Comparison of Major Domination Schemes for Diploid Binary Genetic Algorithms in Dynamic Environments

A. Sima UYAR and A. Emre HARMANCI
Istanbul Technical University
Computer Engineering Department
Maslak TR80626 Istanbul - Turkey
e-mail: {etaner, harmanci}@cs.itu.edu.tr

Abstract*: When working in dynamic environments, it is important for a solution approach to adapt to the change and follow the new optima. For genetic algorithms, using diploid representations is one of the methods for addressing performance and diversity issues in such environments. When using a diploid representation for individuals, the choice of a good genotype to phenotype mapping mechanism is a very important factor. The main aim of this study is to give performance comparisons of some of the existing genotype to phenotype mapping approaches, show their weaknesses and propose a new, adaptive approach that will perform better in the cases where these fail. The proposed approach is compared against the Ng-Wong and the additive diploidy approaches with a dominance change mechanism, using a controlled 0/1 Knapsack problem. All three approaches tested for this study, show similar performances for oscillating change. In the random change cases, the proposed approach shows better performance in automatically adapting to the change.*

Keywords: *genetic algorithms, diploid representations, genotype to phenotype mapping, domination, adaptive dominance values, penetrance, dynamic environments, diversity.*

Introduction

Most studies in application of genetic algorithms deal with static environments. However there is a class of problems where the environment changes in time. From a genetic algorithm point of view, the change may be in the fitness function, in the constraints or the problem instance itself. In such environments, it becomes important for the solution approach to adapt to this change and follow the new optima. Classical genetic algorithms do not have the necessary mechanisms to address the issues encountered when working in such environments. Thus it becomes necessary either to make modifications to existing algorithms or to incorporate other features. The main issues when dealing with dynamic environments is preserving diversity in the gene pool of the population and being able to converge to a solution on the phenotype level. The modifications or the new features have to take this issue into account and incorporate a balance between preserving diversity and converging to a solution.

There are several different approaches in literature which deal with variations of genetic algorithms suitable for working in dynamic environments. Since the choice of a suitable approach depends on analyzing the nature of the change in the environment, it is worthwhile to categorize the change based on specific properties. A good set of criteria for this purpose is given in [1] as follows:

- frequency of change,
- severity of change,
- predictability of change,
- cycle length / cycle accuracy.

Using diploid representations is one of the approaches for addressing performance and diversity issues in dynamic environments. When using a diploid representation for individuals, the choice of a good genotype to phenotype mapping mechanism is a very important factor in performance. In nature this is achieved by the domination property of genes. There has been some research done in the area of artificial domination and discussions and references can be found in literature, some of which are given in [1,2,3,5,6].

The main aim of this study is to give performance comparisons of some of the existing major domination approaches, show their weaknesses and propose a new, adaptive approach that will perform better in the cases where these fail.

Domination Approaches

In this section the proposed domination approach and the major approaches found in literature which will be used in the comparisons will be introduced and their mechanisms will be briefly explained.

The Proposed Approach

The adaptive dominance approach proposed in this study is loosely based on a natural phenomenon known as the *penetrance* of a gene [12]. The main factor for this phenomenon in nature is environmental change. The level of penetrance for a gene may be calculated as the proportion of individuals with a given genotype which exhibit a particular phenotype. Under incomplete penetrance, a dominant trait may not be expressed in the phenotype for a given genotype. This implies that the level of penetrance of a gene may change when environmental conditions are altered.

For the proposed approach in this study, individuals are represented with three binary strings, i.e. two chromosome strings making up the genotype and one phenotype string. A global domination map is used for genotype to phenotype mapping. The length of the map is the same as the length of the chromosomes. Each location on the map may take on a real value in [0.0,1.0]. This value shows the domination factor of the allele 1 over the allele 0 for the corresponding location on the chromosomes. For example if the value for the 5th location on the domination map is 0.7 and the corresponding 5th locations on the chromosomes of an individual are different, the individual will show the allele 1 in its phenotype for the 5th location with probability 0.7 and the allele 0 with probability 0.3. This domination map is re-evaluated at the end of each generation using Equation 1.

$$dom(i) = \frac{\sum_{j} (p_{ij} * f_j)}{\sum_j f_j} \qquad i = 1,...,ChrmLen \;\; and \;\; j = 1,....,PopSize \tag{1}$$

where p_{ij} shows the *i*th phenotypic value of the *j*th individual and f_j shows the fitness of the *j*th individual. All locations on the domination map are initialized to 0.5, meaning that allele 1 and allele 0 are equally dominant in the beginning. Since the domination map is re-calculated in each generation based on the current fitness values of individuals, the adaptation is automatic and it is not required to explicitly detect the change to make the necessary modifications.

Approaches Chosen for Comparisons

There are several diploid approaches in literature and references for these may be found in [1,2,3,4,5,6,8,9].

It has been shown in [7] and [10] that in the case of dynamic environments, some type of a dominance change mechanism is needed. The approaches which do not incorporate this feature do not perform better than a simple, haploid genetic algorithm. Based on these results, the approaches with fixed dominance mechanisms will not be used for comparisons in this study.

In [7], the domination approach proposed by Ng-Wong [8] and the additive diploidy approach proposed by Ryan [9] are modified to incorporate a dominance change mechanism as is explained in the following subsections. The results reported in [7] conclude that both approaches are good for environments where the change oscillates between two solutions with fixed intervals but fail to give the same level of performance for the case where the change instances and severities occur randomly.

Modified Ng-Wong Approach

In the basic Ng-Wong approach [8], there are four alleles, two of which are dominant given by **0** and **1** and the others recessive given by ***i*** and ***o***. In the phenotype, the dominant allele is expressed. The dominance map to obtain the phenotype from the genotype is given in Table 1 where 0/1 means that 0 and 1 are equally likely and the occurrence of 1*i* or 0*o* is prohibited (if it occurrs the recessive gene is promoted to be dominant).

Table 1. Dominance map for Ng-Wong

	0	***o***	**1**	***i***
0	0	0	0/1	0
o	0	0	1	0/1
1	0/1	1	1	1
i	0	0/1	1	1

As given in [7], Ng-Wong allows a dominance change to occur when the fitness of a population member drops by a particular percentage between successive evaluation cycles. Dominance change is achieved by inverting the dominance values of all allele pairs, i.e. 11 becomes *ii*, 00 becomes *oo*, 1*o* becomes *i*0 and 0*i* becomes *o*1.

Modified Additive Diploidy

In the basic additive diploidy approach [9], alleles are represented by ordered values that are combined using a pseudo-arithmetic to obtain the phenotype. In [9], each value is associated with a number (A:2, B:3, C:7, D:9) and addition is performed. If the resulting sum is above a threshold (chosen as 10), the phenotype becomes 1 and if it is below, it becomes 0. The resulting map is given in Table 2.

Table 2. Dominance map for additive diploidy

	A	**B**	**C**	**D**
A	0	0	0	1
B	0	0	0	1
C	0	0	1	1
D	1	1	1	1

In [7], the above approach is modified to incorporate a dominance change mechanism where alleles in the genotype are demoted or promoted by one grade (demoting a B makes it an A and promoting it makes it a C). As in the Ng-Wong approach, dominance change is allowed to occur when the fitness of a population member drops by a particular percentage between successive evaluation cycles. To achieve the dominance change, for each locus one of the genotypic alleles is chosen at random and,

- if the phenotypic expression for this locus is 1, then the chosen genotypic allele is demoted (unless it is an A).
- if the phenotypic expression for this locus is 0, then the chosen genotypic allele is promoted (unless it is a D).

Test Methodology

In the tests performed for this study, the modified Ng-Wong and additive diploidy approaches are compared against the proposed domination mechanism through a controlled set of changes using a modified 0/1 knapsack problem.

Test Cases

In the tests, change is introduced through changing the weight constraint for the knapsack instance. The knapsack is chosen in such a way that for each weight constraint, there is only one possible solution. The following types of change cases are used for the comparisons:

- Case 1: Optima oscillate between two values with fixed intervals
- Case 2: Change is random and is moderate in severity
- Case 3: Change is random and is high in severity

For case 1 the weight constraint oscillates between two fixed values every 500 generations. The change instances for the random change cases (cases 1 and 2) that are used for the discussions are given in Table 3. The *generation no.* shows the generation at which the change occurs, the *weight constraint* shows the constraint to be applied, the *optimum value* is the actual optimum solution for the knapsack for the given constraint and the *hamming distance* is the hamming distance between the previous and the current actual optimum solutions.

Table 3. Discussed Change Instances for Random Change Case

Generation no.	Weight Constraint	Optimum Value	Hamming Distance
0	22	112648	0
1500	100	131066	9
2100	40	128512	11

All algorithms use binary representation for the genes, two point cross over, roulette wheel selection and non-overlapping populations and are run using the same set of parameters as given below.

- population size: 250
- cross-over probability: 0.75
- mutation probability: 0.005

Controlled tests where the actual best fitnesses for each weight constraint is known is performed. The comparisons are done based on the ability of the dominance approaches in tracking the change in the environment for each different type of change case.

Test Results

The tests are performed for all test cases. The results are given as averaged over 5 runs for each test case. The results obtained for the random change cases for each domination approach is given in Table 4. The *value found* column gives the fitness value obtained by each approach for each change interval and the *generation* column gives the generation number at which this given best fitness is reached by each algorithm for each change interval.

The plots for the results obtained are given at the end of the paper. Figures 1, 2, 3 show the plots for the case where the change oscillates every 500 generations and Figures 4, 5, 6 show the plots for the cases where the change occurs randomly as given in Table 3. In all plots, the *x-axis* shows the number of generations and the *y-axis* shows the fitness values found. On all plots, the actual fitnesses are plotted using a dashed line and the fitness values found by the approaches are plotted using solid lines.

Table 4. Found Solutions by Each Method

Ng-Wong		Additive Diploidy		Proposed	
Value Found	Generation	Value Found	Generation	Value Found	Generation
112648	9	112648	25	112648	39
131040	1852	130728	1719	131066	1890
128512	2180	128512	2203	128512	2113

Conclusions

The results obtained for the modified Ng-Wong and additive diploidy approaches are in keeping with the ones found in [7]. The proposed approach shows a similar performance to these for the oscillating change case. The re-calculation of the domination map at the end of each generation brings extra computational load, however unlike the other two approaches, for the proposed approach it is not required to detect the change in the environment. Detecting when the change occurs is an important issue in dynamic environments and a good detection scheme affects the performance of the algorithm. In the proposed approach, the adaptation is done automatically based on a feedback obtained from the population. Because of this, there is no need to do calculations to explicitly detect the change. The detection is implicitly achieved by the adaptive property of the domination mechanism. This property makes the proposed approach very effective in dynamic environments.

In the cases where the change is random, the two tested algorithms do not give an acceptable performance as noted also in [7]. When the severity of the random change is moderate or low, the proposed approach shows good performance in adapting to the change. When the severity of the change is high, even the proposed approach may not be able to give acceptable results. These types of change instances require very high levels of diversity to be present in the population which diploidy alone may not be able to provide. Additional mechanisms to provide this are needed. These mechanisms are further discussed and explored in [11] by the same authors.

References

[1] Branke J. (2002), Evolutionary Optimization in Dynamic Environments, Kluwer Academic Publishers, 2002.

[2] Collingwood E., Corne D., Ross P. (1996), "Useful Diversity via Multiploidy", in Proceedings of AISB Workshop on Evolutionary Computation, 1996.

[3] Goldberg D. E. (1989), Genetic Algorithms in Search, Optimization and Machine Learning, Addison-Wesley, 1989.

[4] Grefenstette J. J. (1992), "Genetic Algorithms for Changing Environments", in Proceedings of Parallel Problem Solving from Nature 2, North Holland, 1992.

[5] Hadad B. S., Eick C. F. (1997), "Supporting Poliploidy in Genetic Algorithms Using Dominance Vectors", in Proceedings of the Sixth International Conference on Evolutionary Programming, Springer Verlag, 1997.

[6] Kim Y., Kim J. K., Lee S., Cho C., Hyung L. (1996), "Winner Take All Strategy for a Diploid Genetic Algorithm", in Proceedings of the First Asia-Pacific Conference on Simulated Evolution and Learning, 1996.

[7] Lewis J., Hart E., Graeme R. (1998), "A Comparison of Dominance Mechanisms and Simple Mutation on Non-Stationary Problems", in Proceedings of Parallel Problem Solving from Nature, Springer Verlag, 1998.

[8] Ng K. P., Wong K. C. (1995), "A New Diploid Scheme and Dominance Change Mechanism for Non-Stationary Function Optimization", in Proceedingsof the Sixth International Conference on Genetic Algorithms, 1995.

[9] Ryan C. (1994), "The Degree of Oneness", in Proceedings of the 1994 ECAI Workshop on Genetic Algorithms, Springer Verlag, 1994.

[10] Smith R. E., Goldberg D. E. (1992), "Diploidy and Dominance in Artificial Genetic Search", Complex Systems, Vol. 6, pp.251-285, 1992.

[11] Uyar A. S., Harmanci A. E. (2002), "Preserving Diversity Through Diploidy and Meiosis for Improved Genetic Algorithm Performance in Dynamic Environments", Advances in Information Systems (Ed. Yakhno T.), LNCS Vol. 2457, pp. 314-323, Springer, 2002.

[12] Weaver R. F., Hedrick P. W. (1997), Genetics 3rd Ed., Wm. C. Brown Publishers, 1997.

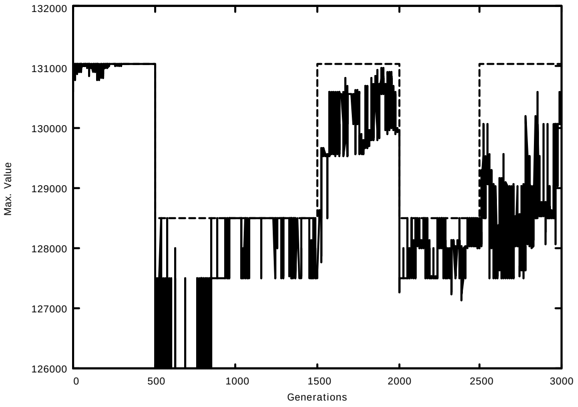

Fig. 1 Plot for Ng-Wong (oscillating change)

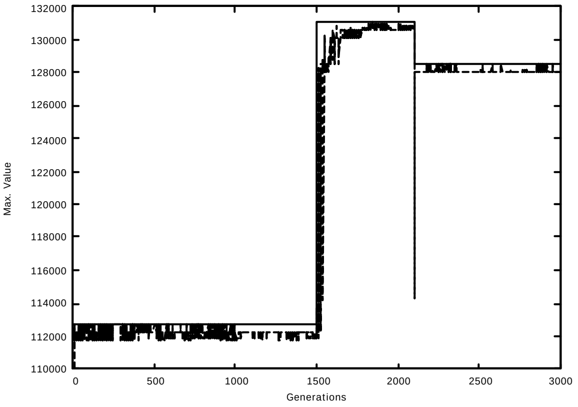

Fig. 4 Plot for Ng-Wong (random change)

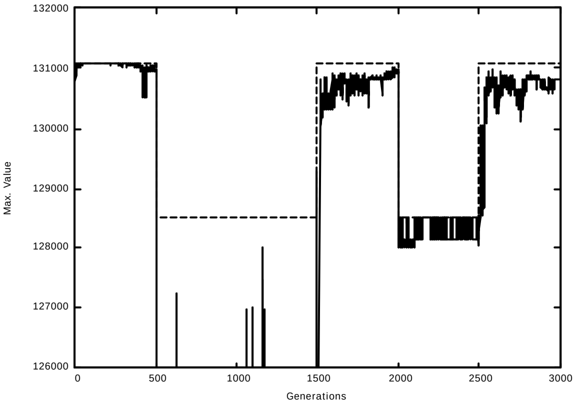

Fig. 2 Plot for additive dipl. (oscillating change)

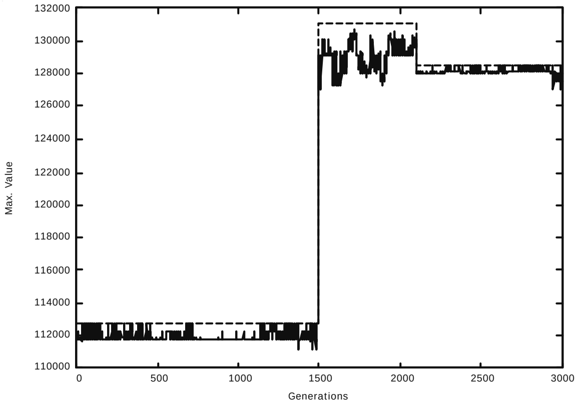

Fig. 5 Plot for additive dipl. (random change)

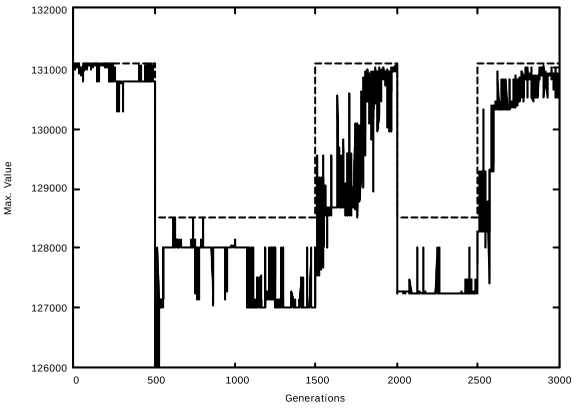

Fig. 3 Plot for proposed (oscillating change)

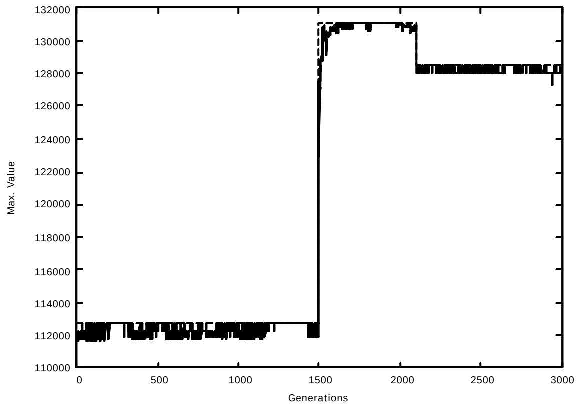

Fig. 6 Plot for proposed (random change)