Ninth Int. Workshop on GNs, Sofia, 4 July 2008, 59-63

A GENERALIZED NET MODEL OF A MUTATION OPERATOR FOR THE BREEDER GENETIC ALGORITHM

O. Roeva[1], A. Shannon[2]

[1]Centre of Biomedical Engineering – Bulgarian Academy of Sciences
105, Acad. G. Bonchev Str., Sofia 1113, Bulgaria
E-mail: *olympia@clbme.bas.bg*

[2]Raffles College of Design and Commerce, North Sydney, 2060, Australia
Warrane College, University of New South Wales, Kensington, 1465, Australia
E-mails: *tonySHANNON@raffles.edu.au*, *tony@warrane.unsw.edu.au*

Abstract: A generalized net model of the mutation operator for the genetic algorithm is developed. The apparatus of generalized nets is considered as an appropriate tool for describing the performance of the genetic algorithm. The proposed generalized net model is a realization of the mutation operator for the Breeder genetic algorithm. The resulting GN model can be considered as a separate net, but also can be aggregated into a generalized net model which describes the whole genetic algorithm performance.

Keywords: Generalized net, Breeder genetic algorithms, Mutation.

Introduction

Evolution of natural organisms is based on three major components – reproduction, variation and selection. Some reproductions of natural organisms occur with "failures" called mutations. A more systematic variation of the genetic material happens in sexual reproduction. Each parent contributes half of its genetic material to the offspring. This method of variation is called recombination. The offspring will be identical to the parents if the parents are genetically equal. Variation is necessary to allow selection to work. Selection in nature is very difficult to define precisely. The term was introduced by Darwin very informally. "The preservation of favourable variations and the rejection of injurious variations, I call Natural Selection". But how can an observer predict which are the favorable variations? The favorable variations are the variations which are preserved! The variations can only be judged after they have competed in the "struggle for life". Natural selection is no independent force of nature, it is the result of the competition of natural organisms for resources. In contrast, in the science of breeding the above problem does not exist. The selection is done by human breeders. Their strategies are based on the assumption that mating two individuals with high fitness more likely produces an offspring of high fitness than two randomly mating individuals. The Breeder genetic algorithm (BGA) introduced by the author previously [8] is based on the science of breeding. The science is part of applied statistics. A major component is the parent-offspring correlation and the heritability coefficient [5, 6, 8, 9].

The main purpose of this paper is to develop a Generalized Net (GN) model to realize the process of a mutation operator of the BGA. Until now GNs have been used for modelling of parallel processes in several areas [2, 3]. Among them a few GN models regarding GA

performance have been developed [1, 4, 10, 11]. The results show that the apparatus of GNs can be considered as an appropriate tool to model and perform parallel processes such as GAs.

Mutation operator of the Breeder Genetic Algorithm

Mutation is a genetic operator that alters one ore more gene values in a chromosome from its initial state. This can result in entirely new gene values being added to the gene pool. With these new gene values, the genetic algorithm may be able to arrive at better solution than was previously possible. Mutation is an important part of the genetic search as helps to prevent the population from stagnating at any local optima. Mutation occurs during evolution according to a user-definable mutation probability. This probability should usually be set fairly low. If it is set to high, the search will turn into a primitive random search.

The goal of mutation operator is for modifying one or more parameters of z_i (individuals or chromosomes), the modified objects (i.e. offspring) appear in the landscape within a certain distance of unmodified objects (i.e. parents). The mutation operator is defined as follows [7]:

A chromosome x_i is selected with probability p_m for mutation. The BGA normally uses $p_m = 1/n$. At least one variable will be mutated. A value out of an interval [– $range_i$, $range_i$] is added to the selected variable. $range_i$ defines the *mutation range*. It is normally set to $a \cdot searchinterval_i$, where a is a constant. $searchinterval_i$ (upper-bound, lower-bound) is the domain of definition of variable x_i.

The new value z_i (*new chromosome*) is computed according to

$$z_{i\,(NewChrom)} = x_{i\,(OldChrom)} \pm range_i \cdot \delta, \tag{1}$$

where

$$range = 0.5 \cdot \left(upperbound - lowerbound \right). \tag{2}$$

The + or – sign is chosen with probability 0.5 (see Eq. (1) - (2)).$\cdot\delta$ is computed from a distribution which prefers small values. This is realized as follows

$$\delta = \sum_{i=0}^{accur\text{-}1} \alpha_i 2^{-i}, \ \alpha_i \in [0, 1], \tag{3}$$

where

$$\alpha_i = rand\left(accur, 1\right) < 1/accur\ . \tag{4}$$

Before mutation $\alpha_i = 0$ is set. Then each α_i is mutated to 1 with probability $p_\delta = 1/accur$. Only $\alpha_i = 1$ contributes to the sum. On the average there will be just one α_i with value 1 say α_j. Then δ is given by

$$\delta = 2^{-j} \tag{5}$$

The mutation operator is similar in spirit to that used by the parallel GA [9] but the BGA operator is much more easy to understand. Furthermore, it is independent of the location in phenotype space.

The standard BGA mutation operator is able to generate any point in the hypercube with center x defined by $x_i \pm range_i$. But it tests much more often in the neigborhood of x. In Eq. (3), '*accur'* (precision of mutation steps) is a parameter originally related to the machine precision, that is, the numbers of bits used to represent a real variable in the machine we are working with, traditionally there were used values of 8 and 16.

Generalized Net Model

The GN model of mutation operator of the BGA is shown in Fig. 1. The proposed GN model generates a matrix '*Chrom'* with the real representation of the individuals in the current population, mutates the individuals with given mutation probability (p_m) and returns the resulting population (*NewChrom*) – the same number of randomly initialized real valued individuals.

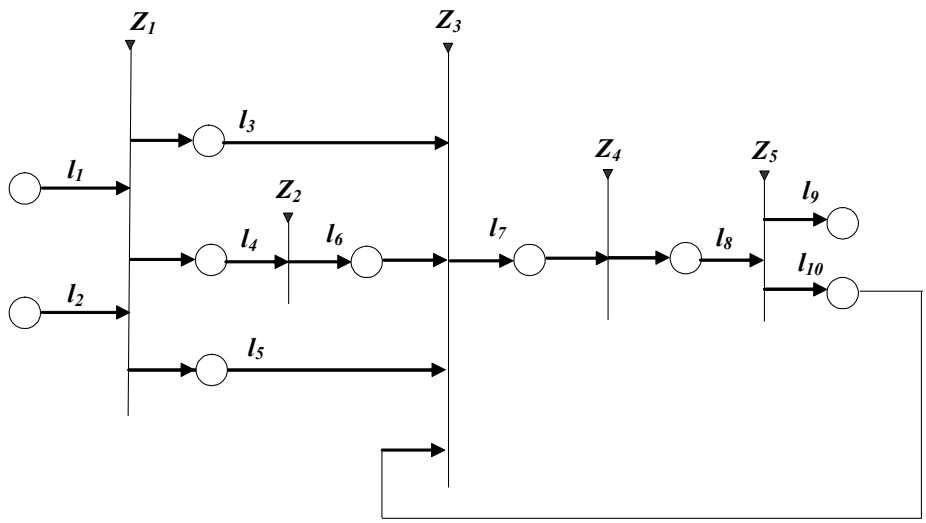

Fig. 1. Generalized net model of mutation operator

The transition Z_1 has the following definition:

$$Z_1 = <\{l_1, l_2\}, \{l_3, l_4, l_5\}, r_1, \vee(l_1, l_2)>,$$

$r_1 =$

	l_3	l_4	l_5
l_1	W_1	*false*	W_3
l_2	*false*	W_2	*false*

where W_1 = "*estimation of the parameter range*"; W_2 = "*evaluation of uniformly distributed random numbers* α_i"; W_3 = "*generation of a matrix Chrom*".

After the transition Z_1 the tokens take on the following characteristics:

- In position l_1 the preliminary parameters are given: individuals number (*nvar*) and matrix of the boundaries of each individual – *upper-bound* and *lower-bound.*

- In position l_2 the initial parameters for mutation operator are given: probability for mutation of a variable (p_m); mutation type - added (+) or subtracted (–) and *accur.*
- In position l_3 the value of the *range* is evaluated (according to Eq. (2)).
- In position l_4 the values of the α_i parameters are evaluated (according to Eq. (4)).
- In position l_5 the matrix *Chrom* is formed.

The form of the transition Z_2 is:

$$Z_2 = <\{l_4\}, \{l_6\}, r_2, \vee(l_4)>,$$

$r_2 =$

	l_6
l_4	W_4

where W_4 = "*δ value evaluation*".

In position l_6 the δ value is obtained (according to Eq. (3)).

The transition Z_3 has the following formal definition:

$$Z_3 = <\{l_3, l_4, l_6, l_{10}\}, \{l_7\}, r_3, \vee(l_3, l_4, l_6)>,$$

$r_3 =$

	l_7
l_3	W_5
l_4	W_5
l_6	W_5
l_{10}	W_6

where W_5 = "*mutation, based on standard BGA mutation operator*"; W_6 = "*mutation, based on standard BGA mutation operator, if boundaries are not satisfied*".

In position l_7 the matrix of *NewChrom* is obtained. The matrix is in the same format as *OldChrom* and containes the chromosomes of the population after mutation.

The next transition Z_4 provides control of the variable boundaries validity, compared to lower and upper boundaries. The form of the transition Z_4 is:

$$Z_4 = <\{l_7\}, \{l_8\}, r_4, \vee(l_7)>,$$

$r_4 =$

	l_8
l_7	W_7

where W_7 = "*control of variable boundaries*".

The form of the transition Z_5 is:

$$Z_5 = <\{l_8\}, \{l_9, l_{10}\}, r_5, \vee(l_8)>,$$

$$r_5 = \begin{array}{c|cc} & l_9 & l_{10} \\ \hline l_8 & W_8 & \neg W_8 \end{array}$$

where W_8 = "*end of mutation process*".

In position l_9 the new chromosome is ready for further examination of the genetic algorithm, if the boundaries of the chromosome are in the appropriate range. Otherwise, if the boundaries of the new chromosome are not in the appropriate range, the result is returned to the transition Z_3 for a new mutation operation.

Conclusion

Using the apparatus of Generalized Nets a GN model has been constructed to realize the mutation operation of a genetic algorithm. Mutation is an important part of the genetic search that prevents the population from stagnation at any local optima. The proposed GN model performs mutation, based on the standard Breeder genetic algorithm mutation operator.

The resulting GN model can be considered as a separate net, but the separate parts can also be accumulated into a composite GN model, which describes the whole genetic algorithm performance.

References

[1] Aladjov H., K. Atanassov, A Generalized Net for Genetic Algorithms Learning, *Proc. of the XXX Spring Conference of the Union of Bulgarian Mathematicians*, Borovets, 2001, 242-249.

[2] Atanassov K., Generalized Nets and Systems Theory, Sofia, Academic Publishing House "Prof. M. Drinov", 1997.

[3] Atanassov K., Generalized Nets, Singapore, New Jersey, London, World Scientific, 1991.

[4] Atanassov K., H. Aladjov, Generalized Nets in Artificial Intelligence, Vol. 2: Generalized nets and Machine Learning, Prof. M. Drinov Academic Publishing House, Sofia, 2000.

[5] Crisan C., H. Mühlenbein, The Breeder Genetic Algorithm for Frequency Assignment, *Lecture Notes in Computer Science*, 1498, 1998, 897.

[6] Montiel O., O. Castillo, P. Melin, R. Sepulveda, Application of a Breeder Genetic Algorithm for Filter Optimization, *Natural Computing: An International Journal Archive*, 4(1), 2005, 11-37.

[7] Montiel O., O. Castillo, R. Sepulveda, P. Melin, Application of a Breeder Genetic Algorithm for Finite Impulse Filter Optimization, *Information Sciences*, 161, 2004, 139-158.

[8] Mühlenbein H., D. Schlierkamp-Voosen, Predictive Model for Breeder Genetic Algorithm, *Evolutionary Computation*, 1, 1993, 25-49.

[9] Mühlenbein H., M. Schomisch, J. Born, The Parallel Genetic Algorithm as Function Optimizer, *Parallel Computing*, 17, 1991, 619-632.

[10] Roeva O., K. Atanassov, A. Shannon, Generalized Net for Evaluation of Genetic Algorithm Fitness Function, *Proceedings of the Eighth International Workshop on Generalized Nets*, Sofia, June 26, 2007, 48-55.

[11] Roeva O., K. Atanassov, A. Shannon, Generalized Net for Selection of Genetic Algorithm Operators, *Proceedings of Scientific Session of the Section "Informatics"*, Union of Scientists in Bulgaria, in press.