Proceedings of the 2007 IEEE Symposium on
Foundations of Computational Intelligence (FOCI 2007)

On the BMDGAs and Neural Nets

Marco Carpentieri

Abstract— **We analyze a bivariate marginal distribution genetic model in case of infinite populations and provide relations between the associated infinite population genetic system and the neural networks. A lower bound on population size is exhibited stating that the behaviour of the finite population system, in case of sufficiently large sizes, can be suitably approximated by the behaviour of the corresponding infinite population system for a number of transitions exponentially greater than that suggested by Vose's analysis. The infinite population system is analyzed by showing that, conversely to what happens in the univariate case, the fitness is not a Lyapunov function for its asynchronous variant. The attractors (with binary components) of the infinite population genetic system are characterized as equilibrium points of a discrete (neural network) system that can be considered as a variant of a Hopfield's network; it is shown that the fitness is a Lyapunov function for the variant of the discrete Hopfield's net. The genetic algorithm based on the proposed infinite population system is experimentally compared with the (neural) network algorithm for the Max – Cut problem. Our main result can be summarized by stating that the relation between marginal distribution genetic systems and neural nets is much more general than that already shown elsewhere for the univariate models.**

I. Introduction

Genetic algorithms are probabilistic search algorithms inspired by mechanisms of natural selection and genetics, introduced by John Holland in the 1970s. They have received considerable attention because of their many applications to several research fields such as optimization, adaptive control and others [10], [15], [16], [19].

A classic way to describe the behaviour of genetic algorithms is obtained by means of homogeneous Markov chains [9], [11], [29] whose states encode populations and are multi-sets of binary strings. General theoretical results were introduced for infinite populations by Vose [36], [37] who showed how to use them to perform the qualitative analysis of the behavior of the finite populations models. In particular, in [36], [37], a dynamical system model is introduced for which simulation is computationally difficult. It is worth noting that the original formalism presented by Vose was intended to model situations in which recombination of genetic material is obtained through the crossover of the chromosomes of two mating parents selected with probability in proportion to their fitness. Thus the intractability of simulating any general system (such as Vose's) is due to the fact that it keeps track of every chromosome [38]. To avoid exponential complexity one may consider approximate models, or, alternatively, restrict the type of fitness functions (see for example [31]). A different approach consists in changing one's mind about what is being modeled, thus, representing some other related genetic system. This last alternative is followed by the present paper (along with restriction of the fitness to some classes of polynomial functions) following the guidelines of references [2], [3]; in such papers a model has been proposed in which the recombination of the genetic material is obtained by means of the bit-based simulated crossover operator [33]. This rule, as well as the gene pool recombination introduced by Mühlenbein and Voight [35], [24] maintains an infinite population in linkage equilibrium: the genotype frequencies are the product of marginal frequencies. In this context, a frequency vector with exponentially many components can be reconstructed by the vectors of marginal frequencies that in the univariate marginal distribution algorithms [2], [3], [27], [24], [35], [38] are $l-$component vectors, where throughout by l we mean the chromosome length. Other models based on the marginal frequencies are presented in [1], [13], [20]. Related work can be also found in [38] in which it is analyzed a recombination-mutation-selection genetic algorithm that uses gene pool recombination. In particular Wright et al. show that in case of linear fitness functions there is a single stable fixed point for their univariate marginal distribution genetic algorithm. Moreover, readers interested in exact mathematical analysis of simple genetic algorithms and their use as an alternative approach to combinatorial optimization are referred to [31].

In this paper, we consider the problem of extending the analysis of univariate marginal distribution genetic algorithms for infinite populations (UMDGAs) to the bivariate case (BMDGAs). We review and analyze the genetic model based on simulated crossover of fixed sequences of two bit genes introduced in [5]. Such a model represents an instance of the Random Heuristic Search (as defined in chapter 3 of reference [37]) and can be considered as an extension of the model presented in [33], [35], [2]; the main characteristic of the system that we shall consider is that the recombination of the genetic material is obtained by performing a weighted average of the alleles along each fixed two-bit locus and using such statistics to produce offspring whose alleles in distinct loci are independently generated. The model we propose is more tightly connected to the classic Holland's framework with respect to the models that have been presented in the literature following the more recent development of marginal distribution genetic algorithms (we mean for example the area of the Estimation of

Manuscript received November, 4, 2006.

Marco Carpentieri is with the Dipartimento di Fisica E. R. Caianiello (DF), Università degli Studi di Salerno (Sa), Italy

Distribution Algorithms [23], [26], [30], [39], [40]). In fact, according to the Holland's theory, the fittest individuals chromosomes are formed by classic genetic algorithms merging short definition length and small specificity order allele schemata, whose fitness remains above the average fitness of the populations generated by the genetic cycles (Holland [10], [15], [16]). This central result has introduced the concept of separability of the fitness functions, with respect to short chromosomic traits, that is recognized [8], [28], [32] as a basic property required to justify the application of a (classic) genetic algorithm. The main idea of the model we present is that of proceeding in bottom-up fashion in such a way to compute the (average) fitness associated to some of the chromosome building-blocks (alleles) and translate such information into statistics that are used to generate new offspring. The way in which such statistics are used is aimed at preserving positions of the alleles. Thus, the reader can understand that the sense in which our bivariate model is an extension of the univariate marginal distribution genetic algorithms is different from that proposed to solve problems that contain significant dependencies and that cannot be classified as linear or decomposable problems [30]. Note that the interest in devising marginal distribution genetic models lies not only in the fact that they consent efficient (state transition) implementation for infinite populations. Indeed, in case of univariate marginal distributions, they have been used to construct approximation algorithms to solve hard combinatorial problems for which error bounds can be theoretically estimated. Moreover, the stability analysis of such systems has evidenced an interesting relation with neural networks (in particular with Hopfield's Networks [2], [3]).

We exhibit the following results about the genetic system based on simulated crossover of fixed sequences of two bit genes:

1) we provide an exponential lower bound on the concentration probability (viewed as function of the population size n) stating that for sufficiently large population sizes the finite population stochastic system can be considered as an approximation of the infinite population deterministic system: the result is interpreted stating that the behaviour of the finite population system, in case of sufficiently large sizes n, can be suitably approximated by the behaviour of the corresponding infinite population system for a number of transitions (considered as a function of n) exponentially greater than that suggested by Vose's analysis in [37];
2) the infinite population system is analyzed by showing that, conversely to what happens in the univariate case [4], the fitness is not a Lyapunov function for its asynchronous variant; the attractors in $\{0,1\}^{\frac{3l}{2}}$ (l even) are characterized as equilibrium points of a discrete (neural network) system that can be considered as a variant of a Hopfield's network;
3) it is shown that the fitness is a Lyapunov function for the discrete neural network system;
4) the genetic algorithm based on the proposed infinite population system is experimentally compared with the (neural) network algorithm for the Max-Cut problem; the results show that the performances are comparable being those obtained by the neural network algorithm slightly worse.

II. The Model

We review the model [5] on which the genetic system is based and introduce the technical formalism useful to define states and dynamics. A population P of individuals is represented by a multi-set of $n \in \mathbf{N}$ $l-$length binary strings (throughout the paper we suppose l even) from the set

$$\Omega \quad = \quad \{0,1\}^l \quad = \quad \{\omega_1, \ldots, \omega_{2^l}\}.$$

Each population P is associated with its frequency vector $\mathbf{F} = (F_{\omega_1}, \ldots, F_{\omega_{2^l}})$ specifying the proportion of the strings in Ω contained in P, where $F_{\omega_k} = \frac{n_k}{n}$ and n_k is the number of occurrences of the string ω_k in P. Let Λ_n denote the set of the frequency vectors that represent populations of n individuals. Each individual is evaluated by his fitness that is measured by means of a fitness function $f : \Omega \longrightarrow \mathbf{R}^+$ that associates a positive real value to each chromosome. Throughout the paper, let $A = \{00, 01, 10, 11\}$ and $B = A - \{00\}$. The strings in the populations represent chromosomes and each chromosome is divided into a sequence of genes that can assume four distinct forms or alleles. For $k = 1, \ldots, \frac{l}{2}$ and $a = a_1 \cdot a_2 \in A$, consider functions $\chi_k[a] : \Omega \longrightarrow \{0,1\}$ defined by

$$\chi_k[a](\omega) = \begin{cases} 1, & \text{if } a_1, a_2 \text{ are in positions } 2k-1, 2k \text{ in } \omega; \\ 0, & \text{otherwise.} \end{cases}$$

In the rest of the paper we shall use notation

$$E_{\mathbf{P}}[X] \quad = \quad \sum_{i \quad 1}^{2^l} X(\omega_i) p_i$$

to mean the expectation of function $X : \Omega \rightarrow \mathbf{R}$ considered as a random variable along with the stochastic vector $\mathbf{P} = (p_1, \ldots, p_{2^l})$.

Starting from an initial population P_0, if at time t the state of the (genetic) system is the population P, represented by its frequency vector $\mathbf{F}$, then the population at time $t+1$ is obtained as follows:

1) for every $k = 1, \ldots, \frac{l}{2}$ and $a \in A$ compute

$$\phi_{k,\mathbf{F}}[a] \quad = \quad \frac{E_{\mathbf{F}}[\chi_k[a] f]}{E_{\mathbf{F}}[f]} \quad = \quad \frac{\sum_{i=1}^{2^l} \chi_k[a](\omega_i) f(\omega_i) n_i}{\sum_{i=1}^{2^l} f(\omega_i) n_i};$$

2) generate a new population P' of n $l-$length binary strings, denoted by

$$P' \quad = \quad \{\omega_{r_1}, \ldots, \omega_{r_n}\},$$

with probability $\phi_{k,\mathbf{F}}[a]$ of obtaining a_1, a_2 in positions $2k-1, 2k$ independently from r_i and k for $1 \leq k \leq \frac{l}{2}$ and $1 \leq i \leq n$.

Steps 1. and 2. describe the way in which recombination of the genetic material is obtained: by performing a weighted average of the alleles along each fixed two-bit position and using such statistics to produce offspring

whose alleles in distinct loci are independently generated. The stochastic rule described in 2. is the bit-based simulated crossover introduced in [33], [2] extended to fixed sequences of two-bit genes. By definition of the recombination process described in 2., if P is a population at time t and $\mathbf{F}$ its frequency vector, the population at time $t+1$ is obtained by selecting n strings with probability distribution

$$\Phi(\mathbf{F}) \quad = \quad (\Phi(\mathbf{F})_{\omega_1}, \ldots, \Phi(\mathbf{F})_{\omega_{2^l}}),$$

where the probability $\Phi(\mathbf{F})_{\omega_j}$ of generating the string $\omega_j = \omega_{j,1} \cdots \omega_{j,l}$ is

$$\Phi(\mathbf{F})_{\omega_j} \quad = \quad \prod_{k=1}^{\frac{l}{2}} \phi_{k,\mathbf{F}}[\omega_{j,2k-1} \cdot \omega_{j,2k}].$$

III. Probability Concentration Results

The following theorem states a probability concentration result on the Markov chains describing the stochastic genetic system in Section II. The result of the theorem allows to derive an iterative deterministic system with states in $[0,1]^{3^q}$ that represents the behaviour of the infinite population system. More particularly, for sufficiently large population sizes n, we prove that (provided that the current state of the systems is $\mathbf{F}$) a transition takes the finite population stochastic genetic system near the next state of the infinite population deterministic genetic system with probability close to one. The reader should notice that this fact has been shown in greater generality for any instance of the Random Heuristic Search [37]; however, in case of the considered instance, we are able to provide an exponential lower bound on the concentration probability (viewed as function of n) that is much better than that exhibited in the more general case in [36], [37]. Other results concerning this topic, that can be considered as special cases, can be found in [2], [3], [4].

Theorem 1: Let $\epsilon, \delta \in (0,1]$ and

$$n \quad \geq \quad 2\left(\frac{M}{\epsilon\, E_{\Phi(\mathbf{F})}[f]}\right)^2 \log\left(\frac{6\,l}{\delta}\right),$$

where M is the maximum value that the fitness function can assume; if at time t the system is in the state $\mathbf{F}$, then the state $\mathbf{F}'$ at time $t+1$ is such that for all $k = 1, \ldots, q$ and $a \in B$ it results

$$|\phi_{k,\mathbf{F}'}[a] - \phi_{k,\Phi(\mathbf{F})}[a]| \leq \epsilon \tag{1}$$

with probability at least $1-\delta$.

Proof It is based on Hoeffding's inequality (see [4] and [6]).

To provide a comparison with Vose's analysis [37], setting

$$\underline{\phi}_{\mathbf{F}} \quad = \quad (\phi_{1,\mathbf{F}}[01], \phi_{1,\mathbf{F}}[10], \phi_{1,\mathbf{F}}[11], .., \phi_{q,\mathbf{F}}[01], \phi_{q,\mathbf{F}}[10], \phi_{q,\mathbf{F}}[11]),$$

and assuming that the fitness is lower bounded by some positive constant $K > 0$, we can state the following.

Theorem 2: Let $\epsilon, \delta \in (0,1]$ and

$$n \geq 3\,l\left(\frac{M}{\epsilon\, K}\right)^2 \log\left(\frac{6\,l\,t}{\delta}\right),$$

where the fitness assumes values in $[K, M]$; if the system is in state $\mathbf{F}_\tau$ at time τ for $0 \leq \tau \leq t$, then it holds that

$$P\left(\bigwedge_{\tau=1}^{t} ||\underline{\phi}_{\mathbf{F}_\tau} - \underline{\phi}_{\Phi(\mathbf{F}_{\tau-1})}|| \leq \epsilon\right) \quad \geq \quad 1-\delta. \tag{2}$$

Proof The guidelines of the proof can be found in [6]).

For example, the bound in Theorem 2 holds with $K = 1$ for positive integer fitness functions that are widely used in the applications. Moreover, note that the bound provided by Vose in [37] states that the probability that the infinite population system is a suitable approximation of the finite population system for t state-transitions is bounded by $1 - \frac{t}{n\epsilon^2}$. In practice, the result of Theorem 2 can be interpreted by saying that the behaviour of the finite population system, in case of sufficiently large sizes n, can be suitably approximated by the behaviour of the corresponding infinite population system for a number of transitions (considered as a function of n) exponentially greater than that suggested by Vose's analysis.

IV. Fitness Functions

First of all, we briefly review the topic of efficient implementation in case of arbitrary finite fitness functions $f : \Omega \to \mathbf{R}^+$ and for infinite populations. For $z \in B$ and $k = 1, \ldots, q$, one has

$$\phi_{k,\Phi(\mathbf{F})}[z] = \frac{E_{\Phi(\mathbf{F})}[\chi_k[z] f]}{E_{\Phi(\mathbf{F})}[f]} = \frac{\sum_{j=1}^{2^l} \Phi(\mathbf{F})_{\omega_j} \chi_k[z](\omega_j) f(\omega_j)}{\sum_{j=1}^{2^l} \Phi(\mathbf{F})_{\omega_j} f(\omega_j)}$$

$$= \frac{\sum_{j=1}^{2^l} \chi_k[z](\omega_j) f(\omega_j) \prod_{k'=1}^{q} \phi_{k',\mathbf{F}}[\omega_{j,2k'-1} \cdot \omega_{j,2k'}]}{\sum_{j=1}^{2^l} f(\omega_j) \prod_{k'=1}^{q} \phi_{k',\mathbf{F}}[\omega_{j,2k'-1} \cdot \omega_{j,2k'}]}. \tag{3}$$

By (3) we observe that efficient implementation depends, not only on the dimension of the involved states, but also on the type of fitness function. In this regard, it is well known that f can be expressed in terms of multivariate polynomials and this fact can be (naturally) used to characterize classes of functions for which we are able to perform efficient implementation in the sense that state transitions can be computed in time polynomial in l. Let, now,

$$Pf(x_1, \ldots, x_l) \quad = \quad \sum_{y_1, \ldots, y_l \in \{0,1\}} w_{y_1 \cdots y_l} x_1^{(y_1)} \cdots x_l^{(y_l)} \tag{4}$$
$$(x_i^{(0)} = 1 - x_i,\ x_i^{(1)} = x_i \text{ for } i = 1, \ldots, l)$$

be a multivariate polynomial defined on $[0,1]^l$ and coincident with f on Ω. Notice that since Pf is a polynomial of degree at most one in each variable its global maximum is on elements in Ω.
Denote, for $u = 1, \ldots, q$ and $a \in A$, by $\hat{x}_u[a]$ the product of $x_{2u-1}^{(a_1)}, x_{2u}^{(a_2)}$, that is:

$$\hat{x}_u[a] \quad = \quad x_{2u-1}^{(a_1)} x_{2u}^{(a_2)}.$$

Polynomial (4) can be rewritten in the form

$$Pf(\mathbf{x}) = \sum_{a\in A} \hat{x}_k[a] b_{k,a}(\hat{\mathbf{x}}),$$

where

$$\hat{\mathbf{x}} = (\hat{x}_1[00], \hat{x}_1[01], \hat{x}_1[10], \hat{x}_1[11], .., \hat{x}_q[00], \hat{x}_q[01], \hat{x}_q[10], \hat{x}_q[11])$$

and $b_{k,a}(\hat{\mathbf{x}})$ does not depend on the variables x_{2k-1}, x_{2k} for every $a \in A$ and $k = 1, \ldots, q$. Calculating the expectations, being $\phi_{k,\mathbf{F}}[00] = 1 - \sum_{a\in B} \phi_{k,\mathbf{F}}[a]$ for each $k = 1, \ldots, q$, we get

$$\begin{aligned} & E_{\Phi(\mathbf{F})}[Pf] \\ &= \sum_{a\in A} E_{\Phi(\mathbf{F})}[\hat{x}_k[a]] E_{\Phi(\mathbf{F})}[b_{k,a}] \\ &= \sum_{a\in A} \phi_{k,\mathbf{F}}[a] b_{k,a}(\underline{\phi}_{\mathbf{F}}) \\ &= \sum_{a\in B} \phi_{k,\mathbf{F}}[a] b_{k,a}(\underline{\phi}_{\mathbf{F}}) + \phi_{k,\mathbf{F}}[00] b_{k,00}(\underline{\phi}_{\mathbf{F}}) \\ &= \sum_{a\in B} \phi_{k,\mathbf{F}}[a] b_{k,a}(\underline{\phi}_{\mathbf{F}}) \\ &+ \left(1 - \sum_{a\in B} \phi_{k,\mathbf{F}}[a]\right) b_{k,00}(\underline{\phi}_{\mathbf{F}}) \\ &= \sum_{a\in B} \phi_{k,\mathbf{F}}[a](b_{k,a}(\underline{\phi}_{\mathbf{F}}) - b_{k,00}(\underline{\phi}_{\mathbf{F}})) + b_{k,00}(\underline{\phi}_{\mathbf{F}}) \quad (5) \end{aligned}$$

by linearity of the mean and by independence. Moreover, since

$$E_{\Phi(\mathbf{F})}[\chi_k[z]\hat{x}_k[a]] = \begin{cases} \phi_{k,\mathbf{F}}[z] & \text{if } z = a \\ 0 & \text{otherwise} \end{cases}$$

$(a, z \in A)$, one has for $k = 1, \ldots, q$ and $z \in B$ that

$$\begin{aligned} E_{\Phi(\mathbf{F})}[\chi_k[z] Pf] &= \sum_{a\in A} E_{\Phi(\mathbf{F})}[\chi_k[z]\hat{x}_k[a]] E_{\Phi(\mathbf{F})}[b_{k,a}] \\ &= \phi_{k,\mathbf{F}}[z] b_{k,z}(\underline{\phi}_{\mathbf{F}}). \end{aligned}$$

Therefore, in case of infinite populations, Theorem 1 implies that as $n \to \infty$ the stochastic genetic system converges to an infinite population deterministic system; the states of such a deterministic system are $3q-$component vectors

$$\underline{\psi} = (\psi_{1,01}, \psi_{1,10}, \psi_{1,11}, \psi_{2,01}, \psi_{2,10}, \psi_{2,11}, .., \psi_{q,01}, \psi_{q,10}, \psi_{q,11})$$

in $[0,1]^{3q}$, with $\sum_{a\in B} \psi_{k,a} \leq 1$ $(1 \leq k \leq q)$ and the dynamics is described, for $z \in B$ and $k = 1, \ldots, q$, by the equations

$$\psi_{k,z}(t+1) = \frac{\psi_{k,z}(t) b_{k,z}(\underline{\psi}(t)}{Pf(\underline{\psi}(t))}, \quad (6)$$

where

$$\underline{\psi}(t) = (\psi_{1,01}(t), \psi_{1,10}(t), \psi_{1,11}(t), .., \psi_{q,01}(t), \psi_{q,10}(t), \psi_{q,11}(t)).$$

Note that $\psi_{k,z}(t)$ in Equation (6) represents the probability of having z as the $k-$th allele after t transitions.

The state space of the iterative deterministic genetic system for infinite populations is a subset $\Lambda(3q) \subset [0,1]^{3q}$ of $3q-$component vectors $\underline{\psi}$ such that $\sum_{a\in B} \psi_{k,a}(t) \leq 1$ $(1 \leq k \leq q)$. Moreover, by (6) it is clear that, to be able to perform (state transition) efficient implementation, the terms $b_{k,z}(\underline{\psi}(t))$ for all $k = 1, \ldots, q$ and $z \in B$ must be computed in time polynomial in the chromosome length l. In such class of functions there are important types of fitness functions such as quadratic ones that are useful to model hard optimization problems; a more general class consists of the functions than can be expressed as sums of monomials (products) of at most $O(\log l^c)$ variables, where $c > 0$ is constant.

V. Analysis

By equations (6) it follows that

$$\begin{aligned} & \Delta\psi_{k,z}(t) \\ &= \psi_{k,z}(t+1) - \psi_{k,z}(t) \\ &= \frac{\psi_{k,z}(t)}{Pf(\underline{\psi}(t))}[(1 - \psi_{k,z}(t))(b_{k,z}(\underline{\psi}(t)) - b_{k,00}(\underline{\psi}(t))) \\ &- \sum_{a\in B-\{z\}} \psi_{k,a}(t)(b_{k,a}(\underline{\psi}(t)) - b_{k,00}(\underline{\psi}(t)))] \quad (7) \end{aligned}$$

for $k = 1, \ldots, q$ and $z \in B$. We, first, remark that the fitness is not a Lyapunov function for the asynchronous variant of the system as in the case of the univariate (infinite population) genetic system introduced in [2], [4] (by asynchronous we mean that the components of the state vector are updated one at a time in a predefined order); in case of the bivariate (infinite population) system, we are only able to state necessary conditions such that this property holds. In the other cases, it is quite simple to find fitness functions and states $\underline{\psi}(t)$ for which state-updating in a single locus and for a single allele produces a decrease of the fitness.

Theorem 3: If $\psi_{k,z}(t+1) = \psi_{k,z}(t) + \Delta\psi_{k,z}(t)$ $(z \in B)$ and $\psi_{k,a}(t+1) = \psi_{k,a}(t)$ for $a \in B - \{z\}$, then conditions

$$\prod_{a\in A-\{z\}} HS'(b_{k,z}(\underline{\psi}(t)) - b_{k,a}(\underline{\psi}(t))) = 1$$

or

$$\prod_{a\in A-\{z\}} (1 - HS(b_{k,z}(\underline{\psi}(t)) - b_{k,a}(\underline{\psi}(t)))) = 1$$

imply

$$\Delta Pf(\underline{\psi}(t)) = Pf(\underline{\psi}(t+1)) - Pf(\underline{\psi}(t)) \geq 0,$$

where $HS'(), HS()$ are heavy-side functions defined by

$$HS'(X) = \begin{cases} 1 & \text{if } X \geq 0 \\ 0 & \text{if } X < 0. \end{cases}$$

and

$$HS(X) = \begin{cases} 1 & \text{if } X > 0 \\ 0 & \text{if } X \leq 0. \end{cases}$$

Proof The result follows by quite straightforward manipulations of $\Delta Pf(\underline{\psi}(t))$ and by inspection of the conditions on the $b_{k,a}, b_{k,z}$ polynomials implying that the fitness does not decrease.

Denote by $N\Lambda_{3q} = \{\mathbf{x} \in \{0,1\}^{3q} : \sum_{a\in B} x_{k,a} \leq 1 \text{ for } k = 1,\ldots,q\}$ the subset of the state space composed by the vectors with components each assuming values in $\{0,1\}$. All points in $N\Lambda_{3q}$ are fixed points of System (6) as stated by the following lemma.

Lemma 1: If $\mathbf{x} \in N\Lambda_{3q}$, then it holds that

$$\Delta\psi_{k,z}(t)|_{\underline{\psi}(t)=\mathbf{x}} = 0, \text{ for each } k = 1,\ldots,q \text{ and } z \in B.$$

Proof It is straightforward by (7).

Set $\mathbf{x}_k = (x_{k,01}, x_{k,10}, x_{k,11})$ $(1 \leq k \leq q)$. The next theorem states sufficient and necessary conditions in order that a point $\mathbf{x} \in N\Lambda_{3q}$ is an attractor of the System (6).

Theorem 4: The point $\mathbf{x} \in N\Lambda_{3q}$ is an attractor of the System (6) if and only if for every $k = 1,\ldots,q$ it holds that $\mathbf{x}_k = \mathbf{0}$ and

$$\prod_{a\in B} HS(b_{k,00}(\mathbf{x}) - b_{k,a}(\mathbf{x})) = 1$$

or $\mathbf{x}_k \neq \mathbf{0}$ and

$$\prod_{a\in A-\{z\}} HS(b_{k,z}(\mathbf{x}) - b_{k,a}(\mathbf{x})) = x_{k,z}$$

for $z \in B$, where $HS()$ is the heavy-side function defined by

$$HS(X) = \begin{cases} 1 & \text{if } X > 0 \\ 0 & \text{if } X \leq 0. \end{cases}$$

Proof First, we get the linear approximation of the infinite population genetic system in the neighbourhood of the point $\mathbf{x} \in N\Lambda_{3q}$. Then we notice that $\mathbf{x}$ is an attractor if and only if the Jacobian matrix of the linearized system has eigenvalues with modulus less than one. Since the Jacobian of the linearized system can be decomposed as the direct sum of three-order blocks, the result follows by using the Laplace's expansion and examining separately each of the cases $\mathbf{x}_k = (0,0,0), (0,0,1), (0,1,0), (1,0,0)$.

VI. Neural Network

By Theorem 4 we are able to design a discrete deterministic network whose equilibrium points, for arbitrary (positive) fitness, include the attractors of System (6). This aim has been already obtained in [2], [3], [4] for a simpler univariate genetic system whose attractors have essentially been proved to be the equilibrium points of a discrete Hopfield's network with sequential updating and having as energy function the fitness. In this context the equivalence seems to be less satisfactory in the sense that for some unstable equilibrium points $\mathbf{x} \in N\Lambda_{3q}$ with $\mathbf{x}_k = \mathbf{0}$ (for some $k \in \{1,\ldots,q\}$), the behaviour of the associated discrete network cannot accordingly be derived by the result in Theorem 4. However, as far as we are interested in maximizing the fitness, it is reasonable to take as updating rule what locally maximizes the fitness. Thus, we define the discrete deterministic network associated to the infinite population bivariate marginal distribution genetic system having as states the points $\mathbf{x} \in N\Lambda_{3q}$ and with sequential updating equations

$$x_{k,z}(t+1) = \prod_{a\in A-\{z\}} HS(b_{k,z}(\mathbf{x}) - b_{k,a}(\mathbf{x})) \quad \text{for } z \in B \tag{8}$$

if it holds that

$$\prod_{a\in A-\{z\}} HS(b_{k,z}(\mathbf{x}) - b_{k,a}(\mathbf{x})) = 1$$

for some $z \in A$ and

$$\mathbf{x}_k(t+1) = \mathbf{x}_k \tag{9}$$

otherwise, where $\mathbf{x}_k \in \{(0,0,1),(0,1,0),(1,0,0)\}$ is the choice that locally maximizes the fitness in $\mathbf{x}(t)$ with $\mathbf{x}_k(t) = \mathbf{x}_k$ $(1 \leq k \leq q)$. The following theorem states that the fitness function is a Lyapunov function for the iterative discrete network; consequently, we are able to derive an approximation algorithm to maximize the fitness over $N\Lambda_{3q}$.

Theorem 5: For every $\mathbf{x}(t) \in N\Lambda_{3q}$ it holds that

$$\Delta Pf(\mathbf{x}(t)) = Pf(\mathbf{x}(t+1)) - Pf(\mathbf{x}(t)) > 0$$

if $\mathbf{x}(t+1)$ is obtained by (8) and $\Delta Pf(\mathbf{x}(t)) \geq 0$ otherwise $(\mathbf{x}(t+1)$ is computed by (9)).

Proof The result follows by inspection of the cases $(\mathbf{x}_k(t+1) = \mathbf{0}) \wedge (\mathbf{x}_k(t) = \mathbf{0})$, $(\mathbf{x}_k(t+1) = \mathbf{0}) \wedge (\mathbf{x}_k(t) \neq \mathbf{0})$, $(\mathbf{x}_k(t+1) \neq \mathbf{0}) \wedge (\mathbf{x}_k(t) = \mathbf{0})$ and eventually $(\mathbf{x}_k(t+1) \neq \mathbf{0}) \wedge (\mathbf{x}_k(t) \neq \mathbf{0})$.

VII. Application to the Max-Cut Problem

The topic of designing approximation algorithms based on infinite population genetic models (with simulated crossover of one-bit genes) has been studied in [2], [3], [4] for hard problems. In summary, the available main theoretical results about such models evidence that the fitness function becomes a Lyapunov function for asynchronous variants of the corresponding iterative dynamical systems. As a consequence of this property, it is conceivable to design univariate marginal distribution genetic algorithms that can be used as approximation algorithms to solve combinatorial optimization problems. Other results connecting such genetic systems with Hopfield's Networks (well known local optimizers on which some approximation algorithms are based) can be found in [2], [3]. The usual way of using the infinite population systems to get local optimization of the fitness is to initialize with (slightly perturbed) equally likely marginal probability distributions for each allele (see also [2], [3] and in particular [4]). In the experiments, the synchronous variants of the considered systems have exhibited convergence and optimization properties very similar to those shown by the asynchronous systems. Following the guidelines of references [2], [3], [4], [5], System (6) has been implemented to solve (in the sense of an approximation algorithm) the Max-Cut problem, namely, that of partitioning the vertices of an arbitrary graph G in two other subsets V_1 and V_2 in such a way that the number $\omega(V_1)$ of edges with one endpoint in V_1 and the other in V_2 is maximized. We remind that the decision version of the Max-Cut problem is NP-Complete [17]. In the genetic algorithm, we have considered the quadratic fitness $f : \Omega \longrightarrow \mathbf{N}^+$ defined by

$$Qf(x_1,\ldots,x_l) = \sum_{\substack{i,j=1 \\ i\neq j}}^{l} w_{i,j}x_i(1-x_j),$$

where the weights $w_{i,j}$ are set $w_{i,j} = 1$ if the input graph G has edge $\{i, j\}$ and $w_{i,j} = 0$ otherwise. We remark that initial equally likely marginal probability distributions for each allele are fixed points, in case of undirected graphs, for the system based on recombination of sequences of one-bit genes; however, in [2], [3] it has been shown that such points are not asymptotically stable. Conversely, for the system based on simulated crossover of sequences of two-bit genes, equally likely marginal distributions are not fixed points, but they initialize trajectories that do not converge towards states in Ω. Nevertheless, the initial slight perturbations ($\psi_{k,z}(0) = \frac{1}{4} \pm \gamma$, where $\gamma \approx 10^{-2}$, for $k = 1, \ldots q$ and $z \in B$) consent to solve this problem. In our experiments, the genetic algorithm with simulated crossover of two-bit genes exhibited convergence characteristics very similar to those held by the univariate marginal distributions genetic algorithm.

In Table $VII.1$ there are the experimental results intended to compare the performance of the algorithm **2BGSC** based on simulated crossover of sequences of two-bit genes with the **2DNN** based on the corresponding discretized (neural network) system initialized with random points and with sequential updating. In the table it is reported the mean size of the cuts found by the two algorithms for $p-$random graphs with $p = \frac{1}{7}$ and $p = \frac{1}{4}$. In the table it is also reported the expected number of edges (row Edges) of the $p-$random graphs. The performance of the algorithm **2BGSC** is slightly better (in average) than that of the **2DNN** algorithm conversely to what happened in the univariate case [2] in which the infinite population genetic algorithm, in case of $p-$random graphs for $p = \frac{1}{7}, \frac{1}{4}$, exhibited performance slightly worse than that of an Hopfield's network [18]. By completeness, in Table $VII.1$ we report the results of the simulations intended to compare the performance of the algorithm **2BGSC** with the univariate **1BGSC** based on recombination of sequences of binary genes introduced in [2]. As we have already noticed elsewhere, the algorithm **2BGSC** consents to improve (in average) the performance of the algorithm **1BGSC**. Moreover, by giving a look at the mean sizes of the cuts obtained by choosing the best ones, for a same $p-$random graph, found by the two algorithms (rows **12BGSC**), we note that the performance was dependent on specific generated $p-$random graphs: this fact suggests, as a conjecture, that mutations in positions and/or lengths of genes could be helpful to obtain better results and to speed convergence. In this regard, even if our models are merely computational, such mutations could be explained as having the precise aim of adaptation to environment modifications that, in case of evolutionary processes, are very slow.

Table $VII.1$: Mean size of the cuts found by the algorithms

p/n		34	38	42	46	50
$\frac{1}{7}$	**2DNN**	61,8	75,6	91,9	107,2	126,9
	1BGSC	62,2	76,4	92,6	108,2	127,3
	2BGSC	63,3	76,7	93,1	108,7	128,2
	12BGSC	63,7	77,5	94,1	109,5	128,8
	Edges	80,1	100,4	123,0	147,9	175,0
$\frac{1}{4}$	**2DNN**	96,2	118,9	145,0	175,5	201,0
	1BGSC	96,7	119,8	145,3	176,1	201,3
	2BGSC	97,2	120,6	146,6	177,1	204,2
	12BGSC	97,9	121,4	147,2	178,5	205,9
	Edges	140,3	175,8	215,3	258,8	306,3

VIII. Conclusion

In this paper, we have reviewed and analyzed a bivariate marginal distribution genetic model. Our aim is oriented to extend the analysis of univariate marginal distribution genetic algorithms for infinite populations to the bivariate framework. The choice of the bivariate model is for sake of conciseness and simplicity both in the exposition and in technical details. By a first preliminary analysis, we conjecture that the results exhibited in this paper can quite straightforwardly be translated into the multivariate framework. However, further research is required to better understand how to provide a suitable more general model for marginal distribution genetic algorithms (some more recent insigths are addressed in the Estimation of Distribution Algorithms literature and the related research lines, see [26], [30] for an introduction). We are convinced that an important topic in this scenario is constituted by the possibility of extending the relation found by (infinite population) univariate marginal distribution genetic systems and discrete Hopfield's networks. Such extension is considered to be of particular relevance to answer to some crucial questions such as the meaningfulness of genes codification, how to improve performance and, in particular, the design of new models that could be more convenient to try to solve hard optimization problems (in this regard the reader is also referred to the area of classic approximation algorithms with special attention to the Goemans and Williamson results [12]).

Acknowledgment

This paper is memory of my father Mario Carpentieri.

References

[1] S. Baluja and R. Caruana, Removing the Genetics from the Standard Genetic Algorithms, Proceedings of the Twelfth International Conference on Machine Learning, pp. 38–46, Morgan Kaufman, San Francisco, California.

[2] A. Bertoni, P. Campadelli, M. Carpentieri, G. Grossi, A Genetic Model and the Hopfield Networks, Icann96: International Conference on Artificial Neural Networks, Lectures Notes in Computer Science, pp. 463 – 468, Springer-Verlag, Berlin, Germany, 1996.

[3] A. Bertoni, P. Campadelli, M. Carpentieri, G. Grossi, Analysis of a Genetic Model, ICGA97: Proceedings of the Seventh International Conference on Genetic Algorithms, pp. 121 – 126, Morgan-Kaufmann, San Mateo, California, 1997, (also presented at Nips96: Neural Information Processing Systems Workshop, Snowmass, Colorado, December, 1996).

[4] A. Bertoni, P. Campadelli, M. Carpentieri, G. Grossi, A Genetic Model: Analysis and Application to MAXSAT, Evolutionary Computation, pp. 291 – 309, Vol. 8, n. 3, Massachusetts Institute of Technology, Mit Press, Cambridge, MA, 2000.

[5] M. Carpentieri, A Bivariate Marginal Distribution Genetic Model, CEC2006: 2006 IEEE Congress on Evolutionary Computation, pp. 1016 – 1022, Vancouver, Canada, 2006.

[6] M. Carpentieri, A Genetic System Based on Crossover of Two-Bit Genes, Theoretical Computer Science, Vol. 367, n. 3, pp. 324 – 335, 2006 (available online at www.sciencedirect.com).

[7] S. A. Cook, The Complexity of Theorem Proving Procedures, Proceedings of the Third Symposium of the ACM on the Theory of Computing, pp. 151 – 158, *ACM* Press, New York, *NY*, 1971.

[8] Y. Davidor Epistasis Variance: A Viewpoint on GA-Hardness, Foundations of Genetic Algorithms, G. J. E. Rawlings Editor, pp. 23 – 35, San Mateo, CA: Morgan Kaufmann, 1991.

[9] A. E. Eiben, E. H. L. Aarts, K. M. van Hee, Global Convergence of Genetic Algorithms: a Markov Chain Analysis, Parallel Problem Solving from Nature, pp. 4 – 12, H. P. Schwefel and R. Männer Editors, Berlin and Heideberg: Springer, Berlin, 1991.

[10] D. E. Goldberg, Genetic Algorithms in Search, Optimization and Machine Learning, Addison-Wesley, Reading, *MA*, 1989.

[11] D. E. Goldberg, P. Segrest, Finite Markov Chain Analysis of Genetic Algorithms, Proceedings of the Second International Conference on Genetic Algorithms and their Applications, pp. 1–8, J. Grefenstette Editor, Lawrence Erlbaum Associates, Cambridge, Massachusetts, 1987.

[12] M. X. Goemans, D. P. Williamson, Improved Approximation Algorithms for Maximum Cut and Satisfiability Problems Using Semidefinite Programming, Journal of the Association for Computing Machinery, Vol. 42, n. 6, 1995.

[13] G. R. Harik, F. G. Lobo, D. E. Goldberg, The Compact Genetic Algorithm, Technical Report 97006, Illinois Genetic Algorithm laboratory (illiGAL), University of Illinois, Urbana Illinois, 1997.

[14] J. Håstad,Some Optimal Inapproximability Results, Proceedings of the Twenty-Ninth ACM Symposium on the Theory of Computation, pp. 1 – 10, *ACM* Press, New York, *NY*, 1997.

[15] J. H. Holland, Induction, Processes of Inference, Learning and Discovery, *MIT* Press, Cambridge, 1989.

[16] J. H. Holland, Adaptation in Natural and Artificial Systems, *MIT* Press, Cambridge, 1992.

[17] J. E. Hopcroft and J. D. Ullman, Introduction to Automata Theory, Languages and Computation, Addison-Wesley, 1979.

[18] J. J. Hopfield, Neural Networks and Physical Systems with Emergent Collective Computational Abilities, Proceedings of the National Academy of Sciences, pp. 2554 – 2558, 1982.

[19] J. R. Koza, Genetic Programming, MIT Press, Cambridge, *MA*, 1992.

[20] P. Larranaga, J. A. Lozano, Estimation of Distribution Algorithms: a New Tool for Evolutionary Optimization, Kluwer Academic Publishers, Boston, 2001.

[21] S. Lipschutz, Theory and Problems of Linear Algebra, Mc Graw-Jill Books Company, 1968.

[22] K. Marton, A Concentration-of-Measure Inequality for Contracting Markov Chains, Geometric and Functional Analysis, Vol. 6, pp. 556 – 571, Birkhuser, Basel, 1996.

[23] A. Mendiburu, J. A. Lozano, J. Miguel-Alonso, Parallel Implementation of EDAs Based on Probabilistic Graphical Models, *IEEE* Transactions on Evolutionary Computation, Vol. 9, n. 4, 2005.

[24] H. Mühlenbein, H. M. Voight, Gene Pool Recombination in Genetic Algorithms, Proceedings of the Metaheuristics International Conference, I. Osman and J. Kelly Editors, Kluwer, Boston, Massachusetts, 1996.

[25] H. Mühlenbein, The Equations for the Response to Selection and its Use for Prediction, Evolutionary Computation, Vol. 5 (3), pp. 306 – 346, 1998.

[26] H. Mühlenbein, R. Höns, The Estimation of Distributions and the Minimum Relative Entropy Principle, Evolutionary Computation, Vol. 13, n. 1, pp. 1 – 27, 2005.

[27] H. Mühlenbein, T. Mahnig, Evolutionary Algorithms: from Recombination to Search Distributions, Theoretical Aspects of Evolutionary Computation, pp. 137–176, L. Kallel, B. Naudts and A. Rogers Editors, Springer-Verlag, 2000.

[28] B. Naudts and L. Kallel A Comparison of Predictive Measures of Problem Difficulty in Evolutionary Algorithms, IEEE Transactions on Evolutionary Computation, Vol. 4 (1), pp. 1 – 15, 2000.

[29] A. Nix and M. D. Vose, Modeling Genetic Algorithms whit Markov Chains, Annals of Mathematics and Artificial Intelligence, Vol. 5, pp. 79 – 88, 1992.

[30] M. Pelikan, H. Mühlenbein, The Bivariate Marginal Distribution Genetic Algorithm, Advances in Soft Computing - Engineering Design and manufacturing, Springer-Verlag, London, pp. 521 – 535, 1999.

[31] Y. Rabynovich, A. Wigderson, An Analysis of a Simple Genetic Algorithm, Proceedings of the Fourth International Conference on Genetic Algorithms, pp. 215 – 221, Rick Belew and Lashon Booker Editors, Morgan Kaufman, San Mateo, *CA*, 1991.

[32] S. Rochet, Epistasis in Genetic Algorithms Revisited, Information Sciences, Vol. 102, pp. 133 – 155, 1997.

[33] G. Syswerda, Simulated Crossover in Genetic Algorithms, Proceedings of the Second Workshop of Foundations of Genetic Algorithms, D. Whitley Editor, pp. 239 – 255, Morgan Kaufmann, San Mateo, California, 1993.

[34] V. Vapnik, The Nature of Statistical Learning Theory, Springer Verlag, New York, 1995.

[35] H. Voight, Mühlenbein, Gene Pool Recombination and the Utilization of Covariances for the Breeder Genetic Algorithm, Proceedings of the Second *IEEE* International Conference on Evolutionary Computation, pp. 172 – 177, Z. Michalewitch Editor, *IEEE* Press, New York, 1995.

[36] M. D. Vose, Modeling Simple Genetic Algorithms, Evolutionary Computation, Vol. 3, 453 – 472, 1996.

[37] M. D. Vose, The Simple Genetic Algorithm: Theory and Foundations, *MIT* Press, Cambridge, *MA*, 1999.

[38] J. H. Rowe, R. Poli, C. R. Stephens, A Fixed Point Analysis of a Gene Pool *GA* with Mutation, Proceedings of GECCO, pp. 642 – 649, Morgan-Kaufmann, 2002.

[39] Q. Zhang, On the Convergence of a Class of Estimation of Distribution Algorithms, *IEEE* Transactions on Evolutionary Computation, Vol. 8, n. 2, 2004.

[40] Q. Zhang, On Stability of Fixed Points of Limit Models of Univariate marginal Distribution Algorithm and Factorized Distribution Algorithm, *IEEE* Transactions on Evolutionary Computation, Vol. 8, n. 1, 2004.