Genetic Modification, Artificial Individual, Vaccine and Serum – Genetic Engineering in Evolutionary Systems

Ireneusz Gościniak[1]

[1] University of Silesia in Katowice, Institute of Computer Science
41-200 Sosnowiec, 39 Będzińska str., Poland
e-mail: goscinia@us.edu.pl

Abstract. The article shows the possibility of genetic engineering implementation in evolutionary systems as: genetic modification, artificial individual, vaccine and serum. In the article the group of immune systems is also proposed. Results of investigation presented in the article suggests that further development of genetic engineering in evolutionary systems will happen, thus raising their efficiency in problem solving.

1 Introduction

Genetics is the science of heredity and changeability of organisms that derive from information contained in the fundamental units of heredity – genes.

Complex living organisms are permanently subject to various external factors which can result in damage and disease. The immune system is the organisms organisational and defensive system which responds to the aggression of such factor. The immune system is built of lymphocytes**,** which genes control their defensive functions.

This system is sometimes supported by the use of vaccines and serums. As a result of vaccination after immunization with antigen, antibodies come into being in the recipient organism as compared to the use of prepared antibodies when serum is used.

Genetic engineering is the science and technology serving the artificial changes initiator in genotypes of organisms to modify those organisms (GMO - genetically modified organism) or population. New uses of genetic engineering are continually being discovered, and in effect it tends to create artificial life forms, i.e. synthetic genome creating.

Evolutionary algorithms, based on evolution and the functioning of natural organisms, arouse large interest in computer science [2]:

- genetic algorithms,
- genetic programming,
- immune systems.

The idea of genetic algorithms and genetic programming is based on computer simulation of life – population of individuals developing in the natural way in next generations. The aim of population developing is the breeding of individual to solve the given problem. For genetic algorithms it is the collection of solution parameters and in case of the genetic programming - the computer programme.

In information processing the principle of immune system function and its huge adaptive possibilities were used in artificial immune system [3, 4]. It contains the antigens - the collection of answers to earlier stated problems (the pathogen). Solution searching is not only held in a random way, as in genetic algorithms, but also by comparison to the collection of antigens.

Genetic algorithms, occurring together with immune systems, create the immune genetic algorithms. Similarly, immune elements applied in genetic programming, then in creating inductive Genetic Programming [6].

There are essential differences between genetic algorithms, genetic programming and immune system which determines their usefulness in solving a particular problem.

The use of genetic engineering in genetic algorithms and genetic programming for controlled changes of genes (genetic modification in range of population) and for the controlled synthesis of the genomes (the production of synthetic individuals) can accelerate the development of population and the breeding of specialized individual to solve a stated problem.

The use of genetic engineering in artificial immune system to produce synthetic antibodies, serums as well as antigens of vaccines, allows the acceleration of proper antigens producing, i.e. the solution of problem.

2 The Genetic Engineering Elements in Genetic Algorithms and Genetic Programming

The description of genetic algorithm as well as genetic programming is presented below [1]. The stars mark the elements of genetic engineering modification.

Genetic algorithm / genetic programming:

1. Randomly create an initial population (generation 0) of individuals (computer programs in genetic programming)
2. Iteratively perform the following subsets (named as generation of population) until the termination criterion will be satisfied:
   a. Assign a fitness value for each individual in the population using the fitness measure for the problem.
   *. ***Use methods of genetic engineering to analyse the genome to create genetic modification of individuals or the production of synthetic individuals.***
   . ***Select a group of individual(s) (program(s)) from the population and create new offspring individual(s) for the new population by genetic modification.*
   . ***Select a group of individual(s) (program(s)) from the population and create new offspring individual(s) for the new population by synthetic individuals.
   b. Select one or two individual(s) (program(s)) from the population with a probability based on fitness (with reselection allowed) to participate in the genetic operation in (c).
   c. Create individual(s) (program(s)) for the new population by applying the following genetic operations with specified probabilities:
      i. *Reproduction:* Copy the selected individual to new population.
      ii. *Crossover:* Create new offspring individual(s) for new population by:
         GA - recombining substrings from two selected individuals at a randomly chosen crossover point;
         GP - recombining randomly chosen part from two selected programs.
      iii. *Mutation:* Create new offspring individual for new population by:

GA - random mutating of accidentally chosen position(s) of one selected individuals;
GP - random mutating of accidentally chosen part of one selected program.

iv. *Architecture altering* operations *in genetic programming:* Choose an architecture-altering operation from the available repertoire of such operations and create one new offspring program for the new population by applying the chosen architecture-altering operation to one selected program.

v. ** ***Genetic modification: Create new offspring individual for new population by:***
GA - genetic modification of one selected individual;
GP - genetic modification of one selected program.

vi. *** ***Synthetic individual: Create new offspring individual for new population by:***
GA - Replacing one selected individuals by synthetic individuals;
GP - Replacing one selected program by synthetic individuals.

3. After reaching the termination criterion (which usually is determined by the maximum number of generations as well as the reaching of the solution value) designate the individual, which is identified by the method of result description (e.g. the best so far individual) as the result of the algorithm run. This result may represent a solution (or an approximate solution) to the problem.

The genetic engineering method can be implemented in the presented algorithm as genetic operators (underline stars) or separately operating on groups of individuals.

3 Immune System with Genetic Engineering Elements

The description of clonal selection algorithm is presented below [5]. The stars mark the elements of genetic engineering modification.

The clonal selection algorithm:

1. Initialisation – The first step of this technique is initialisation, which involves preparing an antibody pool of fixed size N. This pool is then divided into two components, a memory antibody section m that eventually becomes the representative of algorithms solution and the remaining antibody pool r, used for introducing the additional diversity of antibodies in the system.
2. Loop – The algorithm proceeds by executing a certain number of iterations of exposing the system to every known antigen. A single round of exposure or iteration represents a generation. The number of generations G, which the system executes, is determined by user, though the system can use the specific stopping condition.
   ****. Analysis of Antigens and Antibody – Method of genetic engineering to produce vaccines and serums.***
   *****. Vaccine injection -The vaccine can be applied to the pool of antigens.***
   a. Select Antigen – A single antigen is randomly selected without replacement (for the current generation) from the pool of antigens.
   b. Exposure – The system is exposed to the selected antigen. Affinity values are calculated for every antibody in relation to the antigen. The affinity is a measure of probability. The problem is similar to the determining of the distance of Hamming.

c. Selection – A set of n antibodies is selected from the entire antibody pool to have the highest affinity with the antigen.
d. Cloning – The set of selected antibodies is then cloned in proportion to their affinity (rank based).
e. Affinity Maturation (mutation) – The clones (set of duplicate antigens) are then subjected to the affinity maturation process to match the antigen in a better way to the problem in question. The degree of maturation is inversely proportional to their parents affinity (rank based), i.e. the greater the affinity, the lower the mutation.
f. Clone Exposure – The clone is then exposed to the antigen, and affinity measures are calculated.
g. Candidature – The antibody or antibodies with the highest affinity in the clone are then selected as candidates for memory antibodies for placement into m. If the affinity of a candidate memory cell is higher than that of the highest stimulated antigen from the memory pool m, then it replaces antigen. Group replacements occur in a similar, but batched manner.
h. Replacement – Finally, the d individuals in the remaining r antigen pool with the lowest affinity are replaced by new random antibodies ***or antibodies from serum – serum injection****.

3. Finish – After the completing of the training regime, the memory m component of the antigen pool is treated as the algorithm solution. Depending on the problem character, the solution may be a single best individual antigen or the collective of every antigen in the pool.

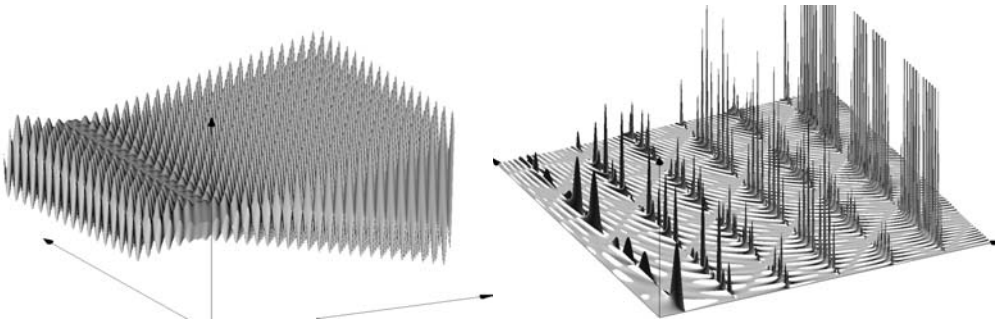
Vaccines and serums can be produced in immune system by the genetic engineering methods. They can be also taken from different organism which produces it – it solves the same problem. In such cases there is the immune system group, similarly to the case of multi-population genetic systems. The methods of genetic engineering in immune system group can be also applied.

4 Method of Genetic Engineering in Evolutionary Systems

The tasks, set before genetic engineering module in evolutionary systems, are as follows: the definition of genetic modification of population, the artificial individuals synthesis, production of vaccines and serums. In the evolutionary systems the development of population leads to specialisation, which results in the expanding of schema. In genetic algorithms the individual genome is described by the data model of the precisely defined map of genes. In genetic programming the continuous evolution of individuals follows. The module of genetic engineering will lead to analysis of population development by identifying the maps of genes, analysing similarity and interdependence. On this base the module can generate the genetic modifications of population or synthetic genomes. The aim of these operations will be the acceleration of schema identification or granting of specific features of solution, i.e. the obtainment of better solutions. The module of genetic engineering can apply the generally understood methods of computational intelligence. The example introduced below presents the applying of statistical analysis of antigens to produce serum antibodies.

5 Results of Investigations

Investigations were carried out on the immune system of a single individual. The antibodies were intensified by synthetic serum produced in module of genetic engineering on the base of statistical analysis of antigens. The maximum of function of very numerous maxima was searched as the solution to a given problem. Graphs of functions are introduced in figure 1.



a) $F_1(x, y)$ b) $F_2(x, y)$

Figure 1. Analysed functions.

Function $F_1(x, y)$ is below-described:

$$F_1(x, y) \Rightarrow 21{,}5 + x \cdot \sin(4\pi x) + y \cdot \sin(20\pi y) \tag{1}$$

by restrictions -3<x<12,1 and 4,1<y<5,8.

Function $F_2(x, y)$ is described as follows:

$$F_2(x, y) \Rightarrow \frac{\sin y}{\cos x} \sin(xy)(\sin x + \cos y) \tag{2}$$

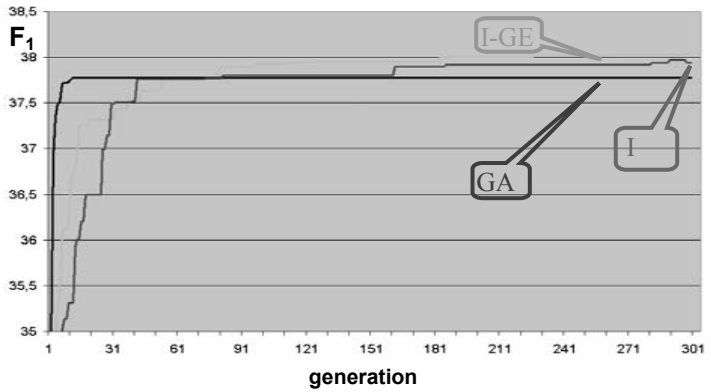
by restrictions 0<x<20 and 0<y<20 and F_2 (x, y)>0.

The maximum for introduced functions was searched by using the immune system, the genetic algorithm as well as immune system with built in elements of genetic engineering (figures 2a and 3a)

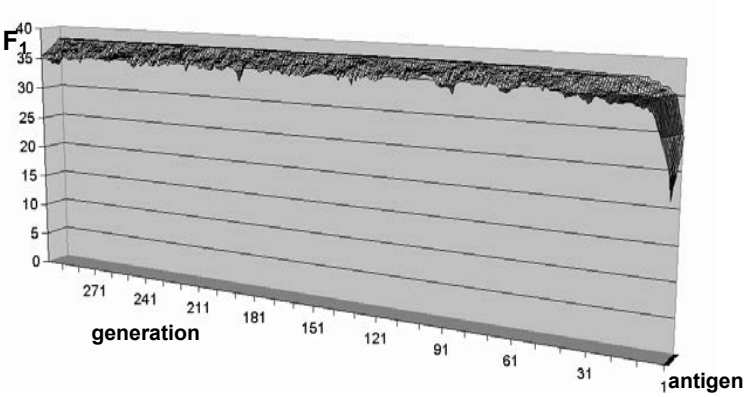
The information of values of antigens in immune system with elements of genetic engineering (figures 2b and 3b), values of antigens in immune system (figures 2c and 3c) as well as the values of individuals in genetic algorithm (figures 2d and 3d) was collected.

The software worked out for needs of study [7] with proper modifications was used for investigations. This programme was worked out on the base of the modified immune system (basing on above-presented algorithm of clonal selection), described in work [8].

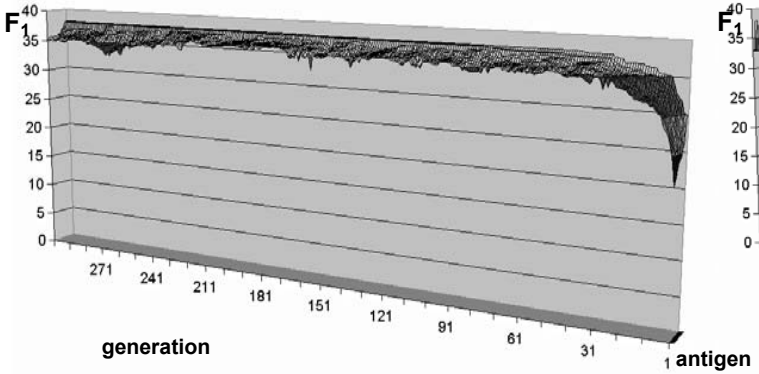
The introduced modification of immune system significantly improves the finding solution of function maximum.



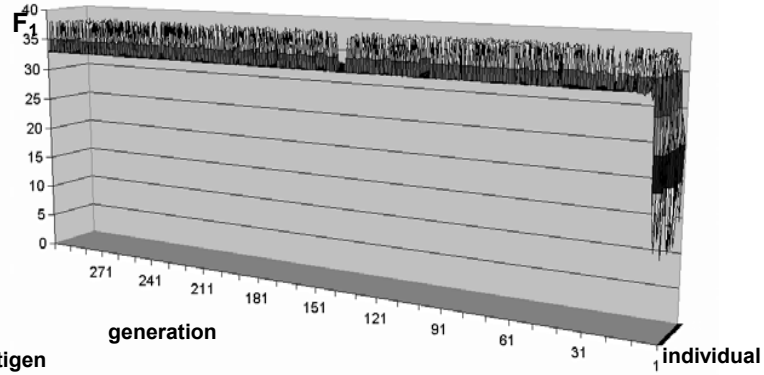
a) Maximum values of function F_1 for algorithm:
I-GE- immune system with elements of genetic engineering; I- immune system; GA- genetic algorithm.



b) Value of function F_1 for antigens in following populations – immune system with elements of genetic engineering.

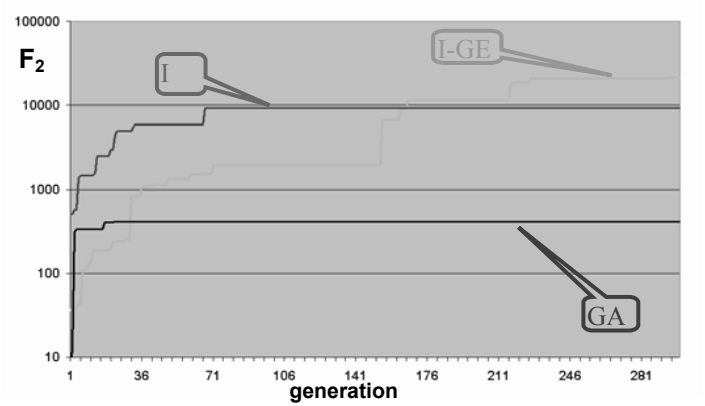


c) Value of function F_1 for antigens in following populations – immune system.

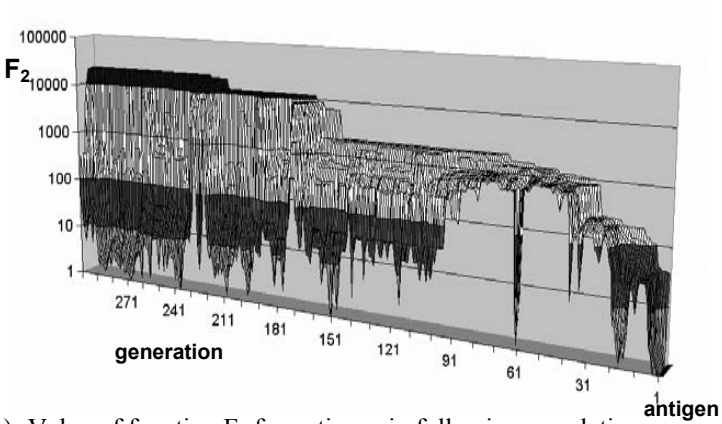


d) Value of function F_1 for individuals in following populations – genetic algorithm.

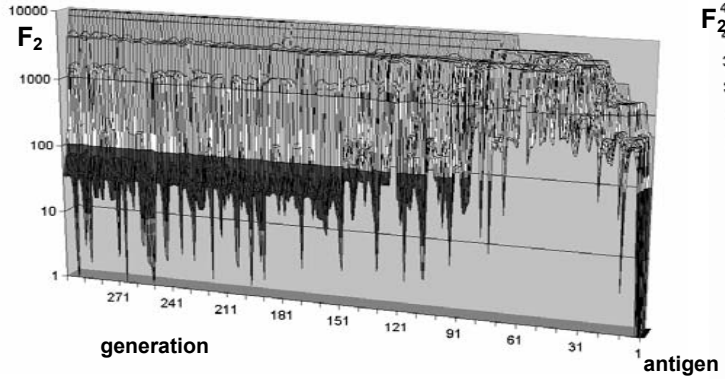
Figure 2. The results of investigations obtained during searching for maximum of function F_1.



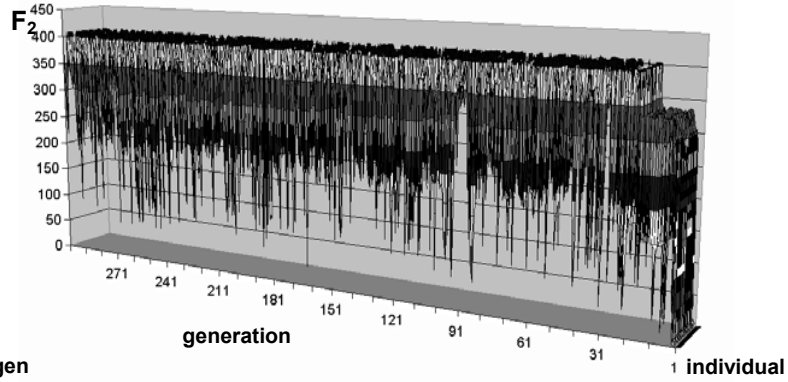
a) Maximum values of function F_2 for algorithm:
I-GE- immune system with elements of genetic engineering; I- immune system; GA- genetic algorithm.



b) Value of function F_2 for antigens in following populations – immune system with elements of genetic engineering.



c) Value of function F_2 for antigens in following populations – immune system.



d) Value of function F_2 for individuals in following populations – genetic algorithm.

Figure 3. The results of investigations obtained during searching for maximum of function F_2.

Figures 2a and 3a represent maximum values of analysed functions obtained in successive generations whereas figures 2b, 2c, 2d and 3b, 3c, 3d illustrate diversification of population in successive generations. Investigations demonstrate that the best effects were obtained in immune system with genetic engineering (particularly for function F2). These investigations also show that genetic engineering to a small degree reduces diversification of population in successive generations.

6 Conclusions

The group immune system proposed in the article decreases the risk of the detection of the local extreme solution, similarly to multipopulation genetic system. The methods of genetic engineering in evolutionary algorithms allow the development of individuals with specialized features. The introduced modification of immune system significantly improves the ability to find the function maximum. Results of investigation presented in the article suggest that further development of genetic engineering in evolutionary systems will happen, thus raising their efficiency in problem solving.

Bibliography

[1] Koza J. R. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*, MIT Press, 1992.

[2] Michalewicz Z. *Algorytmy genetyczne + struktury danych = programy ewolucyjne*, Warszawa: WNT, 1996.

[3] Wierzchoń S. *Sztuczne Systemy Immunologiczne. Teoria i Zastosowania*, Exit 2001.

[4] Wierzchoń S. T. Algorytm selekcji klonalnej i jego zastosowanie do optymalizacji w zmiennym środowisku. In *Systemy wspomagania decyzji*, Katowice: Instytut Informatyki Uniwersytetu Śląskiego, 38-44, 2002.

[5] White J. A., Garrett S. M. Improved Pattern Recognition with Artificial Clonal Selection ICARIS-2003, Lecture *Notes in Computer Science No.2787*, Edinburgh: Springer-Verlag, 181-193, Sept 1st-3rd 2003.

[6] Nikolaev N., Iba H., Slavov, V. Inductive genetic programming with immune network dynamics. In Spector L., Langdon W. B., O'Reilly U. –M., Angeline P. J.,editors, *Advances in Genetic Programming* Volume III, MIT Press, 355-376, 1999.

[7] Burda. M. *Zastosowanie algorytmów immunologicznych w projektowaniu rejestrów z nieliniowym sprzężeniem zwrotnym*, praca magisterska, Sosnowiec: Uniwersytet Śląski, 2005.

[8] Gościniak I. Łańcuchy funkcyjne jako reprezentacja osobników w algorytmach ewolucyjnych – aspekty teoretyczne, In *Systemy wspomagania decyzji*, Katowice: Instytut Informatyki Uniwersytetu Śląskiego, 152-158, 2005.