The Combined Immune Algorithm Based on Clonal Selection

Litvinenko V.I.[2,3]Bidjuk P.I[1]., Bardachov J.N.[2], , Fefelov A.A.[2],Sherstjuk V.G..[3]
[1]*National Ttechnical University of Ukraine "KPI"*
03056,Kiev, Peremogy, 37
[2]*The Kherson National Technical University*
73008 Kherson Berislavske Highway 24
[3]*The Kherson State Marine Institute*
73000, Kherson, Ushakov Prospect 17
peter@bidyuk.carrier.kiev.ua, kstu@tlc.kherson.ua, johndoe2004@mail.ru, fao1976@mail.ru, kmi@kmi.kherson.ua

Abstract. *A dynamic system identification algorithm is developed using the basic mechanisms of clonal selection and an idea of a new evolutionary computing paradigm – gene expression programming. On the basis of the algorithm developed a computer based system is proposed for making decisions relevant to forecasting of a single variable and multivariate time series. The results of computing experiments achieved with the system developed show high quality of short and medium period forecasts.*

Keywords

Artificial immune systems, Clonal selection algorithm, Gene expression programming, Hybrid algorithm

1 Introduction

While analyzing dynamic processes in finances, economy and some other areas, we had to stress the difficulties encountered during the analysis as well as forecasting dynamics of the variables [1]. First, high level of the processes nonlinearity and nonstationarity occurs, that makes impossible application of well-known earlier developed and investigated methods and models. In particular, models of linear regression that often show good results of stationary processes analysis, in this case show substantial deterioration of forecasts. Second, the use of these known methods suggests availability of samples that include at least 50 measurements to get satisfactory forecasts. Besides, to predict changes of processes dynamics that occur in reality, it is necessary to analyze carefully a large variety of factors influencing dependent variable. Also, it is necessary to determine what structure has the dependence between them and on this basis to construct correct mathematical model [1]. On the other hand, a natural immune system is a subject of a high interest for researchers because it possesses powerful and flexible abilities of processing the information as a decentralized intellectual system which provides excellent models with adaptive features at a local level and emergent behavior at global level [2, 6]. There are some theories, explaining the phenomena of immunology. They are used for solving the problems of machine training, recognition of images, optimization problems, fault detection, processing of images and navigation of robots [4]. We have carried out a research on application of principles and mechanisms hidden in natural immune system, especially clonal selection algorithm of [6] and molecular mechanisms of genes expression [3] with practical applications to solving the problems of interpolation, extrapolation and forecasting. The purpose of the study is application of principles of information processing by immune and other biological systems to forecasting a behavior of non-stationary dynamic systems. The primary goal of the research presented is development of a software system on the basis of the clonal selection algorithm (CSA) and an evolutionary computing paradigm – genes expression programming for solving the problem of non-stationary dynamic systems forecasting.

2 The Modified Clonal Selection Algorithm

The standard clonal selection algorithm operates with sequences of data and is used for solving the problems of classification, image recognition and optimization [2, 3]. Each data line of the algorithm, which is named as an individual or an antibody, represents one solution of a given problem. In the classical approach an optimization problem was reduced to search of solutions in multivariate space with a priory defined cost function. Solution of such a problem is represented by a sequence of variables which, after substitution into the cost function, give an optimum value. In the modified clonal selection algorithm the antibodies represent not simply a vector of variables, but connected chains which can be considered as trees of analytical dependence or as computer programs. And in the first case very often the problem is stated as an approximation of some relation usually represented in a tabulated form. For example, when it is required to summarize results of a set of experiments and to determine a form of dependence between output of system and its inputs.

Step 1. Creating of initial population of antibodies and coding of antibodies. There is a goal function which should be optimized. An affinity of an antibody corresponds to an estimate of criterion function for a given antibody: each antibody, $\boldsymbol{Ab_i}$, represents an independent input variable. More formally, we have to do the following: generate initial population $\mathbf{Ab^0}$ of antibodies.

$$\begin{aligned} &Ab^0 = \varnothing, j \in \{1,...N\}, \quad t = 0; \\ &\text{Randomly select } i_j \in I. \\ &Ab^0 = Ab^0 + i_j \end{aligned} \qquad (1)$$

where I is a space of individuals, i.e., a set of all possible structures representing an antibody, $i \in I$ is a subset of the individuals creating a population, t is a number of generation.

An approach to representing the antibodies is suggested in [3]. It is named a gene expression programming according to which the system "genotype – phenotype" is used which creates a computer program coded as a linear chromosome of a fixed length. It is believed that interaction between a genotype (antibody) and a phenotype (expression tree) allows to increase considerably a productivity, and also to overcome so called threshold of a phenotype. In the clonal algorithm proposed the data are represented as antibodies, i.e. as lines of fixed length consisting of symbols, which then are "expressing" as nonlinear objects (systems) of various sizes and forms (expression trees). The symbols are selected from the final alphabet of symbols which in turn will consist of two parts: the functional alphabet and the terminal alphabet. The symbolic line of an antibody is a consecutive record of a treelike structure displaying mathematical relation that uses as symbols of the functional alphabet, the symbols of mathematical operations and functions: $\{+,-,*,\backslash,Q,S,C\}$. In the given set the symbols Q, S and C mean, accordingly, the operation of finding a square root, and functions of a sine and cosine. As terminal symbols the symbols designating variables, arguments of a function are used, and a special symbol of a constant is also used "?": $\{?,a,b,c,d,...\}$ The tree is transformed into a line by consecutive record of all nodes and leaves starting from root node from left to right, and from top to bottom (see Figure 2). The initial population of antibodies is created by casual image, i.e. lines of symbols are initialized by symbols from the functional and terminal alphabet. In order to prevent construction of wrong expressions the lines of symbols are conditionally divided into two parts: a head and a tail. A head can contain both terminal, and functional symbols, and a tail contains only terminal symbols. The idea will be to avoid a situation with syntactically incorrect expression, even in that case when the head will consist completely of functional symbols. On this basis the general length of an antibody is calculated:

$$t = h(n-1)+1, \qquad (2)$$

where t is a general length of an individual; h is a length of a head; n is a maximum quantity of arguments used in a functional set. In our case n=2.

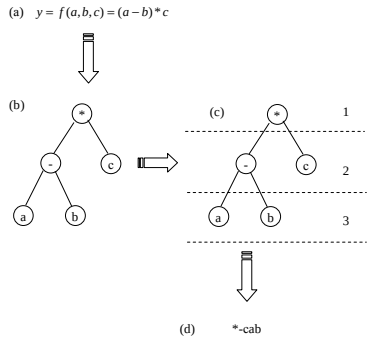


Fig.. 1. Transformation of a mathematical formula into a symbolic line of an antibody. (a) - the initial formula; (b) – an expression tree; (c) - division of a tree into levels; (d) - consecutive record of elements of levels as a line.

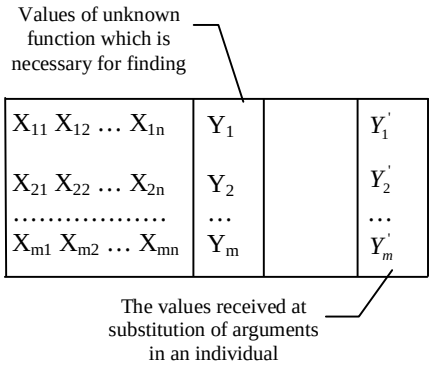


Fig. 2. Estimation of an antibody by affinity values table.

Step 2. Calculation of affinity for each antibody. Affinity of an antibody is a scalar estimate showing affinity of a result to optimum value. For each antibody $Ab_j \in Ab^t$ it is necessary to calculate value of the goal function $y_i = f(Ab_j)$ and determine affinity $g_j = affinity(y_j), j \in \{1,...N\}$. Depending on the type of problem solved on a population, i.e. maximization or minimization, an antibody is considered to be the best that possesses the highest or the lowest value of an affinity, accordingly. Affinity of antibodies is the main criterion for selection of individuals in the clonal selection algorithm. Affinity in the modified clonal selection algorithm is calculated using the table of values of approximated function. This table represents a matrix of sets of values of arguments of the function studied and the values of the function corresponding to them (see Figure 3). During estimation of an antibody the symbolic line of its genetic code is transformed into a tree of mathematical expression, and arguments from the table are substituted into this expression. The values of functions received as a result of this operation are compared to tabulated (real) values corresponding to them and, proceeding from a degree of affinity of values to each other, the judgment of approximation quality by an antibody with respect to the given function is constructed.

Thus affinity of an antibody could be viewed as one of errors: (mean square error: $MSE = \frac{\sum_{i=1}^{m}(Y'-Y)^2}{m}$, or relative mean average error: $RE = \frac{\sum_{i=1}^{m}\left|\frac{Y'-Y}{Y}\right|}{m}$). In this case we deal with the problem of minimization since the error should be reduced to zero. In our implementation information Akaike criterion is used as a measure of affinity that takes into consideration number of measurements ***N*** and a number of estimated

parameters p as well as the sum of squared errors: $AIC = N \ln\left[\sum_{k=1}^{N} e^{k}(k)\right] + 2p$; Bayes-Schwarz criterion (BSC) which in addition takes into account length of sample with due to including $\ln(N)$: $BSC = N \ln\left[\sum_{k=1}^{N} e^{2}(k)\right] + p\ln(N)$. Besides the specified four criteria we have also implemented so called external criterion of self-organizing according to which all sample of initial data is split into two identical parts N_A and N_B on which models A and B with outputs y_t^B and parameters C_i^A and C_i^B are determined accordingly. The used minimum displacement (consistency) criterion is equal to root-mean-square value of deviations (residuals) of outputs A and B for the whole sample $(t \in N)$: $n_{bias}^2 = \sum_{t \in N}\left(y_t^A - y_t^B\right)^2 / \sum_{t \in N} y_t^2 \to \min$.

Step 3. Check of a stop condition. Generally there is a big variety of various stopping criteria for immune algorithms, but a basic of them is a stop on achievement of a certain number of generations or a stop on achievement of some beforehand defined minimal value of an error.

Step 4. Selection of the best antibodies according to their affinity. Each estimated antibody of a population stores the value of its own affinity in its own structure. In the algorithm considered a certain percent of the best antibodies is selected out of the population (antibodies with the smallest value of affinity). This parameter was named as a degree of selection (or selection rate). The antibodies selected from the basic population form a new population – the population of clones. Formally: choose a subset of antibodies with the highest affinity $(Ab_{\{n\}})$.

$$
\begin{aligned}
&Ab_{\{n\}} = \left\{Ab_j \in Ab^t \mid select(Ab_j, Ab^t, n) = 1\right\}, where \\
&select(Ab_j, Ab^t, n) = \begin{cases} 1, rank(Ab_j) < n; \\ 0, rank(Ab_j) \geq n; \end{cases} \\
&rank(Ab_j) = j, if \\
&\forall j \in \{1,...N-1\}: affinity(f(Ab_j)) \geq affinity(f(Ab_{j+1})).
\end{aligned}
\qquad (3)
$$

Step 5. Cloning of selected antibodies. Creation of a population of clones. It is necessary to form a population of clones $C_{\{N_C\}}$ from $Ab_{\{n\}}$.

$$
\begin{aligned}
&C_{\{N_C\}} = \varnothing \\
&\forall j \in \{1...N_C\}: C_j = Ab_k, Ab_k \in Ab_{\{n\}}, where \\
&k = round\left(\frac{j}{round(\beta * N)}\right), \\
&N_C = \sum_{i=1}^{n} round(\beta * N),
\end{aligned}
\qquad (4)
$$

where $round(x)$ is an integer value operator.

Here each antibody in a population of clones copies itself several times forming this way a family of clones. The population of clones usually turns out to be essentially larger in size than the basic population.

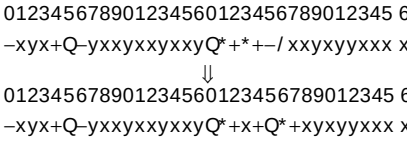
Step 6. Hypermutation of antibodies from a population of clones. A hypermutation is a casual change of one or several (depending on type of a hypermutation) symbols in a line of an antibody. To create a population of changed clones $C^*_{\{N_C\}}$ из $C_{\{N_C\}}$.

$$
\begin{aligned}
&C^*_{\{N_C\}} = \varnothing; \\
&\forall j \in \{1..N_C\}: \boldsymbol{C}_j^* = \begin{cases} mutate(\boldsymbol{C}_j), rnd(p_m) = 1; \\ \boldsymbol{C}_j, rnd(p_m) = 0; \end{cases} \\
&\boldsymbol{C}_j \in C_{\{N_C\}},
\end{aligned}
\qquad (5)
$$

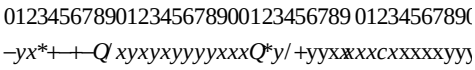
where $rnd(p_m)$ is a function of modeling of appearance of casual event with predefined probability p_m, $mutate(\boldsymbol{C}_j)$ is an operator of a mutation that casually changes one or several genes of an antibody.

The symbols are taken from the alphabet of symbols. Naturally in a tail part of an antibody act restrictions on a hypermutation: the symbols can only be taken from the terminal alphabet. As a result of the hypermutation the population of clones changes.

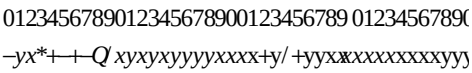
Step 7. IS-transposition of sequences of antibody elements. Moving of short fragments of genes (inserted sequence of genes**)** of **IS**-elements (except for a root) to a head of genes (**IS – "**Insertion sequence elements" or IS-elements). The essence of this operation is that any sequence of an antibody could become an **IS**-element. Here a copy of transpozone is made and it is inserted in any position in a head area of an antibody, except for an initial position.



Step 8. RIS-transposition. Here transposition of short fragments with a function in the first position which move to roots of antibodies (genes) is performed or RIS-elements are formed (RIS – "Root insertion sequence elements"). All RIS-elements begin with a functional symbol. A point is casually selected in a head part, and the gene is scanned downwards from a head until the functional symbol is found. The symbol found this way becomes as an initial position of a RIS-element. If the functional symbol is not found, any changes do not occur. This operator selects an antibody in the casual image, modifies a gene included in it as well as beginning part of a RIS-element and its length. Consider an antibody consisting of two genes



Assume that in a gene 2 as a RIS-element the sequence $"+yy"$ has been chosen. Then the transpozone copy moves to a root of sequence, and we get the following antibody:



Created programs remain syntactically correct because the structural organization of an antibody (chromosome) is preserved. RIS-transpositions make more cardinal changes to an expression tree than IS-transpositions since RIS-transpositions make changes to a root of a tree. In the nature if the element is inserted into the beginning of sequence of a gene, generated mutations displace a mutation and it considerably changes properties of coded fiber. Similarly to a hypermutation and IS-transposition insertion of a root has huge potential for creating the variability.

Step 9. Definition of affinity of a population of the changed clones. This operation works similarly to the block 2. For each antibody $\boldsymbol{C}_j^* \in \mathrm{C}_{\{N_C\}}^*$ calculate value of goal function $y_j = \mathrm{f}(\boldsymbol{C}_j^*)$ and determine affinity $g_j = affinity(y_j),\ j \in \{1..N_C\}$

Step 10. Selection of the best antibodies from a population of clones according to their affinity. Here we choose a subset $\mathrm{C}_{\{n\}}^*$ from *n* antibodies with the highest affinity from a population of changed clones $\mathrm{C}_{\{N_C\}}^*$, i.e. a certain percentage of the best antibodies is selected so that the chosen number coincided with the number of antibodies selected at step 4 from basic population.

Step 11. Replacement or carrying over the best antibodies from a population of clones into the basic population. Antibodies in the basic population, selected on a step 4, are replaced with the antibodies received on a step 8. To replace a subset $\mathrm{Ab}_{\{n\}}$ на $\mathrm{C}_{\{n\}}^*$

$$\forall j \in \{1..n\} : \boldsymbol{Ab}_j = \boldsymbol{C}_j^*, \quad \boldsymbol{Ab}_j \in \mathrm{Ab}_{\{n\}}, \boldsymbol{C}_j^* \in \boldsymbol{C}_{\{n\}}^* \qquad (6)$$

Thus the size of the basic population remains constant.

Step 12. Clone removal or selection of the worst antibodies in the basic population and their replacement with the new casually generated antibodies. A certain percentage of the worst antibodies in the basic

population which are replaced with the same number of the new generated antibodies are selected. The percentage of replaced antibodies is referred to as a degree of replacement (or replacement rate):

$$\mathrm{Ab}_{\{d\}} = \left\{ \boldsymbol{Ab}_j \in \mathrm{Ab}^t \mid negselect\ (\boldsymbol{Ab}_j, \mathrm{Ab}^t, d) = 1 \right\}, \text{ where}$$

$$negselect\ (\boldsymbol{Ab}_j, \mathrm{Ab}^t, d) = \begin{cases} 1, & rank\ (\boldsymbol{Ab}_j) \geq N - d; \\ 0, & rank\ (\boldsymbol{Ab}_j) < N - d; \end{cases}$$

$$rank\ (\boldsymbol{Ab}_j) = j,\ if\ \forall j \in \{1..N-1\} : affinity\ (\mathrm{f}(\boldsymbol{Ab}_j)) \geq affinity\ (\mathrm{f}(\boldsymbol{Ab}_{j+1})),$$

$$k \in \{1..d\},$$

$$random\ \ select\ \ \boldsymbol{i}_k \in \mathrm{I},$$

$$\boldsymbol{Ab}_k = \boldsymbol{i}_k,\ \boldsymbol{Ab}_k \in \mathrm{Ab}_{\{d\}}.$$

Passing from generation to generation, the population is modified by the process of evolution which is influenced by the two major factors: heredity (of clones) and variability (hypermutation); and the whole process converges the problem to optimum solution. Solution of a problem is represented by optimum formula most precisely describing required functional relation.

Step 13. Check of a stop condition. Using selected criterion ε, test performance of an algorithm stop condition.

$$\begin{aligned} &\mathrm{Ab}^{t+1} = \mathrm{Ab}_{\{n\}} \cup \mathrm{Ab}_{\{d\}} \cup \mathrm{Ab}_{\{N-(n+d)\}}, \\ &t = t+1, \\ &Output\ \mathrm{Ab}_{\{n\}},\ if\ \ stop(\varepsilon) = true, \\ &return\ to\ step\ 2,\ if\ \ stop(\varepsilon) = false. \end{aligned} \qquad (8)$$

The operator of mutation is not considered here in detail because as of today there is a large number of various versions of its implementation. Parameters n and d are not connected with each other directly but only in the size of population N, i.e. the following restrictions are imposed on them: $n + d \leq N$. It means that after one generation in a population there can exist not changed individuals. Before application of the operator of selection all individuals should be ranked, i.e. it is necessary to carry out sorting of a population on decrease affinity. Hence, the lowest rank will be assigned to an individual with the largest value of affinity.

3 Experimental researches

For demonstration of the algorithm operation a number of solar activity has been used

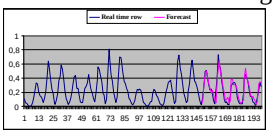


Рис.3 The Forecast of solar activity (Training sample: 150. Test sample: 50)

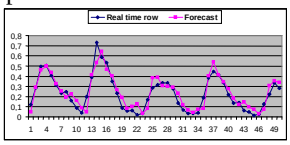


Fig.4 The Single-step forecast (Mistake RMSE on test sample: 0,009076**)**

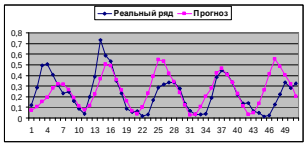


Рис.5 The Multistep-by-step forecast (Mistake RMSE on test sample: 0,021658)

The model of the predicted process can be presented in the form of prefix records:
+(S(L(+d(L(/hf)))))(S(Lh))(Q(*(S(-aj))(C(-hh))))eb(-(Cj)e)(-1.56)(C(L(S(Lb)))))(Lk)(C(+gd)))
или в виде дерева:

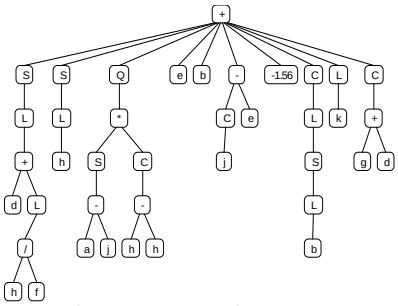


Fig.6 Representation of the decision of a problem in the form of a tree

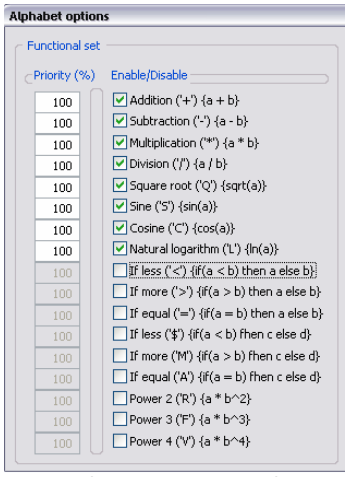


Fig.5 Functions used for construction of trees of expressions

4 Conclusions

In the paper for the first time is presented a hybrid immune algorithm and its object-oriented implementation, intended for solving the problems of extrapolation and interpolation. In the algorithm proposed it is possible to combine the best qualities of the clonal selection algorithm (a combination of global and local search) and gene expression programming algorithm (a way of coding of chromosomes and an opportunity to overcome the threshold of a phenotype that allows to carry out unlimited study of space of search that all other evolutionary algorithms are deprived). It is shown, that the algorithm yields effective results for solving the problems of forecasting of non-stationary dynamic systems. Program implementation and the computing experiments have shown that the algorithm yields comprehensible results for short-term and medium-term forecasts. The powerful tool opportunities of the developed object-oriented library determine a good prospect for its further adaptation to various computing problems, in particular, for solving the problems of identification. Besides, in the algorithm developed the so-called "internal" criterion has been used, but authors believe it is necessary to carry out in the future research on use of "external" criteria [7], that in a combination with the used approaches will allow to get quite stable models of nonstationary and nonlinear dynamic systems in various areas.

References

[1] Bidyuk P.I., Backlan I.V., Litvinenko V.I. (2003), "Modelling and forecasting of heteroscedastic processes," *Automatics, automation, automatic complexes and systems*, N. 2(12), pp. 11-19.

[2] D. Dasgupta (editor), "Artificial Immune Systems and Their Applications," *A book published by Springer Verlag Inc.*, January 1999.

[3] Ferreira, C., (2001), "Gene Expression Programming: A New Adaptive Algorithm for Solving Problems," *Complex Systems*.

[4] De Castro, L. N. & Von Zuben, F. J. (2000a), "The Clonal Selection Algorithm with Engineering Applications", *submitted to GECCO'0*0.

[5] Litvinenko V.I., Fefelov A.A., Goravski S.P. (2003), "Object-oriented realization of clonal selection algorithm," *Radio electronics, computer science, management, Zaporozhie*, N. 9, pp. 81-88.

[6] Gritsik V.V., Litvinenko V.I., Tsmots I. G., Stekh S.M. (2003), "Theoretical and applied problems of use of artificial immune systems," *Information technologies and systems*, vol. 6, N1-2, pp. 7-45.

[7] Ivahnenko A.G., Stepashko V.S. (1984), "Noise stability of modelling," *A scientific idea, Kiev*, 295 p.