* or *addition modulo-2*. In this case w is called the *coset leader* or *coset representative*. If C has dimension k , then there are exactly $2^{\ell-k}$ different cosets of C , and each coset contains exactly 2^k words, where the code C itself is one of its cosets. It is easy to show that if $w \oplus v$ is in C , then w and v are in the same coset and have the same syndrome and if $w \oplus v$ is not in C , the w and v are in different cosets and have different syndromes.

Also, according to [3] two linear q -ary codes, represented by two $k \times \ell$ matrices, generate equivalent linear (ℓ, k) codes over $GF(q)$ if one matrix can be obtained from the other by a sequence of the following operations:

- 1) Multiplication of a row by a non-zero scalar;
- 2) Addition of a scalar multiple of one row to another;
- 3) Permutation of rows;
- 4) Multiplication of any column by a non-zero scalar;
- 5) Permutation of columns.

While the first three row operations preserve the linear independence of the rows of the generator matrix, the last

¹Consists of the two-element set $\{0,1\}$, where the addition and multiplication operations correspond to *exclusive-or* or *addition modulo-2* and operations, respectively.

two column operations convert the matrix to one which will produce an equivalent code.

If the size of the genotypic space is $|\Phi_g| = 2^\ell$ and the size of the phenotypic space is $|\Phi_p| = 2^k$, then the genotype space can be divided into $2^{\ell-k}$ classes of equal cardinality 2^k in such a way that single gene mutations allow to move from one class to another. Viewing the word transmitted v as the genotype and the word decoded after transmission u as the phenotype, one of the $2^{\ell-k}$ classes can be seen as the codewords of a Hamming (ℓ, k) code C . The other $2^{\ell-k} - 1$ classes are cosets of the linear code C . One difference exists between these cosets and the cosets defined before. In this case the coset leader do not need to be the vector of least weight as is defined in the context of error-control coding and group theory. If the coset leaders are chosen in such a way that they are connected, then every coset leader is a single point mutation of, at least, one of the other coset leaders, and a connected neutral network can be defined by the coset leaders. The genotype-phenotype mapping adopted in the neutral network representation used in this work can be defined as:

- 1) $u = [I_k 0_{k \times (\ell-k)}]v$ if $v \in$ codewords (coset C_0) \rightarrow syndrome = 0
- 2) $u = [I_k 0_{k \times (\ell-k)}](v + z_j)$ if $v \in$ coset $C_j \wedge 0 < j < 2^{\ell-k} \rightarrow$ syndrome = j

The vectors z_j are the coset leaders that are chosen as the genotypes that represent the all-zero phenotype (the coset leaders will be called *zeros*). In order to obtain the phenotype that corresponds to a given genotype, the coset C_j to which that genotype belongs has to be determined through syndrome decoding, and then the corresponding coset leader has to be added (addition modulo-2) to the given genotype. As the mapping between each coset C_j and the space of the phenotypes is defined through the corresponding neutral coset leader z_j , different genotypes that map to the same phenotype may also reach different phenotypes through a single point mutation. Next, the minimum canonical form of the neutral network representation will be defined.

B. Minimum canonical form of neutral network representation

Definition 1 A connected neutral network² defined using a redundant binary genotype-phenotype mapping $f_g(x_g) : \Phi_g \rightarrow \Phi_p$ that determines which phenotypes are represented by which genotypes, where $|\Phi_g| = 2^\ell$ represents the size of the genotype space and $|\Phi_p| = 2^k$ represents the size of the phenotype space, can be defined based on a linear code C (ℓ, k) , with cardinality $L = 2^k$, generated by a generator matrix of dimension $k \times \ell$, as a vector of $L = 2^{\ell-k}$ genotypes:

$$(z_0, z_1, \dots, z_{L-1}) \quad \forall i \quad z_i \in C_i \wedge z_i \in \{0, 1\}^\ell$$

All of these genotypes map to the zero phenotype, where each zero is chosen from each of the $2^{\ell-k}$ cosets C_i of the linear

²For the purpose of simplification the term neutral network is used instead of connected neutral network.

code C , where for each zero there is, at least, another zero at a Hamming distance of 1:

$$\forall i \quad \exists j \quad d_H(z_i, z_j) = 1$$

A neutral network is in a minimum canonical form if:

$$(z_0, z_1, \dots, z_{L-1}) \quad z_0 = 0 \quad z_i \in \{0, 1\}^\ell$$

where:

- 1) The first zero is chosen to be the all-zero genotype;
- 2) The vector of zeros is the lexicographic minimal representation of all equivalent codes.

The set of neutral networks which corresponds to a given ℓ and k will be denoted as $\text{NN}(\ell, k)$. By default $\text{NN}(\ell, k)$ denotes the neutral networks which represent the phenotype 0, the *zeros*. When indicated as $\text{NN}(\ell, k)_{p_i}$, it denotes the neutral networks which represent the phenotype p_i . The phenotypes which are neighbors of a neutral network $\text{NN}(\ell, k)_{p_i}$ are the phenotypes encoded by the neighboring genotypes (Hamming distance of 1) of the *zeros* of $\text{NN}(\ell, k)_{p_i}$. Also, by the context, it is possible to distinguish when it is being used as a specific neutral network or as the set of all neutral networks. In [5] the number of neutral networks obtained for each $\text{NN}(\ell, k)$, where $0 < k \leq 8$ and $0 < \ell - k \leq 4$, is listed.

C. Non-neutral representation with same neighborhood of neutral case

Consider $\text{NonNeutral}(\ell', k)$ a set of redundant representation without neutrality as defined in [5]. In this case the non-neutral genotype-phenotype mapping is defined by:

$$u = G \cdot v \quad (4)$$

where G is a $k \times \ell'$ binary matrix which columns consist of the phenotypes which are reachable from the all-zero phenotype through single gene mutations and the rows determine how likely each trait is to be changed through a single gene mutation in comparison to the other traits. If G is denoted as $G = [g_{\ell'-1}, \dots, g_0]$, where each g_i denotes a column of G and $0 \leq i \leq \ell'$, the result of a single-bit mutation at position i of v may be written as $v + e_i$, where e_i is a vector of length ℓ' with a single non-zero bit at position i . In this case:

$$G \cdot (v + e_i) = G \cdot v + G \cdot e_i = u + g_i \quad (5)$$

When u is zero, g_i is the phenotype obtained through mutation of gene i . The effect of mutation on an arbitrary phenotype u does not depend on the original genotype v , but only on the bit mutated and the corresponding column of G .

Selecting ℓ' and the appropriate G , it is possible to define the same phenotypes neighborhood of a $\text{NN}(\ell, k)$ representation, in terms of connectivity³ and phenotypes.

³Number of phenotypes accessible from a given phenotype

III. NK FITNESS LANDSCAPES TRANSFORMATION

A. NK fitness landscape problem

The NK landscapes [7] are stochastically generated fitness functions on bit strings parameterized with N genes and K interactions between genes. The NK fitness landscapes allow to measure the difficulty of finding good solutions based on the difficulty of climbing up to the globally optimal fitness solution, avoiding locally optimal solutions. The main parameters of the model are N , the length of the binary strings or chromosomes, that form the points in the landscape, and K , the number of other genes that influence a particular gene, where the fitness contribution of each gene is determined by the gene itself and K other genes. Two different alternatives can be chosen, adjacent neighborhoods⁴, which was the alternative chosen for this study, where the K genes nearest to the locus which is being evaluated on the chromosome are chosen, or random neighborhoods, where the K genes are chosen randomly in the chromosome. The computational complexity of the adjacent neighborhoods variant is $O(2^K N)$, thus in P , while the random neighborhoods alternative is NP complete for $K \geq 2$ and in P for $K = 1$ [16], [18]. The fitness contributions are drawn from a uniform distribution ranging from 0.0 to 1.0 and the fitness of a chromosome is the average fitness of the genes at all N loci. The ruggedness of the landscape is controlled by the parameter K , and is largest when K takes the maximal value of $N - 1$. When $K = 0$ (no epistasis), there is a single peak, the problem is unimodal, and the fitness of strings is highly correlated with Hamming distance. When $K = N - 1$ (maximum number of interactions between genes), there are many sub-optimal peaks and the fitness of strings is uncorrelated with Hamming distance. Also, as both N and K increase, an increasing number of fitness peaks fall towards the mean fitness as a result of conflicting constraints among the genes.

B. Local optimum analysis

As the NK fitness landscape can be gradually tuned from smooth to rugged, it is a good fitness model to study different types of neutral networks. A way to characterize the nature of the landscape is to understand the ruggedness or smoothness of the landscape based on the number and distribution of local optima. As it is known, a landscape is induced by the operator which is used to define neighborhoods [10]. In this case, as the landscape is defined over the binary space, the Hamming metric is used and the neighborhood relation can be represented by a graph, the well known Hamming cube. Using the Hamming cube with one more dimension, a Hamming hypergraph can be defined. The Hamming hypergraphs are a good tool to be used to represent a NK fitness landscape. Figure 1 represents a $\text{NK}(4, 1)$ fitness landscape, where $N = 4$ and $K = 1$. Having in mind that a *local optimum* in the NK landscape is a point which is better (in this case higher because the optimization problem is to

⁴This notion of neighborhood is different from neutral network neighborhood

be maximized) than any of the points of the search space that surround it (if it is the best possible solution to the problem, it is also the global optimum), the global optimum corresponds to genotype 1101(0.658), while 0010(0.422) and 1110(0.645) are local optima. Graphically, the global optimum in Figure 1 is represented as a rectangle, while the local optima are represented as circles. As the number of

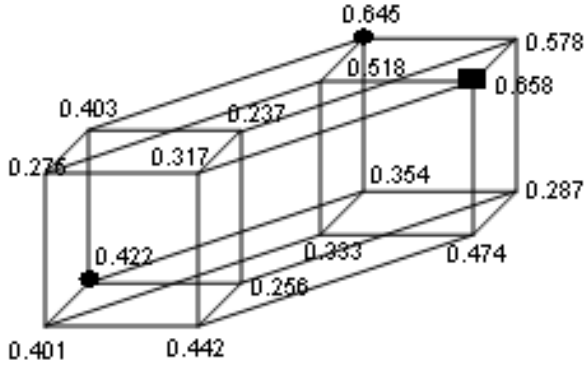


Fig. 1. Hypergraph representing an instance of a NK(4,1) fitness landscape

local optima is a measure of NK fitness landscape difficulty, the neutral networks were used to check if it is possible to reduce that number. From the neutral networks defined with 3 redundant bits, the NN(7, 4) set of neutral networks can be used in order to change the number of local optima in a NK(4, 1) fitness landscape. Now, instead of 2^4 genotypes, the NK fitness landscape is transformed into a landscape with 2^7 genotypes. Figure 2 shows a hypergraph which represents the genotype search space of that fitness landscape [1]. For simplicity, only some genotypes are denoted and the four leftmost bits are highlighted. To better understand the figure, an example of a genotype, for instance 1100101, is given, as well as the corresponding neighbors. Graphically, the genotype in question is drawn inside a rectangle, while its neighbors are inside a circle.

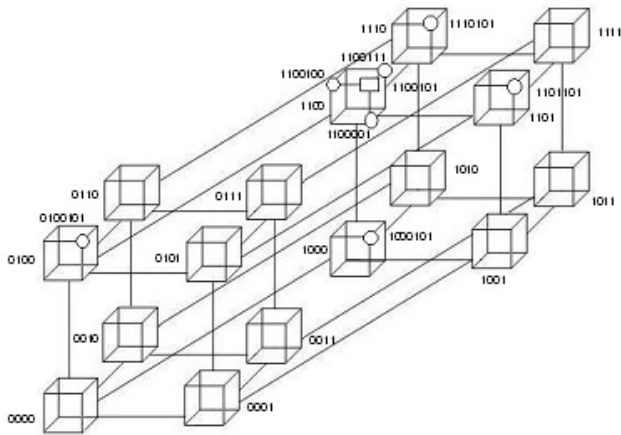


Fig. 2. Hypergraph representing genotype search space with $\ell = 7$ representation to an instance of a NK fitness landscape and

using the genotype-phenotype mapping presented in [5] and attributing the corresponding fitness to each phenotype, it is easy to verify that it is not a point of the search space that is a local optimum or not, instead it is the neutral network that represents a phenotype (*phenotype neutral network*) that is a local optimum or not. As there is always, at least, one neighbor with the same fitness (because there is always a neighbor which maps to the same phenotype), it is possible to use that neighbor to go to other phenotype neutral network with better fitness. Then, it is sufficient that one of the genotypes which belong to a phenotype neutral network not to be a local optimum (with some neighbor with better fitness) to consider that phenotype neutral network as not being a local optimum. As all genotypes that map to a specific phenotype are connected together through a neutral network, instead of a local optimum, it is possible to think of *neutral network local optimum*. The differences between the two are:

- 1) A local optimum is a point in the landscape which is better (better can be higher if optimization problem is to be maximized or can be lower if optimisation problem has to be minimized) than any of the points of the search space that surround it (if it is the best possible solution to the problem, it is also the global optimum). If the fitness of some neighbor is equal to the fitness of the point in question, then the other neighbors define whether the point is a local optimum or not.
- 2) As all genotypes have, at least, a neighbor with the same fitness, that *neutral neighbor* can be used to “escape” from that phenotype neutral network, in order to, when possible, reach a fitter phenotype neutral network. This can be visualized as “bridges” or “plateaux” in the fitness landscape.

In the next section, the probability of reaching the global optimum of NK fitness landscapes using a stochastic hill climber modelled as a Markov chain will be explained. It is important to know that the evolution process of a stochastic hill climbing can be modelled as a Markov Chain [17]. The variant of the *Hill Climber* heuristic [17] used here considers that given the current individual i , a neighbor j is randomly generated and is accepted only if it has a fitness not lower than the fitness of i . Note that, as defined, this *Hill Climber* will never stop.

IV. MARKOV CHAIN MODELLING

The evolution process of a stochastic hill climber can be modelled as a Markov Chain, because the transitions of an individual correspond to the transformation operations between neighbors in the hill climber [17]. In fact, the current individual influences the possible transitions to the next individual, and the evolution of an individual is time homogeneous.

As the goal is to compare the probabilities obtained using the neutral network representation, the non-neutral representation and the non-redundant case, the way how the Markov chain transition matrix is determined and the probability of

reaching the global optimum in NK fitness landscapes will be explained for these three cases. Next a brief introduction of Markov chains will be presented (for more explanations see [17]).

A. Markov chain

Definition 2 A Markov chain is a sequence of random variables $\{X_0, X_1, X_2, \dots\}$ which satisfies the Markov property based on the assumption that the probability to go to the next state is entirely dependent on the current state. The Markov property can be stated as:

$$\begin{aligned} \Pr(X_{n+1} = i_{n+1} | X_0 = i_0, X_1 = i_1, \dots, X_n = i_n) = \\ \Pr(X_{n+1} = i_{n+1} | X_n = i_n) \end{aligned} \quad (6)$$

The Markov property means the system is memoryless, it does not “remember” the states it was in before, just “knows” its present state, and bases its “decision” to which future state it will move to, purely on the present, not considering the past. The changes of state are called transitions. The Markov Chain that is considered here is *time homogeneous*, the transition probabilities are independent of the current point of time, so the transition matrix stays constant during time evolution. As the state space is finite, the transition probabilities, can be defined as:

$$p_{ij} = \Pr(X_{n+1} = j | X_n = i)$$

The one-step transition matrix is defined as $P = [p_{ij}]$, where $0 \leq p_{ij} \leq 1$ and $\sum_{j=1}^{\text{num states}} p_{ij} = 1$. Raising the transition matrix to power k , the new matrix will contain the probabilities of going from each state to each state after k times. The long-term transition matrix will predict where the system (in this case, the hill climber) will end up in the long run.

Next, for simplicity reasons, the way in which the Markov chain transition matrix is determined and the probability of reaching the global optimum in NK fitness landscapes will be explained for the non-redundant case first.

B. Non-redundant representation case

Consider that $\text{Neighbor}(i, j)$ defines the neighbor j of an individual i , $|b_i|$ is the cardinality of the set b_i of neighbors of i , and the set of better or equally good neighbors is defined as B_i [17]:

$$B_i = \{j | \text{Neighbor}(i, j) \wedge \text{Fitness}(j) \geq \text{Fitness}(i)\}$$

Each element of the Markov chain transition matrix P for the non-redundant representation is defined by:

$$p_{ij} = \begin{cases} \frac{1}{|b_i|} & j \in B_i \\ 1 - \frac{|B_i|}{|b_i|} & i = j \\ 0 & \text{otherwise} \end{cases}$$

If an individual i only has neighbors with a lower fitness, meaning that i is a local optimum, then $B_i = \phi$. In this case the chain will remain in this state forever without finding any new and better individuals (i is absorbing). In the transition

matrix, for each local optimum, the corresponding column will appear with a unique value of 1 in the corresponding row, meaning that there is no chance to continue to other neighbors. Taking the example of the NK fitness landscape of Figure 1 which have three local optima, the matrix P displayed in Table I corresponds to the Markov chain transition matrix obtained.

C. Neutral network representation case

Consider a $\text{NN}(\ell, k)$ which corresponds to a neutral network representation with 2^ℓ genotypes and 2^k phenotypes. For each genotype the respective phenotype has to be calculated using the *zeros* of the neutral network in question. In this case $\text{Neighbor}(g_i, g_j)$ defines the genotype neighbor g_j of a genotype g_i , ℓ is the number of neighbors of g_i and the set of better or equally good neighbors is defined as B_{g_i} :

$$B_{g_i} = \{g_j | \text{Neighbor}(g_i, g_j) \wedge \text{Fitness}(p_{g_j}) \geq \text{Fitness}(p_{g_i})\} \quad (7)$$

$$\text{Fitness}(p_{g_i}) \quad (8)$$

$$(9)$$

Each element of the Markov chain transition matrix P for this representation is defined by:

$$p_{ij} = \begin{cases} \frac{1}{\ell} & g_j \in B_{g_i} \\ 1 - \frac{|B_{g_i}|}{\ell} & g_i = g_j \\ 0 & \text{otherwise} \end{cases}$$

In this case the transition matrix is $2^\ell \times 2^\ell$, the principles are the same presented for the non-redundant case, but adapted for the redundant with neutrality case.

D. Non-neutral with same neighborhood representation case

Consider a $\text{NonNeutral}(\ell', k)$ equivalent in terms of phenotypic neighborhood to a $\text{NN}(\ell, k)$. $\text{Neighbor}(p_i, p_j(\text{NN}(\ell, k)_{p_i}))$ defines the phenotype neighbor $p_j(\text{NN}(\ell, k)_{p_i})$ of phenotype p_i , where $p_j(\text{NN}(\ell, k)_{p_i})$ corresponds to phenotype p_j which is neighbor of the neutral network which represents the p_i phenotype (see II-B to remember what are neighbors of the neutral network). Also ℓ is the number of genotype neighbors of the equivalent $\text{NN}(\ell, k)$ and the set of better or equally good neighbors is defined as B_{p_i} :

$$B_{p_i} = \{p_j | \text{Neighbor}(p_i, p_j(\text{NN}(\ell, k)_{p_i})) \wedge \text{Fitness}(p_j(\text{NN}(\ell, k)_{p_i})) \geq \text{Fitness}(p_i)\}$$

Each element of the Markov chain transition matrix P for this representation is defined by:

$$p_{ij} = \begin{cases} \frac{f(p_j)}{\ell'} & p_j \in B_{p_i} \\ \frac{f(p_i)}{\ell'} + \left(1 - \frac{\sum_{p_j \in B_{p_i}} f(p_j)}{\ell'}\right) & p_i = p_j \\ 0 & \text{otherwise} \end{cases}$$

where f corresponds to the frequency of the indicated phenotype. In this case the transition matrix is $2^k \times 2^k$.

TABLE I
MARKOV CHAIN TRANSITION MATRIX OBTAINED FOR THE NK FITNESS LANDSCAPE OF FIGURE 1

$$\mathbf{P} = \begin{bmatrix}
 0.50 & 0.25 & 0.25 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0.75 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.25 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 1.0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0.25 & 0.25 & 0.25 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.25 & 0 & 0 & 0 & 0 \\
 0.25 & 0 & 0 & 0 & 0 & 0.25 & 0.25 & 0 & 0 & 0 & 0 & 0 & 0.25 & 0 & 0 & 0 \\
 0 & 0.25 & 0 & 0 & 0 & 0.50 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.25 & 0 & 0 \\
 0 & 0 & 0.25 & 0 & 0 & 0 & 0.50 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.25 & 0 \\
 0 & 0 & 0 & 0.25 & 0 & 0.25 & 0.25 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.25 \\
 0.25 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.25 & 0.25 & 0 & 0.25 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.75 & 0 & 0 & 0 & 0.25 & 0 & 0 \\
 0 & 0 & 0.25 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.50 & 0 & 0 & 0 & 0.25 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.25 & 0.25 & 0.25 & 0 & 0 & 0 & 0.25 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.50 & 0.25 & 0.25 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1.0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1.0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.25 & 0.25 & 0.50
 \end{bmatrix}$$

E. Probability of reaching the global optimum

Having in mind that the long-term transition matrix P (resulting from a stochastic hill climber had been simulated as a Markov chain) will predict where the hill climber will end up in the long run, then the algorithm to calculate the stationary distribution can be defined as follows:

```

y = global optimum
r[:, 1] = 1 / |search space|
for t from 2 to long run do
    r[:, t] = P × r[:, t - 1]
    p = [p; sum(r(y, t))]
end for

```

The algorithm will be explained using the example presented in Table II, which represents the vector of probabilities obtained for the NK fitness landscape of Figure 1. In this case $r[:, 1]$ corresponds to the probability of presence of an individual in the first state. At the beginning, all states have equal probability $\frac{1}{|\text{search space}|}$, where $|\text{search space}|$ corresponds to the cardinality of the *search space*, which in the NK fitness landscape analysed corresponds to $\frac{1}{24}$. For the neutral case, the global optimum (and each phenotype) corresponds to a phenotype neutral network, as explained in section III-B, with $2^{\ell-k}$ genotypes. Then, it is necessary to calculate the probability p at each t , $p = [p; \text{sum}(r(y, t))]$, which corresponds to the sum of the probabilities for all genotypes which belong to the phenotype neutral network of the global optimum y , at state t . As can be seen the maximum probability 0.5000 is achieved by the global optimum, which corresponds to phenotype 13, a probability of 0.2639 is obtained for the local optimum 14. Finally, the other local optimum 2 has a probability of 0.2361 to be reached. Obviously, $\forall t \sum_{i=1}^{|\text{search space}|} p_{it} = 1$.

V. EXPERIMENTAL RESULTS

In the first part of the experimental study, the probability of reaching the global optimum as a function of the number of iterations is computed for the three types of representation when applied to a particular instance of a NK fitness landscape. The purpose of the second part is to determine

which of the three representations has better behaviour when applied to a set of instances of NK.

The results obtained after calculation of the long-term probability of reaching the global optimum of an instance of NK(11, 1) using one of the neutral networks in NN(14, 11) and the corresponding non-neutral representation with same neighborhood are displayed in Figure 3. In this case, the non-neutral representation is faster than the neutral one, but the final probability is higher for the neutral representation (0.6099) than for the non-neutral representation (0.5375). For this NK(11, 1) instance, the neutral and non-neutral representations do not perform better than the standard, non-redundant binary encoding.

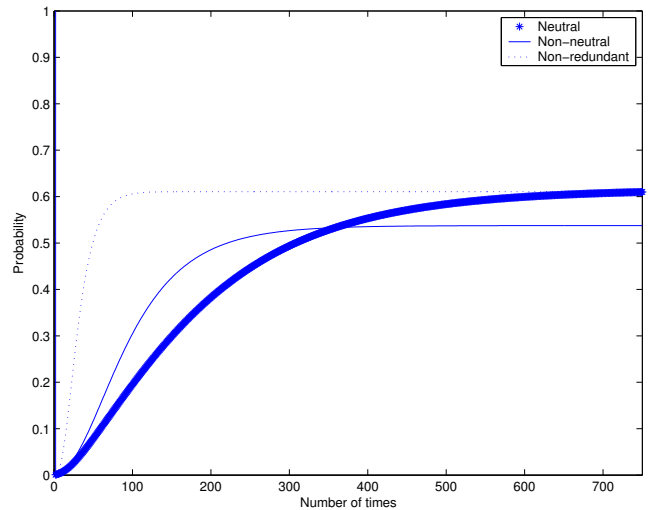


Fig. 3. Long-term probability of reaching the global optimum NK(11,1) using a neutral network in NN(14,11) and a non-neutral encoding with same neighborhood

However, there are neutral networks in NK(11, 1) that lead to better probabilities of reaching the global optimum than both the non-neutral and the non-redundant representations, while being as fast as the equivalent non-neutral and non-redundant ones. Figure 4 shows the behavior obtained with such a neutral network.

TABLE II

PROBABILITIES VECTOR OF REACHING LOCAL OPTIMA APPLIED TO A NK FITNESS LANDSCAPE

1	2	3	...	10	...	20	...	30	...	40	...	50	...	60	t
0.0625	0.0625	0.0312	...	0.0002	...	0.0000	...	0.0000	...	0.0000	...	0.0000	...	0.0000	...
0.0625	0.0938	0.1094	...	0.0230	...	0.0013	...	0.0001	...	0.0000	...	0.0000	...	0.0000	...
0.0625	0.1250	0.1797	...	0.2356	...	0.2361	...	0.2361	...	0.2361	...	0.2361	...	0.2361	...
0.0625	0.0312	0.0078	...	0.0000	...	0.0000	...	0.0000	...	0.0000	...	0.0000	...	0.0000	...
0.0625	0	0	...	0	...	0	...	0	...	0	...	0	...	0	...
0.0625	0.0625	0.0312	...	0.0002	...	0.0000	...	0.0000	...	0.0000	...	0.0000	...	0.0000	...
0.0625	0.0625	0.0312	...	0.0002	...	0.0000	...	0.0000	...	0.0000	...	0.0000	...	0.0000	...
0.0625	0	0	...	0	...	0	...	0	...	0	...	0	...	0	...
0.0625	0	0	...	0	...	0	...	0	...	0	...	0	...	0	...
0.0625	0.0938	0.1016	...	0.0615	...	0.0078	...	0.0007	...	0.0001	...	0.0000	...	0.0000	...
0.0625	0.0625	0.0391	...	0.0005	...	0.0000	...	0.0000	...	0.0000	...	0.0000	...	0.0000	...
0.0625	0.0312	0.0156	...	0.0000	...	0.0000	...	0.0000	...	0.0000	...	0.0000	...	0.0000	...
0.0625	0.0625	0.0312	...	0.0002	...	0.0000	...	0.0000	...	0.0000	...	0.0000	...	0.0000	...
0.0625	0.1250	0.1953	...	0.4148	...	0.4908	...	0.4992	...	0.4999	...	0.5000	...	0.5000	...
0.0625	0.1250	0.1875	...	0.2632	...	0.2639	...	0.2639	...	0.2639	...	0.2639	...	0.2639	...
0.0625	0.0625	0.0391	...	0.0005	...	0.0000	...	0.0000	...	0.0000	...	0.0000	...	0.0000	...

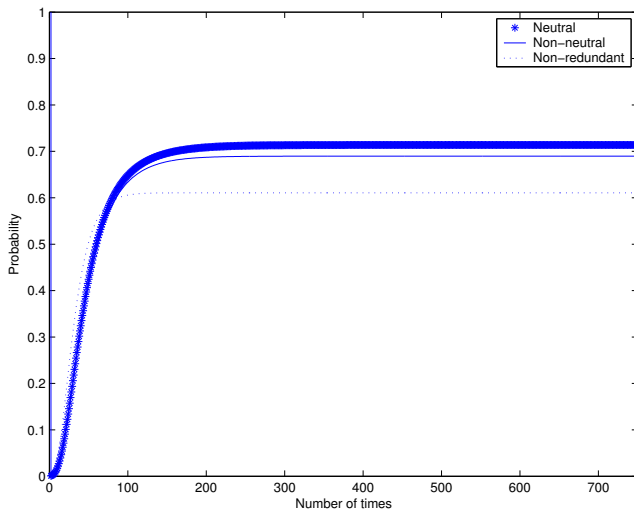


Fig. 4. Long-term probability reaching the global optimum NK(11,1) using a neutral network in NN(14,11) and a non-neutral encoding with same neighborhood

These results show that:

- 1) Neutral networks modify the NK fitness landscapes, and may lead to a higher probability of reaching the global optimum than the corresponding non-neutral representation. The contrary may also happen, i.e., the probability of reaching the global optimum may be lower than with the corresponding non-neutral representation;
- 2) The convergence speed of the hill-climber may be affected by the neutrality of the encoding, even if neutral and non-neutral encodings exhibit the same phenotypic neighbourhood. Although non-neutral representations were observed to usually lead to faster convergence, there were some neutral networks that led to better probabilities and were as fast as the corresponding non-neutral ones.

The second part of the study was performed using 20000 neutral networks randomly chosen from the 2350336 possible neutral networks of NN(14,11) and the corresponding non-neutral counterparts. Eight instances of NK(11, 1) and eight

instances of NK(11, 10) were used to detect the influence of K on the behavior of the representations. Time execution and space storage are the reasons why only 20000 neutral networks were used.

Tables III and IV show the results of this second part of the study. Each of the three columns compares two of the three representations. In each column, the first value corresponds to the percentage of encodings of the first type that were better than the corresponding encodings of the other type, while the second value corresponds to the percentage of encodings of the first type that were equal or worse than those of the first type (Nt refers to neutral network representation, NNt to non-neutral representation and NR to non-redundant representation).

These results show that:

- 1) When K is small (the ruggedness of the landscape is low and there are few local optima) neutral network representations and the corresponding non-neutral representations generally lead to similar behavior, and that both types of representation perform worse than the non-redundant representation;
- 2) When K is large (there are many local optima due to the high ruggedness of the landscape) both redundant representations perform better than the non-redundant representation, and the results of the neutral network representations tend to be slightly worse than those obtained with the corresponding non-neutral representations.

VI. CONCLUSION

In evolutionary computation, the notion of *neutrality* and *neutral networks* has attracted increasing attention for its potential to create alternative paths for evolution and, in this way, improve the quality of the search. An experimental study was developed using Markov chains to model a stochastic hill climber in NK fitness landscapes modelled with three types of representation: the neutral network representation proposed in [5] that exhibits neutrality and interesting properties, a redundant representation without neutrality which exhibits the same neighborhood of the neutral case and the non-redundant representation. The purpose was to detect

TABLE III

RESULTS OF APPLYING NEUTRAL NETWORK (NT), NON-NEUTRAL (NNT) AND NON-REDUNDANT (NR) REPRESENTATIONS ON NK(11,1)

<i>NK instance</i>	% <i>Nt</i> better <i>NNT</i> ; % <i>Nt</i> eq worse <i>NNT</i>	% <i>Nt</i> better <i>NR</i> ; % <i>Nt</i> eq worse <i>NR</i>	% <i>NNt</i> better <i>NR</i> ; % <i>NNt</i> eq worse <i>NR</i>
1	50.51 - 49.49	9.87 - 90.13	9.99 - 90.01
2	50.30 - 49.70	64.84 - 35.16	64.91 - 35.09
3	56.11 - 43.89	0 - 100	0 - 100
4	53.52 - 46.48	48.99 - 51.01	49.02 - 50.98
5	37.72 - 62.28	0.79 - 99.21	0.77 - 99.23
6	66.86 - 33.14	0 - 100	0 - 100
7	66.96 - 33.03	14.32 - 85.68	14.19 - 85.80
8	61.55 - 38.45	35.84 - 64.16	35.33 - 64.67

TABLE IV

RESULTS OF APPLYING NEUTRAL NETWORK (NT), NON-NEUTRAL (NNT) AND NON-REDUNDANT (NR) REPRESENTATIONS ON NK(11,10)

<i>N NK instance</i>	% <i>Nt</i> better <i>NNT</i> ; % <i>Nt</i> eq worse <i>NNT</i>	% <i>Nt</i> better <i>NR</i> ; % <i>Nt</i> eq worse <i>NR</i>	% <i>NNt</i> better <i>NR</i> ; % <i>NNt</i> eq worse <i>NR</i>
1	48.02 - 51.99	74.82 - 25.17	74.78 - 25.22
2	39.25 - 57.24	98.67 - 1.33	98.81 - 1.18
3	47.02 - 52.99	99.78 - 0.22	99.88 - 0.12
4	41.37 - 58.63	54.65 - 45.40	55.04 - 44.96
5	40.71 - 59.29	84.28 - 15.72	84.82 - 15.18
6	44.19 - 55.82	96.75 - 3.26	96.90 - 3.10
7	39.87 - 60.13	85.59 - 14.41	86.30 - 13.70
8	43.89 - 56.12	100 - 0	100 - 0

whether, in the context of NK-landscapes, the neutrality of a representation might influence the performance of a stochastic hill-climber in a way which could not be simply attributed to the larger neighbourhood associated with it.

The results indicate that neutrality may affect both the probability of reaching the global optimum and the speed of convergence, even in comparison to an analogous non-neutral redundant encoding. It is also possible to conclude that the behavior of the three types of representations depends on the ruggedness of the NK fitness landscape. When K is small, the non-redundant representation seems to perform better than the redundant ones; when K is large, neutral and non-neutral representations performed better than the non-redundant one. This suggests that, in general, the search neighborhood induced by the representation affects search performance more strongly than whether or not the representation is neutral. However, the results also show that neutrality may improve search performance beyond the effect of the neighborhood, by structuring the way in which it is searched. Given that the neutral encodings used in this work are much more compact than their non-neutral counterparts [5], neutrality remains an interesting encoding property.

Finally, it may seem disappointing that, over the entire space of neutral representations considered, the percentage of those which bring performance benefits in comparison to non-neutral and possibly non-redundant representations tends to be low, especially in contrast with what happens in nature. However, one must realize that the genetic code itself has been the object of natural selection throughout the years, which may explain why neutrality seems to work well in nature and yet remains difficult to harness in practice.

ACKNOWLEDGMENT

The authors would like to thank Prof. Fernando Lobo for his useful suggestions.

REFERENCES

- [1] G. Barreau, *The Evolutionary Consequences of Redundancy in Natural and Artificial Genetic Codes*, Ph.D. Thesis, University of Sussex, 2002.
- [2] R. E. Blahut, *Digital Transmission of Information*, Addison Wesley, 1990.
- [3] R. Bose, *Information Theory, Coding and Cryptography*, Tata McGraw-Hill, New Delhi, 2002.
- [4] M. Ebner, M. Shackleton and R. Shipman, "How Neutral Networks Influence Evolvability", *Complexity*, vol. 7, pp. 19–33, 2001.
- [5] C. M. Fonseca and M. B. Correia, "Developing Redundant Binary Representations for Genetic Search", *Proc. of the 2005 IEEE Congress on Evolutionary Computation (CEC05)*, vol. 2, pp. 1675–1682, 2005.
- [6] J. F. Humphreys, *A Course in Group Theory*, Oxford Science Publications, 1996.
- [7] S. A. Kauffman, *The Origins of Order - Self-Organization and Selection in Evolution*, Oxford University Press, New York, 1993.
- [8] M. Kimura, "Evolutionary Rate at the Molecular Level", *Nature*, vol. 217, pp. 624–626, 1968.
- [9] J. D. Knowles and R. A. Watson, "On the Utility of Redundant Encodings in Mutation-based Evolutionary Search", *Proc. of Parallel Problem Solving From Nature (PPSN VII)*, pp. 88–98, 2002.
- [10] C. R. Reeves, "Landscapes, Operators and Heuristic Search", *Annals of Operational Research*, vol. 86, pp. 473–490, 1999.
- [11] F. Rothlauf, *Representations for Genetic and Evolutionary Algorithms*, Springer, 2nd Edition, 2006.
- [12] M. Shackleton, R. Shipman and M. Ebner, "An Investigation of Redundant Genotype-Phenotype Mappings and Their Role in Evolutionary Search", *Proc. of the 2000 Congress on Evolutionary Computation (CEC00)*, pp. 493–500, 2000.
- [13] R. Shipman, M. Shackleton, M. Ebner and R. Watson, "Neutral Search Spaces for Artificial Evolution: A Lesson from Life", *Proc. of the Seventh International Conference on Artificial Life*, 2000.
- [14] P. Schuster, "Landscapes and Molecular Evolution", *In Physica D.*, vol. 107, pp. 351–365, 1997.
- [15] P. Schuster and W. Fontana, "Chance and Necessity in Evolution: Lessons from RNA", *In Physica D.*, vol. 133, pp. 427–452, 1999.
- [16] R. Thompson and A. Wright, *Additively Decomposable Fitness Functions*, Technical Report, University of Montana, Missoula, 1996.
- [17] P. G. H. Toonen, *Markov Chain Analysis of Evolutionary Database Optimization*, Master's Thesis, Katholieke Universiteit Nijmegen, 1994.
- [18] E. Weinberger, *NP Completeness of Kauffman's NK Model, a Tuneably Rugged Fitness Landscape*, Technical Report, Santa Fe Institute, 1996.