Application of Genetic Algorithm in Problems of Approximation of Complex Multidimensional Dependencies and Identification of Parameters of Theoretical Models

Dmitry Glukhov[1][0000-0003-4983-2919], Tatsiana Hlukhava[2][0000-0002-3154-3863],
Aliaksandr Lukyanau[3][0000-0001-6812-2184]

Polotsk State University, Blokhin str., 29, Novopolotsk, 211440, Republic of Belarus
[1]d.gluhov@psu.by, [2]t.gluhova@psu.by, [3]a.lukyanov@psu.by

Abstract. The article proposes a method for constructing an analytical approximation of n-dimensional data, based on the use of a genetic algorithm. A feature of the method is that the encoding of the search space is performed in the form of a parsing tree for an algebraic expression by the parser of the context-free grammar of the class LR (1). In addition, during the evolutionary process, in addition to the use of structure mutations (subject to their positive influence), the stage of mutation of the coefficients is performed, which allows avoiding the target function falling into local extremum. And also at each step of the evolutionary process, there is a stage for searching for an extremum in the space of coefficients and a stage for simplifying the analytical model.

Keywords: genetic algorithm, approximation, theoretical models

1 Introduction

By the complexity of a multidimensional dependency, we mean the uncertainty regarding the functional dependencies present in the data between input variables and the output value of the function. Such tasks include problems of identifying parameters of analytical models of approximation of multidimensional dependencies, identifying parameters of theoretical probability distributions, identifying parameters of trend models in forecasting.

Data approximation in the absence of a priori information about the real model requires the use of universal approximators, such as artificial neural networks, fuzzy logical approximators. However, the analytical representation of the model is the most compact and most accurate. Approximation methods based on the identification of parameters of some theoretical model solve the problem using an immutable model adopted from any assumptions of reasonableness.

We are making an attempt to use the genetic algorithm to find the most accurate analytical model of the approximator. The genetic algorithm performs not only the identification of parameters of a theoretical model, but also the search for the model itself in the space of possible modifications of the structure. The purposeful movement of the population of models to models providing a qualitative approximation is carried out due to the law of evolutionary selection according to the criterion of minimizing the standard deviation.

2 Implementation of genetic algorithm

A genetic algorithm (GA) is a heuristic search algorithm used to solve optimization and simulation problems by sequentially selecting, combining, and varying the desired parameters using mechanisms that resemble biological evolution.

Previously, we made attempts to solve this problem on multi-agent systems, which for analytical models in the space of 3 variables showed results superior to results of artificial neural networks, but on a limited variety of models [1, 2, 3]. The main disadvantage of such approximators is a limited set of structure transformations formulated in the form of local heuristics. Heuristics are obvious ways for a human expert to change the structure. Accordingly, modifications that are not obvious to the expert become impossible. In our opinion, the use of GA removes this restriction.

In addition, we have built approximators based on fuzzy logic (FLA). Fuzzy logic systems proposed by Lotfi Zadeh [8] were developed in works [9, 10]. If we consider the unknown parameter as continuous, then in this case we can draw a parallel between the conclusion about the value of the unknown parameter and the approximation of the function and talk about the property of a fuzzy system to act as a linguistic approximator. The advantage of such approximators is their speed, and the main disadvantage – limited opportunities to improve approximation accuracy.

The context-free grammar Gexp of the LR(1) class of algebraic expressions proposed by us is presented below in the Beckus-Naur form:

```
formula:   exp    {ANode* o = (ANode*)$1; created-
Nodes.push_back(o);}
|    formula ';' exp  {ANode* o = (ANode*)$3; created-
Nodes.push_back(o);}
;
exp : NUM    {ANodeNUM* o = new ANodeNUM($1); $$ =
(void*) o;}
| '(' exp ')'  {$$ = $2;}
| exp '+' exp  {ANode* o = (ANode*) new AN-
odeOp((ANode*)$1, (ANode*)$3, PLUS); $$ = (void*) o;}
| exp '-' exp  {ANode* o = (ANode*) new AN-
odeOp((ANode*)$1, (ANode*)$3, MINUS); $$ = (void*) o;}
| exp '/' exp  {ANode* o = (ANode*) new AN-
odeOp((ANode*)$1, (ANode*)$3, DIV); $$ = (void*) o;}
```

```
| exp '*' exp  {ANode* o = (ANode*) new AN-
odeOp((ANode*)$1, (ANode*)$3, MUL); $$ = (void*) o;}
| 'p' 'o' 'w' '(' exp ',' exp ')' {ANode* o = (ANode*)
new ANodeOp((ANode*)$5, (ANode*)$7, POW);
          $$ = (void*) o;}
| 's' 'i' 'n' '(' exp ')'   {ANode* o = (ANode*) new ANo-
deFunc((ANode*)$5, SIN); $$ = (void*) o;}
| 'c' 'o' 's' '(' exp ')'   {ANode* o = (ANode*) new ANo-
deFunc((ANode*)$5, COS); $$ = (void*) o;}
| 'l' 'o' 'g' '(' exp ')'   {ANode* o = (ANode*) new ANo-
deFunc((ANode*)$5, LOG); $$ = (void*) o;}
| 'e' 'x' 'p' '(' exp ')'    {ANode* o = (ANode*) new
ANodeFunc((ANode*)$5, EXP); $$ = (void*) o;}
| 't' 'a' 'n' '(' exp ')'   {ANode* o = (ANode*) new ANo-
deFunc((ANode*)$5, TAN); $$ = (void*) o;}
| 'c' 't' 'a' 'n' '(' exp ')' {ANode* o = (ANode*) new
ANodeFunc((ANode*)$6, CTAN); $$ = (void*) o;}
| NAME  {ANode* o = (ANode*) new ANode-
NAME(string((char*)$1)); $$ = (void*) o; delete $1;}
;
```

3 Parsing process

In the process of syntactic parsing, as we can see, the right-side parsing tree is built in the form of an object-oriented structure. In order to be able to compile an algebraic expression into a tree of objects nested into each other, we have developed a class system presented in the class diagram (Figure 1).

As a result of syntactic parsing the initialization file, an initial population of analytical n-dimensional models is created. Initial variations of the models are formulated by an expert in the formal language of the Gexp grammar.

Objects of the base class ANode and derived classes ANodeNUM, ANodeNAME, ANodeOp, ANodeFunc, ANodeProxy attribute semantic stack nodes and form a tree of right-side parsing of an algebraic expression.

The Genetic class implements the main stages of the GA. Namely:

1. initialization() - initialization of the population with expert-defined algebraic expressions for the case of n-variables;
2. calcAllErrors() - calculating of the approximation error for each individual in its current state;
3. coefficientMutation() – the process of random cloning of individuals with mutation of coefficients to ensure the possibility of exit from local extremes;
4. structureMutation() – the process of cloning with a structure mutation, subject to the positive effect of the mutation;
5. fitCoefficiets() – the process of searching for the extremum of the target function in the coefficient space. In the implementation of this process, we have chosen the

gradient descent algorithm, but using coordinate relaxation. This algorithm is not fast, but it is reliable enough;

6. simlifyStructure() – simplifies the structure of an expression in specific cases.

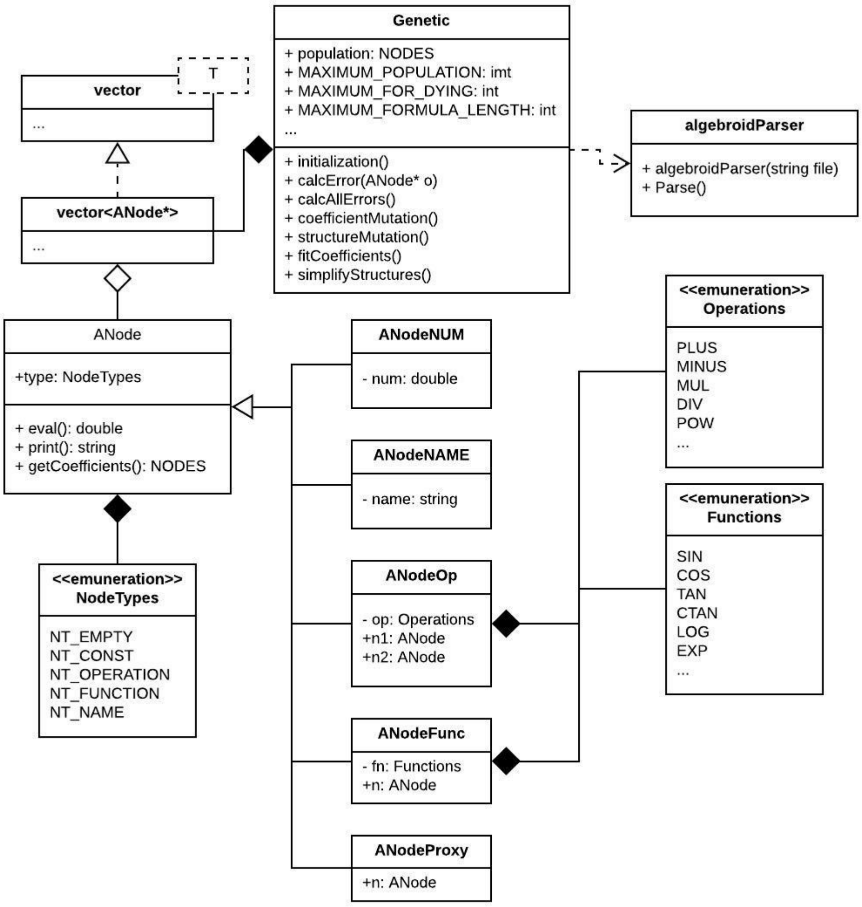


Fig. 1. Class diagram of GA software for searching the model of an analytical n-dimensional approximator

Simplification of the structure leads to the removal and replacement of semantic tree nodes (the ANodeProxy class is provided for the simplification procedure) and is provided in the following cases:

- x * 0 | 0 * x → 0;
- x^0 → 1; x^1 → x;
- x * 1 | 1 * x → x;

Simplification is triggered when the algorithm detects the convergence of the coefficient to 0 or 1.

The block diagram of the evolutionary search algorithm is shown in Figure 2.

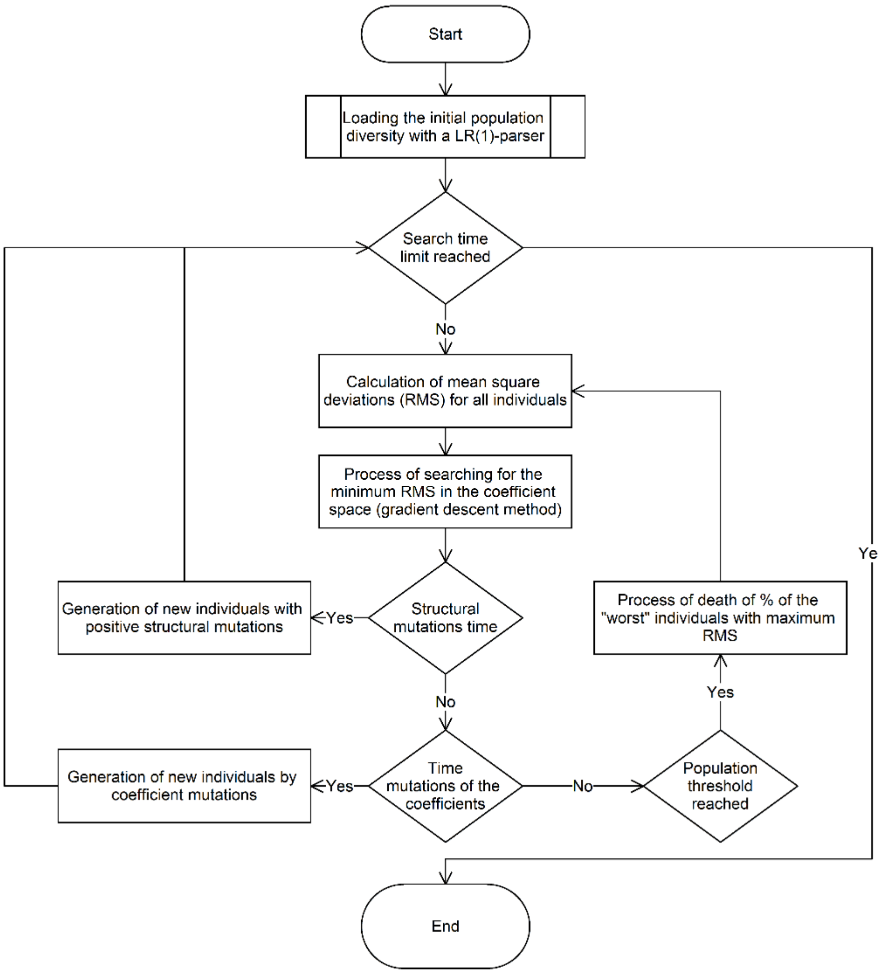

Start
Loading the initial population diversity with a LR(1)-parser
Search time limit reached
No
Yes
Calculation of mean square deviations (RMS) for all individuals
Process of searching for the minimum RMS in the coefficient space (gradient descent method)
Generation of new individuals with positive structural mutations
Yes
Structural mutations time
Process of death of % of the "worst" individuals with maximum RMS
Yes
No
Generation of new individuals by coefficient mutations
Yes
Time mutations of the coefficients
No
Population threshold reached
End

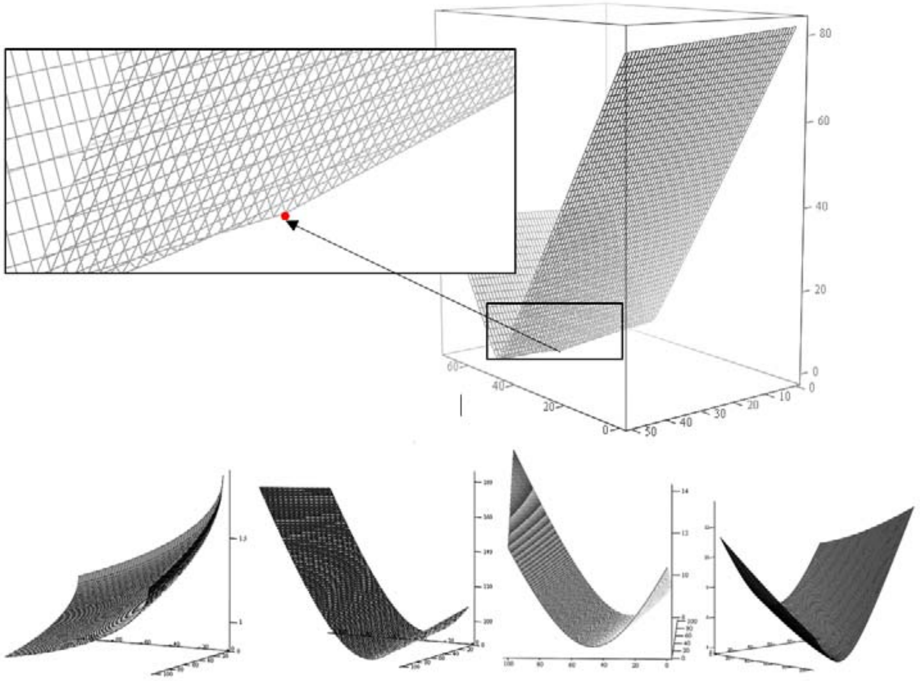
Fig. 2. Block diagram of the evolutionary search algorithm of the developed genetic algorithm

As the objective function we have chosen the standard deviation between the theoretical and calculated values of the function (SD). In the m-dimensional space of the coefficients of analytical models, the SD surface may have local extremes due to the limited data area and the complexity of approximated dependencies. However, the nature of the surface allows us to suggest the possibility of using the gradient descent algorithm to search for a local or global extremum. The nature of the SD surface in the space of two coefficients a, b of a function of one variable is illustrated in Figure 3.

In a number of works devoted to this topic, it is proposed to search for coefficients that minimize SD as a result of mutation of coefficients and to use the Nelder-Mead algorithm to search for a local extremum. [5, 6, 11]

Given that we are trying to minimize the standard error of the deviation of some analytical function from a discrete data set, we conducted a comparative analysis of the applicability of various numerical optimization methods and provide the user of the system with a choice of a specific method. Our approach is based on two points:

1. The mutation of coefficients is necessary for an abrupt transition to a new extremum search point, which allows the algorithm to run several extremum search strategies from different areas of the n-dimensional search space, and, therefore, to avoid getting into local extremes.
2. Application of the most qualitative methods for finding the extremum depending on the specifics of the problem (gradient descent method, simplex method, etc.) for identification of coefficients with a given accuracy.



1) 2) 3) 4)

Fig. 3. The SD surface in the space of coefficients *a*, *b* of models:

1) $y(x) = \sqrt{\frac{a}{x}} + \sqrt{\frac{b}{x^2}}$, 2) $y(x) = ax^2 + bx$, 3) $y(x) = a\log(x) + bx$, 4) $y(x) = 5\,e^{1/x} + 3x$

Table 1. Example of the result of identifying model coefficients using the coordinate descent method

№	Formula	Standard deviation
Before identification of coefficients		
1	5 * x1 * x1 + 3	0.00000
2	2 * x1 + 3 * x2 + 4	56.92715
3	2 * pow(x2, 2) + 2	154.52993
4	15 * pow(x1, 2) + 13	166.70333
5	2 * x1 * x1 + 3 * x2 * x2 + 2 * x1 * x2 + 3 * x1 + 3 * x2 + 3	359.15568
6	11 * pow(x1, 3) + 3	+INF
7	pow(x1, x2) * 5	+INF
After identification of coefficients		
1	5 * x1 * x1 + 3	0.00000
2	27.9921 * x1 + -2.02022 * x2 + 3.12347	17.37498
3	1.8 * pow(x2, 1.5) + 1.579	57.33454
4	5.00068 * pow(x1, 1.99993) + 2.99844	0.00100
5	4.99973 * x1 * x1 -0.48889 * x2 * x2 + 0.33344 * x1 * x2 -3.33275 * x1 + 4.86652 * x2 + 3.2222	0.00099
6	5.00068 * pow(x1, 1.99993) + 2.99844	0.00100
7	pow(x1, x2) * 3.3e-006	50.52150

4 Algorithm operation example

In our numerical experiments, the best convergence rate was shown by the coordinate descent method, in which the step is performed along the coordinate giving the maximum gain. However, the search and implementation of more efficient optimization algorithms remains the subject of further research.

It is important to note that we have proposed a new method for screening out equivalent formulas. Since the same formula, due to the commutativity of operations (+, -, /, *) and the distributivity of the corresponding pairs of operations, as well as the possibility of arbitrary arrangement of brackets, can be written in many different ways, we proposed to form numerical hashes of formulas using special codes of variables, coefficients and functions and a mirror - modified algebra (+ instead of *,- instead of /, * instead of +, / instead of -).

An example of the operation of this hash function is shown in the Table 2:

Table 2. An example of the operation of this hash function

Formula	Hash
2 * log(x1) * x2	16994
log(x1) * 2 * x2	16994
x2 * 2 * log(x1)	16994
log(x1) * x2 * 2	16994
2 * (log(x1) + x2)	2.89382e+007
2 * log(x1) + x2 * 2	2.89722e+007

An example of the gradient descent method for 7 population individuals with an accuracy of 0.001 is shown in Table 1. The example also shows that we have developed a mechanism for processing results in the form of non-numbers and infinities, as large SD values. This approach allows you to continue searching without the occurrence of exceptional overflow situations and divisions by zero.

5 Algorithm application example

As an example of the application of the GA we constructed for solving an applied problem, we consider the problem of accounting for transients in stationary non-isothermal models of gas transportation.

Formulation of the problem: when the compressor station is switched on, gas heating begins and, despite the fact that the heated gas has not yet spread through the pipeline, the stationary model describes the state of the pipeline that will occur after the transition to steady state. Thus, significant sections of the pipeline in the model are found to have an overvalued gas temperature and, given the strong dependence of gas density on temperature, with an underestimated value of the gas reserve.

We calculated the correlation coefficient, which for the considered areas with temperature increase showed a high negative correlation, so for the 4 graphs given for example in Figure 4, the correlation coefficient was Correl(X, Y) = { -0,84369, -0,88839, -0,88509, -0,85517}.

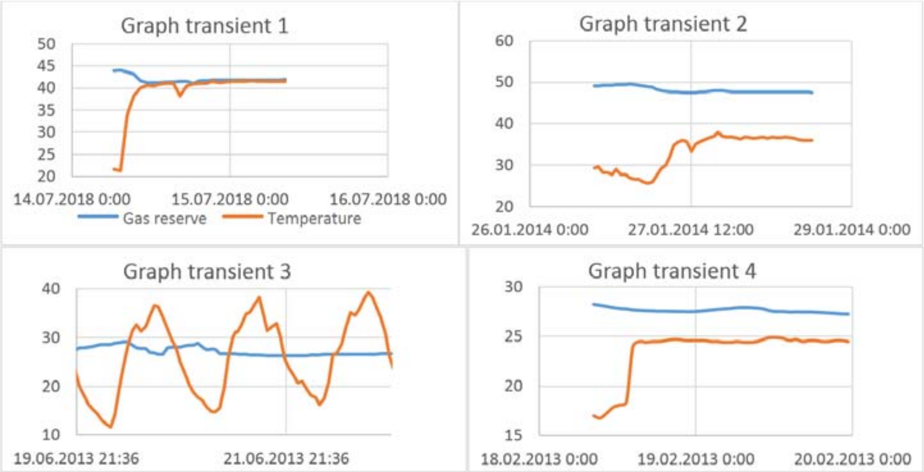


Fig. 4. Examples of transient graphs for the main gas pipeline model OJSC «Gazprom transgaz Belarus»

As the authors of the software package for calculating the gas reserve on the main gas pipeline of OJSC "Gazprom transgaz Belarus», we attempted to eliminate the influence of such transients by introducing the inertia of the temperature change at the compressor output during the time of system transition to the steady state.

We used GA to find an analytical relationship between the value of the transition time y and the current value of the gas flow Q, the ground temperature T_g, and the temperature difference at the output of the compressor station dT. The task of the approximator is to find such an analytical dependence that has a minimum SD for describing $y(Q, T_g, dT)$.

Comparative results of GA operation are given in Table 3.

Table 1. Comparative results of GA operation

№	Approximation method	Model	SD
	GA	1.66597 * pow(1.7573, 1.1225 * log(3.21177 * pow(dT * 4.23434, 1.1878))) / pow(log(0.0858575 * Q), 0.0742341) - 4.15915 * dT	1.078879
1	GA	52 - 14.48 * pow(dT, 0.08961) / pow(Q, -0.1755)	1.330954
2	GA	102.607 * pow(dT, -0.369501) / log(0.265488 * Q)	1.200513
3	FLA	The model is described in our work [10]	1.080616

The solution found by GA exceeded in accuracy the fuzzy logical approximator with the selected optimal smoothing settings, while giving the most compact representation of the approximator.

The dynamics of improving the approximation quality can be demonstrated by the graph of positive mutations shown in Figure 5.

One of the parameters of the proposed GA is the limit size of the formula, which allows obtaining a wide variety of compact variants of approximators.

Developed GA has the possibility of flexible adjustment of limit thresholds of population size, percentage and death period of unpromising individuals, the probability of a period and type of mutations of the structure and coefficients, formulations of model simplification tools, algorithms of search of extremum. In particular, to solve the problem of accounting for transients in stationary non-isothermal models of gas transportation, we are interested not only in time, but also in the nature of temperature inertia, which requires collecting and processing of additional information.

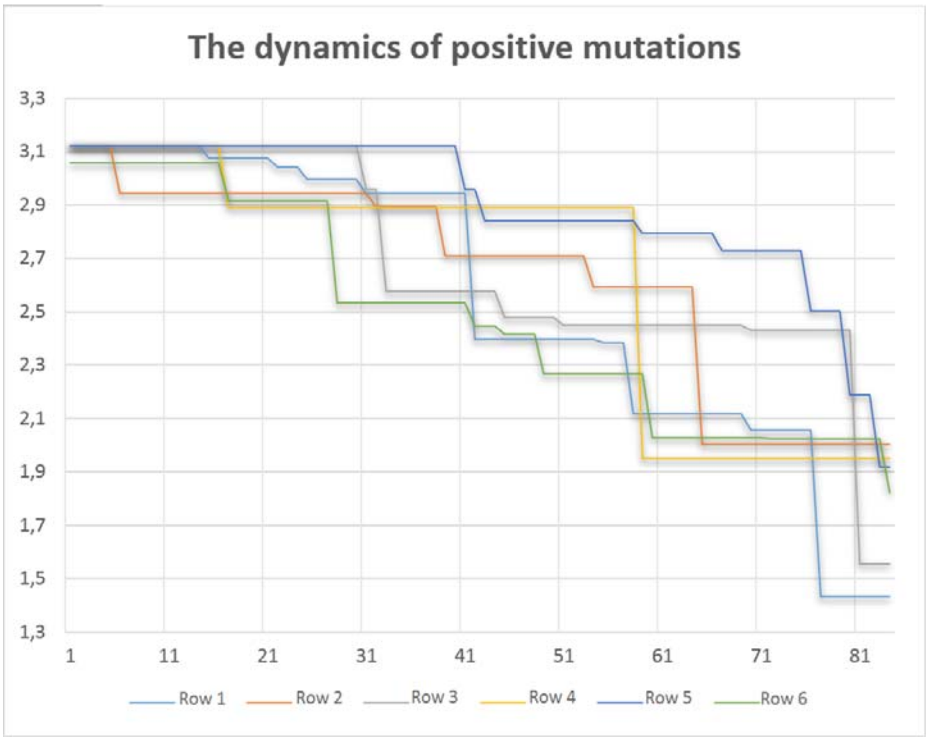


Fig. 5. SD change graph as a result of mutation steps

6 Conclusion

This article proposes a new method for constructing an analytical approximation of complex n-dimensional dependencies based on the use of the genetic algorithm.

The developed GA has the following features:

1. Search space encoding is performed as a parsing tree of an algebraic expression by the parser of the context-free grammar of the class LR(1);
2. In the course of the evolutionary process, in addition to the use of structure mutations (provided their positive effect), the stage of coefficients mutation is performed, which allows avoiding the target function falling into local extremes;
3. At each step of the evolutionary process there is a stage of searching for an extremum in the coefficient space and a stage of simplifying the analytical model.

The developed GA helped us solve the problem of accounting for transients in stationary non-isothermal gas transportation models. The resulting models were implemented as part of the software package for calculating the gas reserve on the main gas pipeline of OJSC "Gazprom transgaz Belarus», which reduced the impact of transients on the gas imbalance.

References

1. Glukhov A.O., Trofimov V.V.: Multi-agent structures for solving the traveling salesman problem. Problemy menedzhmenta: sb. nauchn, tr. Vypusk №3; pod obshch. red. prof. O.A. Strakhovoy, pp. 71–76. Izd-vo SPbGUEF, Saint-Petersburg (2000).
2. Trofimov V.V., Glukhov A.O.: Approximation on multi-agent structures. Ekonomicheskaya kibernetika: sistemnyy analiz v ekonomike i upravlenii: sb. nauchn. tr. Vypusk №1; pod red. D.V. Sokolova i N.N. Pogostinskoy, pp. 40–53. Izd-vo SPbGUEF, Saint-Petersburg (2000).
3. Glukhov A.O., Glukhov D.O., Trofimov V.V., Trofimova L.A.: Modified hybrid genetic algorithm of discreet optimization problems. 2017 XX IEEE International Conference on Soft Computing and Measurements (SCM 2017). Saint-Petersburg (2017).
4. Pozharskiy D. A., Zolotov N.B., Semenov I.E.: Genetic algorithm for finding the approximation coefficients of a function in contact problems for a cylinder. Molodoy uchenyy №24 (158), pp. 122–125 (2017).
5. Kildiushov M.S.: A program for reconstructing approximated algebraic functions from several variables from a set of discrete values of a function. Online magazine «NAUKOVEDENIE» Vol. 7, №5, (2015) http://naukovedenie.ru/PDF/136TVN515.pdf. doi: 10.15862/136TVN515
6. Kildiushov M.S.: The use of genetic algorithms for the restoration of approximated algebraic functions with a certain accuracy. Nauka i biznes: puti razvitiya. – Fond razvitiya nauki i kultury (Tambov) № 1 (55), pp. 25–31 (2016). ISSN: 2221-5182
7. Driankov D., Hellendoorn H., Reinfrank M.: An introduction to fuzzy control. Springerverlag (1993).
8. Lotfi A. Zadeh: Fuzzy Sets. Information & Control vol. 8, pp. 338–353 (1965).
9. Glukhov D.O.: Dynamic expert system by fuzzy inference rules to automations an examination of complex objects. Budownictwo i Inzynieria Srodowiska. – Zielonogorsk: Politechnika Zielonogorska, pp. 105–109 (1998).

10. Glukhov D.O., Glukhova T.M., Kundas S.P.: Soft calculations for organization of computer representation of nomograms on the example of calculating the limit creep coefficient. Vestnik Polotskogo gosudarstvennogo universiteta. – Series C: Basic Sciences, №3. pp. 2–6 (2010).
11. Zvonkov V.B., Popov A.M.: Comparative study of classical optimization methods and genetic algorithms. Vestnik Sibirskogo gosudarstvennogo aehrokosmicheskogo universiteta im. Akademika M.F. Reshetneva, pp. 23–27. Izdatelstvo: Federalnoe gosudarstvennoe byudzhetnoe obrazovatelnoe uchrezhdenie vysshego obrazovaniya «Sibirskij gosudarstvennyj universitet nauki i texnologij imeni akademika M.F. Reshetneva», Krasnoyarsk (2013). ISSN: 1816-9724