# USING INFORMATION THEORY TO ANALYZE AND IMPROVE GENETIC ALGORITHM PERFORMANCE

FRANCISCO M. ASSIS, EDMAR C. GURJÃO AND BRUNO B. ALBERT*

*Departamento de Engenharia Elétrica, Universidade Federal de Campina Grande, Av. Aprigio Veloso, 882, CEP 58109-970, Campina Grande PB, Brazil

Email: {fmarcos, ecandeia, albert}@dee.ufcg.edu.br

**Abstract—** There are two contributions in this work, the first one is the presentation of the Vose (Vose, 1999) model for the behaviour of random search algorithm under the point of view of the method of types (Csiszár and Kórner, 1981), the second one is the use of algebraic code with genetic algorithms (GA) in order to speedup their convergence and precision in solutions.

**Keywords—** Simple Genetic Algorithm (SGA), Infinite Population Model, Markov Chain (MC)

**Resumo—** Este trabalho apresenta duas contribuições, a primeira é a apresentação do modelo de Vose (Vose, 1999) para o comportamento de algoritmo de busca aleatória sob o ponto de vista do método dos tipos (Csiszár and Kórner, 1981). A segunda é o uso de códigos algébricos em algoritmos genéticos (GA) para aceleração da convergência e aumento da precisão das soluções.

**Palavras-chave—** Algoritmos Genéticos Simples, Modelo de População Infinita, Cadeia de Markov

## 1 Introduction

Random Heuristic Search (RHS) stands for a large class of algorithms that can be classified by the heuristic procedures. For example, the well known Simple Genetic Algorithm (SGA)(Holland, 1975) is a sort of RHS whose heuristic is based on mimic of biological processes of selection, mutation and crossover. There is a number of variations of this SGA paradigm, we denote GA any of such variations. In general such that variations are proposed to overcome some setbacks on the SGA performance like premature convergence.

Inspite of setbacks, SGA works very well in a number of optimization problems and many authors have proposed theories in order to explain how and why SGA efficacy. The most successful of these models make use of Markov chains being due to Vose and others (Vose, 1999). The Vose model is the most general and it is applicable to RHS whose machinery appears to be a discrete dynamical system on a certain simplex through identification of populations with corresponding population types. The first contribution of this paper is to interpret Vose model in the framework of the method of types due to Csiszár (Csiszár and Kórner, 1981).

The SGA can be conceived by specifying the search space $\Omega \subset \mathbb{Z}_2^k$ and defining the heuristic procedure as the repetition of selection, mutation and crossover operations over elements of a given subset $P \subset \Omega$. But there is no structure defined for the search space. This lack of structure was recently identified as one of the the causes for the diversity loss by Bryden et all (Willson, 2006). Diversity loss makes the algorithm to search at each iteration in smaller and smaller portions of the search space resulting frequent premature convergence. In their work authors explore the use of graphs to limit such loss in order to give to the search space some structure and reduce loss of diversity.

The second contribution of this paper is the application of algebraic codes as an alternative technique to keep diversity, speeding up and improving solutions precision of GAs. The mechanism is to define the search space as the linear subspace defined by an algebraic code, performing genetic operations over the information words and selection operation on the codewords.

The paper is organized as follows. In Section 2 we present the relation of the Method of Types and RHS. In Section 3 we present the application of algebraic codes in the SGA. Finally, in Section 4 we present some conclusions.

## 2 Method of Types and RHS

For $t = 0, 1, \ldots$ a population is a ordered multiset denoted by $\mathbf{X}^t = \{X_0,\ X_1, \ldots, X_{r-1}\}$ where $r$ is the size of the population and each $X_i,\ i = 0, \ldots,\ r-1$ is chosen i.i.d. under some probability distribution (PD) from the search space $\Omega = \{0,\ 1, \ldots, n-1\}$. Let $k(x|\mathbf{X}^t)$ the number of copies of individual $x \in \Omega$ in the population $\mathbf{X}^t$, define $p_i^t(\mathbf{X}) = k(i|\mathbf{X}^t)/r$ be the proportion of $i$th individual. The vector $\mathbf{p}(\mathbf{X}^t) = \left(p_0^t, \ldots, p_{n-1}^t\right)^T$ is called **population type** $\mathbf{X}^t$. If it is clear from the context we will omit the superindexes and the symbol $\mathbf{X}$ from $\mathbf{p}(\mathbf{X}^t)$.

Let $\mathcal{P}_r$ denote the set of size $r$ population types with individuals selected from the search space $\Omega$. Observe that $\mathcal{P}_r$ is a proper subset of the n-simplex

$$\Lambda = \left\{\mathbf{p} \in \Re^n : \sum p_i = 1,\ p_i \geq 0\right\}.$$

If $\mathbf{p} \in \mathcal{P}_r$, the set of populations of size $r$ and empirical distribution $\mathbf{p}$ is called **composition class** of $\mathbf{p}$ defined by $T(\mathbf{p}) = \{\mathbf{X} \in \Omega^r : \mathbf{p}(\mathbf{X}) = \mathbf{p}\}$.

We shall compare facts from the method of types (Csiszár and Kórner, 1981) usually applied in the framework of communications engineering with results of the Vose model proposed in the framework of RHS algorithms. We remark that the term

**population** in RHS jargon is correlate to the word **sequence** in communication engineering jargon and similar comparison correlate term **search space** with term **source alphabet**, respectively in RHS and communication contexts. However an important distinction is that population size is small compared with usual sequences size and search space is large compared with usual source alphabets. More precisely we set $n \in \mathcal{O}\left(2^r\right)$, where $r$ is around a couple of tenths. Below we recall facts from the method of types.

**Fact 1:** In the context of communications there are relatively few composition classes and in consequence there are composition classes with a huge number of sequences. Universal source coding theorems are based on this fact.

$$|\mathcal{P}_r| = \binom{r+n-1}{n-1} \leq (r+1)^n \tag{1}$$

In the context of RHS algorithms $n \in \mathcal{O}(2^r)$ so there a huge number of composition classes. It is not expected that the use of techniques inspired on universal source coding to be profitable.

**Fact 2:** Population $\mathbf{X}^t$ is a random sequence. Denote a specific realization of such that random sequence by $\mathbf{x}^t$, or $\mathbf{x}$ if $t$ it is clear from the context. Let $\mathbf{q}(\mathbf{x})$ the probability of a fixed sequence $\mathbf{x} = \{x_0,\ x_1, \ldots, x_{r-1}\}$ to be selected i.i.d. under a PD $\mathbf{q}$, then $\mathbf{q}(\mathbf{x})$ depends only on its type $\mathbf{p}(\mathbf{x})$ and is given by

$$\mathbf{q}(\mathbf{x}) = 2^{-r(H(\mathbf{p})+D(\mathbf{p};\mathbf{q}))} \tag{2}$$

where $H()$ is the entropy function and $D()$ divergence between distributions $\mathbf{p}$ and $\mathbf{q}$. Remark that in the context of communications $r$ correspond to length of sequences and entropy and divergence are small compared to $r$, however in the context of RHS both entropy and divergence are comparable with $\log r$. This is because inspite of types have $n$ entries only $r$ of them can be nonzero.

**Fact 3:** For any type $\mathbf{p} \in \mathcal{P}_r$ and any PD $\mathbf{q}$, the probability of a composition class $T(\mathbf{p})$ under $\mathbf{q}$ satisfies

$$\frac{2^{-rD(\mathbf{p};\mathbf{q})}}{(r+1)^n} \leq \mathbf{q}\left(T(\mathbf{p})\right) \leq 2^{-rD(\mathbf{p};\mathbf{q})}. \tag{3}$$

**Fact 4:** For any type $\mathbf{p} \in \mathcal{P}_r$,

$$\frac{2^{rH(\mathbf{p})}}{(r+1)^n} \leq |T(\mathbf{p})| \leq 2^{rH(\mathbf{p})}. \tag{4}$$

A basic component of any RHS is the so called **transition rule**, $\tau : \Lambda \to \Lambda$. It is iterated to generate populations $\mathbf{X}^0 \xrightarrow{\tau} \mathbf{X}^1 \xrightarrow{\tau} \ldots$ Given a current population type $\mathbf{p}$ the next population type $\tau\left(\mathbf{p}\right)$ results of two maps, one of them is denoted **heuristic function** or simply **heuristic** is deterministic and the other one is denoted **sampling** or **selection** is a nondeterministic map. Such composition is illustrated schematically in Fig. 1. Vose model interpret the RHS machinery as a

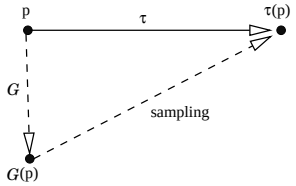


Figure 1: General transition rule $\tau$ for any composed of deterministic heuristic function $\mathcal{G}$ and random sampling. Note sampling is made i.i.d. under DP $\mathcal{G}(p)$

dynamical system on the simplex $\Lambda$ through identification of populations with corresponding types. But to the authors knowledge, Vose papers do not mention the method of types. An important question is what is the probability of the composition class the next population will be within. Note that the information on the problem is the same to any population from a specific composition class.

In order to illustrate the interpretation of Vose results by the method of types we take the probability that $\mathbf{q} \in \mathcal{P}_r$ is the next population type given that $\mathbf{p}$ is the current population type. Vose derives the exact formula $\Pr[\mathbf{q}] = r! \prod \mathcal{G}(p_j)^{rq_j}/(rq_j)!$ to this probability, but this is hard to manipulate due to the presence of factorials and there is no informational measure inserted.

**Proposition 1**

$$\frac{2^{-rD(\mathbf{q};\mathcal{G}(\mathbf{p}))}}{(r+1)^n} \leq \Pr[\mathbf{q}] \leq 2^{-rD(\mathbf{q};\mathcal{G}(\mathbf{p}))}. \tag{5}$$

To verify the proposition above begin with the Vose formula, apply ( 4) to the factorials and ( 2) to the product $\mathcal{G}(p_j)^{rq_j}$, then after some simplification the result follows. Note that $\Pr[\mathbf{q}] = \Pr[T(\mathbf{q})]$, the probability of sample a population $\mathbf{X} \in T(\mathbf{q})$, falls exponentially at rate given by the divergence $D(\mathbf{q};\mathcal{G}(\mathbf{p}))$. Hence depending on heuristic function $\mathcal{G}()$, that generation to generation the population can concentrates near populations with type given by $\mathcal{G}(\mathbf{p})$ with fast loss of diversity. This characterize the frequent effect of premature convergence.

In fact, probability $\mathbf{q}$ depends only on the current one $\mathbf{p}$, so the sequence of *random vector* $\mathbf{p}$, $\tau(\mathbf{p})$, $\tau^2(\mathbf{p})\ldots$ can be viewed as an Markov chain whose huge transition probabilities matrix are such that shown in ( 5), explicitly: $\mathbf{P}_{p,q} = \Pr[\mathbf{q}]$. From the comments above it is clear the great importance of the heuristic function $\mathcal{G}\left(\right)$ to the performance of RHS algorithms. In fact we can say that $\mathcal{G}\left(\right)$ *defines* the RHS. We remark so that RHS algorithms are full characterized by its heuristic function that determine the kind of paths they will follow in the simplex $\Lambda$.

**Input:** objective-function, $f(v) : \mathcal{D} \rightarrow \mathbb{R}, \quad \mathcal{D} \subset \mathbb{R}^m$, linear algebraic code $(l, k)$ with generator matrix $G$
**Output:** optimum domain value $v^\star$
**Initialization:**

- $t \leftarrow 0; \quad \mathbf{X}(t) \subset_R \{0,1\}^k; \quad$ evaluate $\mathbf{X}(t)\, G;$

**Iteration:**
**WHILE** STOP = FALSE **DO**

- $\mathbb{Z}^k \supset \mathbf{X}'(t) \leftarrow$ variation: crossover, mutation $\mathbf{X}(t)$;
- evaluate $\mathbf{X}'(t)\, G;$
- $\mathbf{X}(t+1) \leftarrow$ selection $\mathbf{X}'(t); \quad t \leftarrow t+1;$

**END**

## 3 Algebraic Code-Based Genetic Algorithm (CBGA)

The pseudo-code of the CBGA shown in the Algorithm above is inspired in (de Assis, 1997) that established an association between the SGAs performance parameters thoroughness and sparsity with covering and packing radii parameters of algebraic codes. The idea is to explore the geometric structure of the algebraic codes in order to overcome premature convergence and speed up convergence of SGAs.

A code is defined as a linear map $G : \mathbb{Z}_2^k \rightarrow \mathbb{Z}_2^l$ where $\mathbb{Z}_2$ is the binary finite field (Blahut, 1983). As an example consider the well known Hamming (7,4) code, whose map $G$ is given by the matrix below and the relation betwen a codeword $c$ and a information word $i$ is $c = iG$ with operations in $GF(2)$, e.g. the information $i = 1000$ is mapped to $c = 1000101$.

$$G = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}$$

The main difference between SGA and CBGA concerns to the search space size. Search space is now a subset defined by the code. Genetic operations mutation and crossover are performed with elements from $\mathbb{Z}_2^k$, after that each element of $\mathbf{X}'$ is multiplied by the generator matrix of the code to obtain the set $\mathbf{X}'G \subset \mathbb{Z}^l$ whose elements have $l-$ bit binary representation. As it is well known the domain $f()$ domain is quantized in $2^l$ pieces. The procedure to compute fitness is therefore performed with precision of $l$ bits.

Crossover is the operation that combine information from individuals that are submitted to selection step, this is necessary to find the optimal element, but crossover is also responsible by diversity loss from one generation to the next, what characterize the GA premature convergence. Mutation is utilized to combat such that setback, but other measures can be necessary, e.g., recently Bryden et all (Willson, 2006) propose control the speed of convergence intermediating crossover by the use of graphs. Note that the aim is giving time to competing solutions mature. CBGA seeks attain the same objective but work by means the weight structure of the algebraic code that assure the Hamming distance between elements of a population submitted to selection to be greater than the minimum distance of the code.

We have applied CBGA and GA to the test generalized Rastrigin function, describe in Equation 6 and plotted in Figure 2 to $m = 2$ and $-2 \leq v_i \leq 2$, chosen from 23 functions benchmark table presented by Yao (Liu, 1999). This class of multimodal functions is one of the 23 found in the table whose number of local minima increases exponentially with the dimension $m$. Note that they appear to be the most difficult class of problems for many optimization algorithms.

$$f(v_1, \ldots, v_m) = \sum_{i=1}^{m} \left(v_i^2 - 10\cos(2\pi v_i) + 10\right), \quad -7.75 \leq v_i \leq 7.75 \quad (6)$$

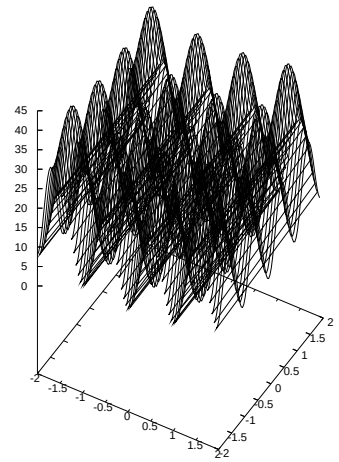


Figure 2: Rasting function to $m = 2$ and $-2 \leq v_i \leq 2,\ i \in 1, 2$

In all steps, that is selection, mutation and crossover operations, GA has its individuals (chromosomes) represented with $600-$ bit integers . This means that are 6 bits per dimension for GA. For CBGA genetic operations must be performed with information symbols of the linear code, so if $R$ is the code rate individuals must be represented with approximately $R \times 600$ -bit integers, that is, 6 bits per argument $v_i,\ i = 1, \ldots 100$. We adopt the Golay $(23, 12, 7)$ to our test because it is a well known binary code with nice combinatorial properties. Hence to make genetic operations the individuals must be represented approximately $12/23 \times 600$-bit integers. We assume 312-bit to that representation. After ge-

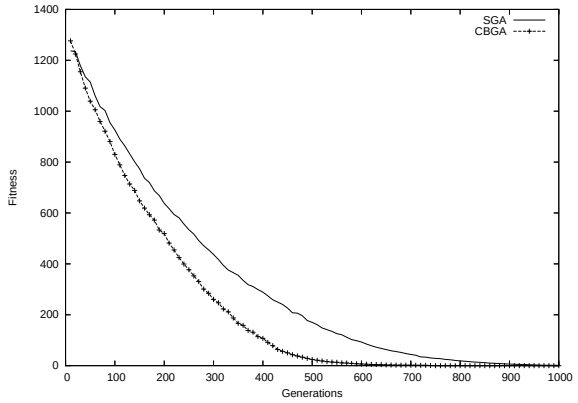


Figure 3: Convergence comparison CBGA and GA

netic operations these 312 bits are mapped into 598-bit integers that are the submitted to selection step of the algorithm. Note that $598 = 26 \times 12$ meaning that in fact 26 "information words" are encoded. With dimension $m = 100$ 312 bits are distributed to represent $v_i s$. Hence $v_i s$ are represented by $3.12$ bit on average.

Fig.( 3) displays that as expected the CBGA converges faster than the GA. The number of generations to attain the minimum is near 64% compared to the number required by the GA.

As the mutation probability must be adjusted to each problem problem, for example to the values of Figure 3 it was 0,01%, a new experiment with both the SGA and CBGA algorithms with zero mutation probability was performed and the results ~~where~~ **are** presented in Figure 4, with population size (PS) 500 and 600 to CBGA and 600 to SGA. As expected, without mutation the premature convergence occur. But it is worthy note that CBGA gives a better result than SGA. It is a topic for future research to analyze if the CBGA can avoid the use of the mutation probability.

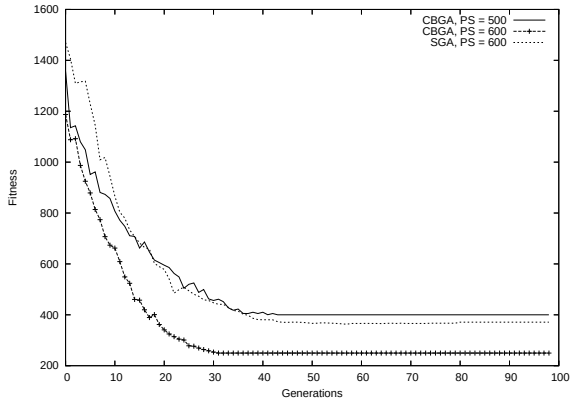


Figure 4: Convergence comparison CBGA and GA with mutation probability equals to zero .

## 4 Summary

In this paper we have compared the Vose model for RHS algorithms under the point of view of the method of types. In particular we have shown that transition probability $\mathbf{P}_{q,p}$ is dependent only of the heuristic function. Another contribution of this is the proposition of an simple modification of GAs by means utilization of algebraic codes to combat premature convergence of GAs. The new algorithm performance is compared with a GA by application both to a well known benchmark problems. Future directions includes comparing CBGA with a recent graph-based technique proposed to fight premature convergence in GAs.

## References

Blahut, R. E. (1983). "Theory and Practice of Error Control Codes", Addison-Wesley Publishing Company, Inc., Owego, NY.

Csiszár, I. and Kórner, J. (1981). Information Theory: Coding Theorems for Discrete Memoryless Systems, Academic Press, New York, USA.

de Assis, F. M. (1997). "Genetic Algorithms and Packing of Block Codes", Proceedings of the International Conference on Telecommunications - ICT97, Vol. 3, Melbourne, Australia, pp. 1045–1048.

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

Liu, G. L. X. Y. Y. (1999). "Evolutionary Programming Made Faster", IEEE Transactions on Evolutionary Computation, Vol. 3, pp. 82–102.

Vose, M. D. (1999). The Simple Genetic Algorithm, MIT Press, Cambridge, Massachusetts.

Willson, K. M. B. D. A. A. S. J. (2006). "Graph-Based Evolutionary Algorithms", IEEE Transactions on Evolutionary Computation, Vol. 10, pp. 550–567.