



I T H E A



International Journal
INFORMATION THEORIES
&
APPLICATIONS



2014 **Volume 21** **Number 4**



International Journal
INFORMATION THEORIES & APPLICATIONS
Volume 21 / 2014, Number 4

Editorial board

Honorable editor: **Victor Gladun** (Ukraine)
 Editor in chief: **Krassimir Markov** (Bulgaria)

| | | | |
|------------------------------|------------|----------------------------|------------|
| Adil Timofeev | (Russia) | Luis F. de Mingo | (Spain) |
| Aleksey Voloshin | (Ukraine) | Lyudmila Lyadova | (Russia) |
| Alexander Eremeev | (Russia) | Martin P. Mintchev | (Canada) |
| Alexander Kleshchev | (Russia) | Natalia Bilous | (Ukraine) |
| Alexander Palagin | (Ukraine) | Natalia Pankratova | (Ukraine) |
| Alfredo Milani | (Italy) | Nikolay Zagoruiko | (Russia) |
| Arkadij Zakrevskij | (Belarus) | Rumyana Kirkova | (Bulgaria) |
| Avram Eskenazi | (Bulgaria) | Stoyan Poryazov | (Bulgaria) |
| Boris Fedunov | (Russia) | Tatyana Gavrilova | (Russia) |
| Constantine Gaidric | (Moldavia) | Valeriya Gribova | (Russia) |
| Galina Rybina | (Russia) | Vasil Sgurev | (Bulgaria) |
| Hasmik Sahakyan | (Armenia) | Vitalii Velychko | (Ukraine) |
| Iliia Mitov | (Bulgaria) | Vitaliy Lozovskiy | (Ukraine) |
| Juan Castellanos | (Spain) | Vladimir Donchenko | (Ukraine) |
| Koen Vanhoof | (Belgium) | Vladimir Jotsov | (Bulgaria) |
| Krassimira B. Ivanova | (Bulgaria) | Vladimir Ryazanov | (Russia) |
| Levon Aslanyan | (Armenia) | Yevgeniy Bodyanskiy | (Ukraine) |

International Journal "INFORMATION THEORIES & APPLICATIONS" (IJ ITA)
is official publisher of the scientific papers of the members of
the ITHEA International Scientific Society

IJ ITA welcomes scientific papers connected with any information theory or its application.

IJ ITA rules for preparing the manuscripts are compulsory.

The **rules for the papers** for IJ ITA are given on www.ithea.org.

Responsibility for papers published in IJ ITA belongs to authors.

General Sponsor of IJ ITA is the **Consortium FOI Bulgaria** (www.foibg.com).

International Journal "INFORMATION THEORIES & APPLICATIONS" Vol. 21, Number 4, 2014

Edited by the **Institute of Information Theories and Applications FOI ITHEA**, Bulgaria, in collaboration with:

Institute of Mathematics and Informatics, BAS, Bulgaria,

V.M.Glushkov Institute of Cybernetics of NAS, Ukraine,

Universidad Politécnica de Madrid, Spain,

Hasselt University, Belgium,

St. Petersburg Institute of Informatics, RAS, Russia,

Institute for Informatics and Automation Problems, NAS of the Republic of Armenia

Printed in Bulgaria

Publisher ITHEA®

Sofia, 1000, P.O.B. 775, Bulgaria. www.ithea.org, e-mail: info@foibg.com

Technical editor: **Ina Markova**

Copyright © 1993-2014 All rights reserved for the publisher and all authors.

© 1993-2014 "Information Theories and Applications" is a trademark of ITHEA®

© ITHEA is a registered trade mark of FOI-Commerce Co.

ISSN 1310-0513 (printed)

ISSN 1313-0463 (online)

ISSN 1313-0498 (CD/DVD)

INTELLECTUAL INFORMATION SUPPORT OF BRANCH ENTERPRISE EXECUTIVES' DECISION MAKING PROCESSES

Aleksey Voloshyn, Bogdan Mysnyk, Vitaliy Snytyuk

Abstract: *The optimization problem of functioning of the producing similar products enterprise is considered. For its solution proposed multiagent technology, the use of which will make informed decisions about the expansion of production, reduction, and so branching. Models of the enterprise functioning in a competitive environment are constructed.*

Keywords: *enterprise, interaction, models, multiagent technology.*

ACM Classification Keywords: *I.2.11 Distributed Artificial Intelligence – Multiagent Systems.*

Introduction

Current economic realities characterized by even slowly but steadily growing development of small and medium enterprises. Information technologies and its dominant influence on decision-making processes lead to the emergence and functioning of an increasing number of e-business enterprises. At a time when on the market are dominated major natural and artificial monopolies, and are a growing number of small businesses, for which by the effective organization of production continues to be a high rate of profit. But for all unchanging factors values that affect on enterprises activities, this rate has been steadily decreasing, due to the saturation of the market. There is the problem of businessmen for the development or production cuts. And it is particularly relevant to the industry, which, as you know, is the set of firms producing similar (identical) products. Today it is the production of doors, plastic windows, and provision of buildings repair, their construction, e-commerce appliances and more.

It is known, progress in any sphere of human activity is realized as a result of an idea or need. And it is in business at an early stage of the life cycle of business activity, these two concepts are closely integrated into each other. Stage of functioning has more realistic aspects. It is based on just the need, need for profit to pay salaries, social security, implementation of environmental activities and more. It is important that at this stage of accumulating and developing capital assets created some capital. Due to various circumstances, any enterprise and its manager is faced with the dilemma of liquidation or development. The solution of this problem stems from the necessity but the corresponding areas, mainly based on the ideas.

Analysis of recent research and results

The use of multiagent systems does not total also 20 years old. The basic idea posited in its functioning, is the implementation of autonomous software agents that are able to accept the situation, make decisions and interact with their own kind. Thus, the solution of any complex problem emerges in an evolutionary way by agents that continuously compete and cooperate with one another.

Multiagent technology applied in such industries and firms in UK to manage: tanker fleet (Tankers International, London); corporate taxi fleet (Addison Lee, London); fleet of freight cars (Gist, Manchester), the provision of

vehicles for hire (Avis, Liverpool); in solution of problems related to: aerospace studies, intelligent transportation, smart factories functioning mobile teams of rescuers, working railroad and logistics.

Multiagent technologies based on the principles of self-organization and evolution of behavior characteristic of living systems as ant colonies and swarms of bees.

Multiagent technology capable to solving problems of planning and resource optimization, pattern recognition, comprehension of texts under the scheme: initialize the system load model monitoring the current situation analysis of the problem situation refine your monitoring of resource allocation plans expected result.

Unlike classical systems, the MAC is a large network of small agents operations are performed in parallel with the evolution of the place and the conditions for the development of [Gurevich, 2005].

It is important to note the complexity and dynamics of decision making in the management of production in real time. And with such problems multiagent technology can prove itself best [Ivashchenko, 2011].

Author [Skobelev, 2003] provides multiagent systems for timely processing of information and operational models of network requirements and capabilities. Competing elements of each of them in operation are capable to find optimal solutions. In [Masloboev, 2011] proposed multiagent information technology support decision making in the management of quality educational services for regional research and education complex, designed simulation model of quality management education, where agents are "student" and "teacher".

There are researches where for simulation of industrial enterprises activities applied the ideas and principles of "artificial life" by using neural network technologies to train agents and providing them with the properties of memory and prediction [Snytyuk, 2010].

Multiagent systems built and to the development of telecommunications management, where agents are companies, competitors, some state participants, agents of the state policy in the field of telecommunications, local government agents. The advantages of this multiagent technology are the effectiveness of strategic management by increasing the validity of decisions, taking into account a number of factors, and the analysis of different scenarios of interaction of the telecommunications market participants.

Formalized statement of the problem and its solution

How can multiagent technologies help to solve the problem of optimizing the life cycle processes of the industry? As these enterprises produce the same product, then we consider two cases:

- Products durable, resulting in market saturation;
- Products that have a limited useful life, requiring replacement, and the market demands constant with minor fluctuations the number of goods in time.

Problem being considered in the paper is to maximize the efficiency of the enterprise sector, defined range of tasks, production structure and management strategy that consists in allocating resources [Voloshyn, 2013].

Consider the features of the first case of production. Suppose that in a particular area, there is a need for products P , and $|P| = N$, but after a long time the product needs replacement or repair, so $|P| = N + \delta$, where δ - some positive integer. It can be argued that the required number of products that must stand up for the first time and the number of consumer products that must be replaced are values that depend on time, i.e. $N = N(t), \delta = \delta(t)$, with a monotonous no increasing function $N(t)$ and $\delta(t)$ - monotone no decreasing

function. Functions $N(t)$ and $\delta(t)$ can be partially or fully known to producers, consumers and analysts. In addition, it is possible the case when only known expert assumptions about their structure and parameters. So for the moment t , the number of products to market saturation is $N(t) + \delta(t)$.

Assume that the products P produced by M enterprises. Each of them can be represented by an agent acting for a certain application $V_i, i = \overline{1, M}$. The result of the program is a recommendation for a decision maker, or directly the solution. We divide the time interval enterprise operation (T) at intervals $T = \{t_0 < t_1 < \dots < t_k < \dots\}$. We describe the essence of the points in time $t_i, i = \overline{1, L}$, where L - integer or infinity.

Each company will present how a certain system S . Operation of the system is the continuous-discrete process defined by the function

$$F(t) = \begin{cases} f_1(t), t \in [t_0, t_1], \\ f_2(t), t \in [t_1, t_2], \\ \dots\dots\dots, \\ f_k(t), t \in [t_{k-1}, t_k], \\ \dots\dots\dots \end{cases}$$

Functions $f_i(t)$ determines the efficiency of the system: profit, cost of production, capital-like. Transitions $f_i(t) \rightarrow f_{i+1}(t)$ occur as a result of decisions taken at the times t_i . Determine which factors influence the occurrence of such values t_i . To this consider the system S as part of a higher level of hierarchy, interacting with it, and makes an impact.

Let Ω is the system of higher hierarchy level. There is an interaction between S and Ω , which is expressed in entering in S the material flows (H), power (E), information (I), finance (U), human resources (R) and in production (P) and data (D) coming in Ω from S (Fig. 1). Since all systems $S_i, i = \overline{1, M}$, perform the following exchange with Ω , it is obvious that there are implications $S_i \rightarrow \Omega \rightarrow S_j, S_j \rightarrow \Omega \rightarrow S_i, i \neq j$. Note that each implication $G_{ij} : S_i \rightarrow S_j$ has a different quantitative and qualitative features. |This interchange is done on the already mentioned seven levels. There is an implication $\Omega \rightarrow S_j = S_j^{in}(H_j, E_j, I_j, U_j, R_j)$, where $S_j^{in}(\ast)$ – incoming flows of system S_j . By system S_j performed implication $S_j^{in}(\ast) \rightarrow S_j^{out}(P_j, D_j)$ and the result is output to the system Ω . Thus, there is a chain of implications

$$\Omega \rightarrow S_j^{in}(H_j, E_j, I_j, U_j, R_j) \rightarrow S_j^{out}(P_j, D_j) \rightarrow \Omega, \forall j = \overline{1, M}. \tag{1}$$

But implementation of implications (1) takes time and during that time, changing the system Ω . In addition, the transformation (1) is closed, which can be represented formally as follows:

$$\begin{aligned} \Omega_{in}^i &\rightarrow \Omega_{out}^i, \\ \Omega_{in}^{i+1} &= V(\Omega_{out}^i) = V((P_j, D_j), \forall j = \overline{1, M}). \end{aligned} \tag{2}$$

Expressions (2) indicate that the system Ω is constantly changing, the main influence on it realize the activity results S_j in a complex operation, often obtained by transformation V finance or new information. This, in turn, determines the performance of transformation (1). Considering (1) and (2), we can conclude that the interaction of

the system Ω is carried out with each S_j and systems S_j of each other. As a result are changes in the input flows next (s) period (s) of time occurred. Therefore, we assume that there comes a time t_i when $\exists S_j$, or in what comes $H_j \vee E_j \vee I_j \vee U_j \vee R_j$, or from which is obtained $P_j \vee D_j$, with this time $t_i > t_{i-1}$ and it is the minimum $\forall S_j, j = \overline{1, M}$.

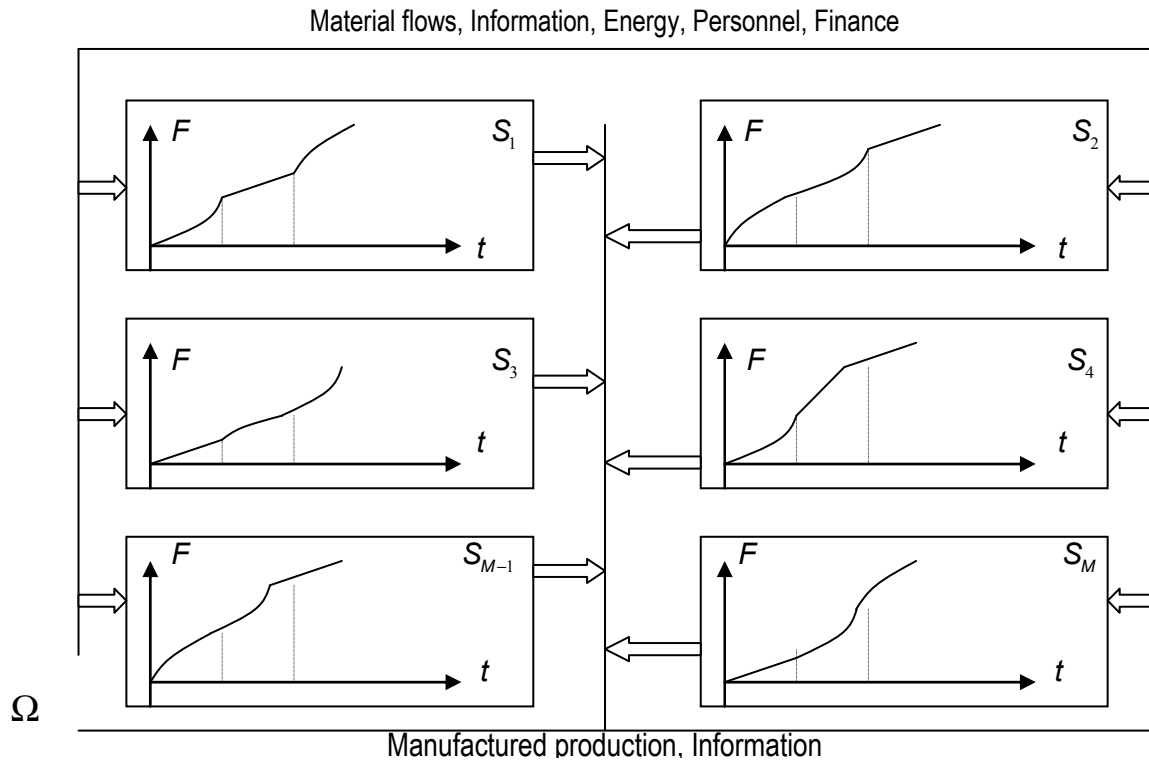


Figure 1. Companies interaction as multiagent technology

In Figure 1 also shows how to change the function $f_i(t)$ at different time intervals. In particular, it can be regarded as characteristic number of products, as evidenced by the monotony of imprinted features.

Thus, multiagent system will take into account information from the outside at times $t_i, i = \overline{0, L}$, which will allow it to carry out adjustment of the elements of production.

Functioning features of multiagent system

Obviously, the simulation is done in the interest of an enterprise. We assume that firms operate in the market industry for a long time. Initial time t_0 of modeling and analysis of the real situation is known. For the moment t_0 initialization of multiagent system is performed. In practice this means setting the values of the states as well system Ω as systems S_j that are viewed as agents. The basic parameters of the systems S_j are:

- Number of raw materials in stock;
- Number of units that may be made with this material;

- The time at which all other things being equal need for a manufacturing unit, or a certain number of units by parallel production;
- The value of fixed assets;
- Number of employees;
- The cost of production;
- Other.

For the system Ω main features are:

- Market demand in a number of products;
- Legal restrictions;
- The financial situation of the company (the amount of money in the account);
- Payables and liabilities.

Obviously, these indicators and characteristics of the whole range are not limited, but the scope of this paper does not allow carrying out them a full recalculation also because of the characteristics of each industry.

After initializing the values of key indicators occurs the process of modeling, which is done loading models (agents). These models reflect indicator of activity efficiency for systems S_j , each performance indicator is the criterion of the system S_j performance of one its functions. The construction of such functions is based on retrospective data. In addition to assessment of the actual state (at the time) this functions allow you to analyze situations "and if A, then ..." in the future. Obviously, the best option simulation corresponds to the absence of critical situations. But in the multiagent system occurrence of any such events should be reflected in the knowledge base along with the production rules of appropriate actions. The occurrence of such extreme events corresponds to one of the points t_i in time of decision making.

The next step after the initialization state enterprises need to make a boot a model of their operation. Here are the main components of this model and define the features of its use. Assume that in the interval (t_{k-1}, t_k) the functioning each of enterprises will be continuous. Then the general model is as follows:

$$f_k^{ij}(t), k = \overline{1, L}, i = \overline{1, W}, j = \overline{1, V},$$

$$\text{if } t = t_k + \delta, \text{ then the transition } f_k^{ij}(t) \rightarrow f_{k+1}^{ij}(t), \quad (3)$$

where k - number of time period, i - number of the enterprise, j - number of functions (efficiency indicators), δ - small enough positive number. Functions $f_k^{ij}(t)$ may have a different specification. If $f_k^{ij}(t)$, for example, is the number of issued goods, then $f_k^{ij}(t) = \text{const} \vee 0, 5t + 7 \vee 2t^2 + 3t + 1 \dots$ In the first case, the number of produced goods is constant, in the second - a linear growth, in the third - is the quadratic dependence and $t \in (t_{k-1}, t_k)$.

Assuming that produced goods are sent to the warehouse and from there comes to consumers and the storehouse has a limited capacity, the subset of models describing the activities of the enterprise is as follows:

$$f_k^{ij}(t) = kt, k > 0 \text{ (offer),}$$

$$f_k^{ij+1}(t) = lt, l > 0 \text{ (demand),}$$

$$f_k^{ij+2}(t) = (k - l)t \text{ (number of products in storehouse).}$$

For example: $f_k^{ij}(t) = 3t, f_k^{ij+1}(t) = 2t, f_k^{ij+2}(t) = t$ (Figure 2a).

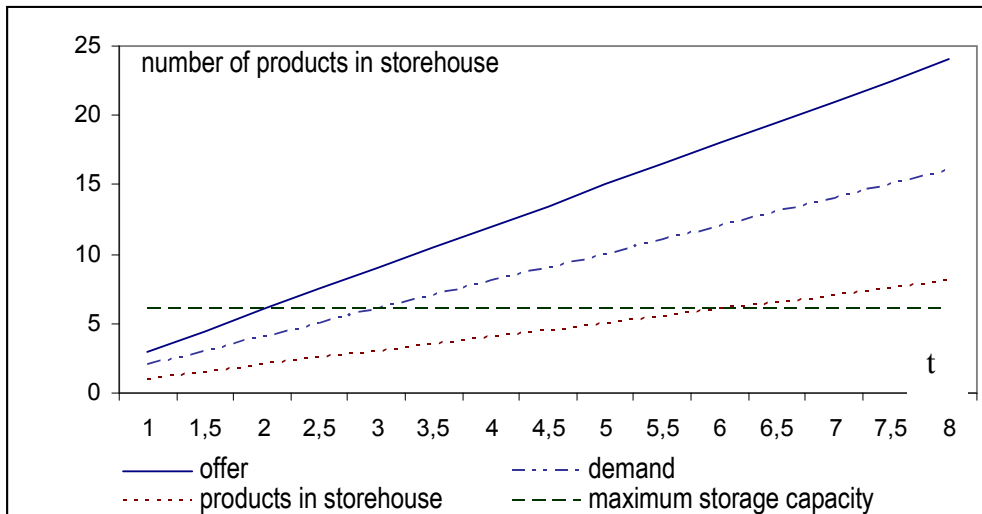
If the demand is decreasing then models are as follows:

$$f_k^{ij}(t) = kt, k > 0 \text{ (offer),}$$

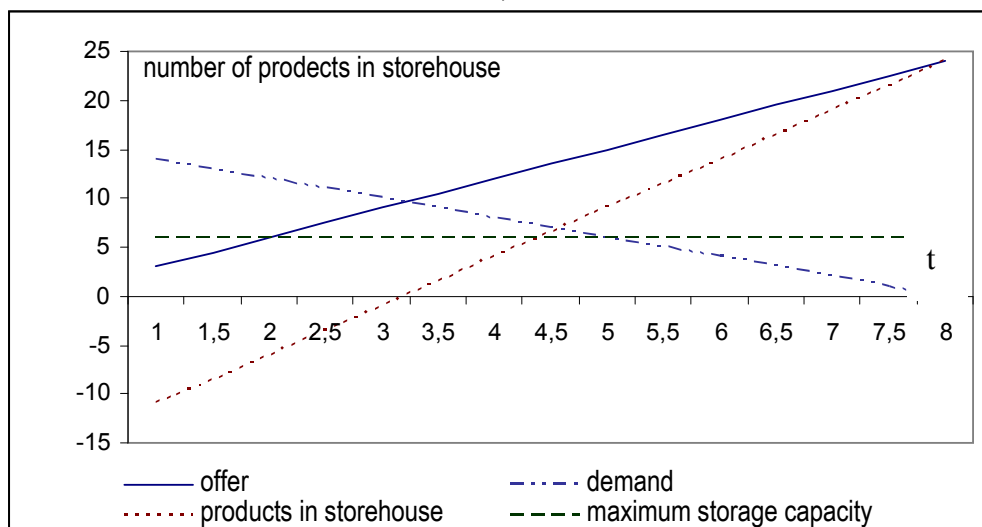
$$f_k^{ij+1}(t) = lt + a, l < 0 \text{ (demand),}$$

$$f_k^{ij+2}(t) = (k - l)t - a \text{ (number of products in storehouse).}$$

For example: $f_k^{ij}(t) = 3t, f_k^{ij+1}(t) = -2t + 16, f_k^{ij+2}(t) = 5t - 16$ (Figure 2b).



a)



b)

Figure 2. Enterprise activity and features dynamics

It is obvious that in the cases in:

- Figure 2a - at time $t^* = 6$ and at time $t^* = 4, 4$;
- Figure 2b - number of products in storehouse up to the maximum capacity of staff,

it is necessary to make a decision that $\exists k \in N: t^* = t_k$. It is similarly constructed as other models and defined decision points. Note that for many models there will be dependence

$$f_k^{ij}(t) = G_{k-1}^{ij}(f_{k-1}^{ij}(t)), \tag{4}$$

indicating that the dependence of certain features of i^{th} enterprise on k^{th} time interval of the same function of ij^{th} on $(k-1)^{\text{th}}$ time interval, $ij \neq i$. Without loss of generality, we consider that the exchange of information between companies, enterprises and the environment about the features and results of activity occurs in moments time t_k . Dependence (4) for most companies allows generalization:

$$f_k^{ij}(t) = G_{k-1}^{\Lambda(i)j}(f_{k-1}^{\Lambda(i)j}(t)), \tag{5}$$

where $\Lambda(i)$ is a set of enterprises, and j^{th} function of which affect on j^{th} function of i^{th} enterprise. Expression (5) indicates that, as j^{th} function of i^{th} enterprise depends on the values of j^{th} function of enterprises from the set $\Lambda(i)$ of companies in $(k-1)^{\text{th}}$ time interval.

Consider a number of models on which implemented the functioning of enterprises. In particular:

$$f_k^{ij}(t) = G_{\Psi(k,\lambda)}^{ij}(f_{\Psi(k,\lambda)}^{ij}(t)), \tag{6}$$

which shows that the value of ij^{th} function of i^{th} enterprise depends on the values of j^{th} function of ij^{th} enterprise in times $\{t_{k-\lambda}, t_{k-\lambda+1}, t_{k-1}\}$;

$$f_k^{ij}(t) = G_{\Psi(k,\lambda)}^{\Lambda(i)j}(f_{\Psi(k,\lambda)}^{\Lambda(i)j}(t)), \tag{7}$$

where the function G indicates the existence of values of j^{th} function for i^{th} enterprise depending on the values of j^{th} function of enterprises from the set at the time points $\{t_{k-\lambda}, t_{k-\lambda+1}, t_{k-1}\}$;

$$f_k^{ij}(t) = G_{k-1}^{ij}(f_{k-1}^{ij}(t)), \tag{8}$$

which shows the relationship between j^{th} function of i^{th} enterprise and ij^{th} function of ij^{th} enterprise on the previous time period;

$$f_k^{ij}(t) = G_{k-1}^{\Lambda(i)\Theta(j)}(f_{k-1}^{\Lambda(i)\Theta(j)}(t)), \tag{9}$$

which indicates on a relationship between the values of j^{th} functions of i^{th} enterprise and of the functions from the set $\Theta(j)$ of enterprises group $\Lambda(i)$ in the previous step;

$$f_k^{ij}(t) = G_{\Psi(k,\lambda)}^{\Lambda(i)\Theta(j)}(f_{\Psi(k,\lambda)}^{\Lambda(i)\Theta(j)}(t)), \quad (10)$$

which shows that the value of j^{th} function of i^{th} enterprise on k^{th} time interval depends on the values of functions from the set $\Theta(j)$ of enterprises group $\Lambda(i)$ at previous time intervals $\{t_{k-\lambda}, t_{k-\lambda+1}, t_{k-1}\}$;

Assume that all models that determine the behavior of the agent are formed. Structural and parametric identification model is based on a database (DB) for the actual point in time, the bank mathematical models (BMo) and a set of identification methods (BMe). Thus, the agent in formation stages served as a set

$$A = \langle \text{DB}, \text{BMo}, \text{BMe} \rangle \quad (11)$$

Method of agent forming will have the following steps:

Step 1. Establish the range of internal and external features of the system (*In*, *Out*).

Step 2. Let $|In| = n$, $|Out| = m$.

Step 3. For $i = 1$ to n do (to internal features):

Step 3.1. For all the data with capacity $|DB|$ to do:

Step 3.1.1. For $k = 1$ to $|BMo|$ do:

Step 3.1.1.1. For $l = 1$ to $|BMe|$ do:

Step 3.1.1.2. Implement structural and parametric identification of i^{th} internal feature of system based on data from *DB* using k^{th} model and l^{th} method, if possible.

Step 3.1.1.3. End of cycle.

Step 3.1.2. End of cycle

Step 3.2. End of cycle.

Step 4. Among all the obtained models by certain criteria to choose the best and consider it one of the programs under which the agent operates.

Step 5. Perform steps 3-4 items for external features.

As a result of the above method for the entire sector enterprises will be provided a set of agents (multiagent system), which will have a specific architecture and program operation. Note that the architecture of the agents has minor differences, but the program will vary much more.

In the next stage of multiagent technology will offer a method for operation of multiagent systems:

Step 1. Initialize the multiagent system using the above method, $i = 0, t = t_0$.

Step 2. To monitor the environment that incident to sector enterprises

Step 3. If in the environment held at least one change, then $i = i + 1, t = t_i$, and to record to DB.

Step 4. If changes not occurred but ended the period of monitoring, to record the values of features enterprise characteristics to DB

Step 5. If any of the values of internal features of one of the systems has changed, due to a change in the spectrum of problems, management strategy or structure of production that do not affect the changes in the environment, then record the appropriate entries to DB, $i = i + 1, t = t_j$.

Step 6. Perform structural and parametric identification of the model based on the obtained data.

Step 7. If obtained information and models usage suggest the possibility of critical modes, to warn the decision maker.

Thus, the results of the methods for forming and using agents allow the information support of the decision maker. In practice, there is uncertainty, which is caused by the variety by different purposes of the different decision makers, no significant amount of data about the results of the enterprise activity. These problems can be partially overcome by insider information or analysis of simulation results under certain assumptions, integrating features of enterprises-competitors and implementation assumptions about the interaction with them as unconscious opponent (nature).

Idealized object of study and believing that one is in the interest of the enterprise, you can get the conclusions that will be the basis for decision making processes for relevant decision maker. It is necessary to carry out continuous analysis and monitoring of the production market. Knowing the extent and dynamics of demand, as well as the dynamics of its features, you can make informed decisions about further development. These solutions are solutions of the synthesis problem. Dual to it is the problem of the analysis, which in this case is to determine the consequences of decisions based on the results that are predicted on the basis of models.

Fragmented model of sector market

Analyzing the effective functioning of a particular enterprise and across the industry, it is necessary to consider a set of features. Assuming that enterprise performance is determined by their values, we assume that

$$S_j = \langle X_j^1, X_j^2, \dots, X_j^{d1} \rangle, \tag{12}$$

where S_j - is an j^{th} enterprise, (X^1, X^2, \dots, X^d) - the features of the enterprise, and

$$S_j^t = (x_j^{1t}, x_j^{2t}, x_j^{dt}), \tag{13}$$

where t - time, moreover $\exists k \in [1; +\infty): t \in [t_{k-1}, t_k], x_j^{it} - X^i$ feature value of j^{th} enterprise at the time t .

Formulae (12) and (13) give the possibility of constructing a fragmented sector model, a two-dimensional version of which is shown in Figure 3. Generally, such a model can be written as

$$F_M = \langle X_1^1, X_1^2, \dots, X_1^d, X_2^1, X_2^2, \dots, X_2^d, \dots, X_M^1, X_M^2, \dots, X_M^d \rangle \tag{14}$$

Thus, the fragmented model is multidimensional $(M+1)$ rectangular hyper parallelepiped illustrating the trajectory of the enterprise activity in the space of its internal and external features. The length of one side of the hyper parallelepiped increases, since it corresponds to the time. Each cell of the model corresponds to a period of time, which did not change the values features of the enterprise. If at least for one company they have changed, then $t_k \rightarrow t_{k+1}$ and in the model there is another band that reflects the new time interval. In addition, due to a variety of enterprises and the values of their features, we claim that no two trajectories of functioning enterprises pass

through some cell. Note that there are restrictions, i.e. $\forall X_j \in [X_{jmin}, X_{jmax}]$. If the value of at least one features beyond the established limits, then holds an exclusive option functioning. The relevant enterprise is excluded from consideration and requires individual solutions.

Fragmented model is the basis for the preliminary analysis of the overall state on the market. Its practical application associated with the use of technology OLAP (online analytical processing), performance cuts of model, bringing it to a smaller dimension. These operations allow determining the range of features that are informing and influencing on the overall efficiency of the activity. The decision maker's solutions would affect not only the optimization of the enterprise activity, but also adjusting model parameters and process control planning in the company.

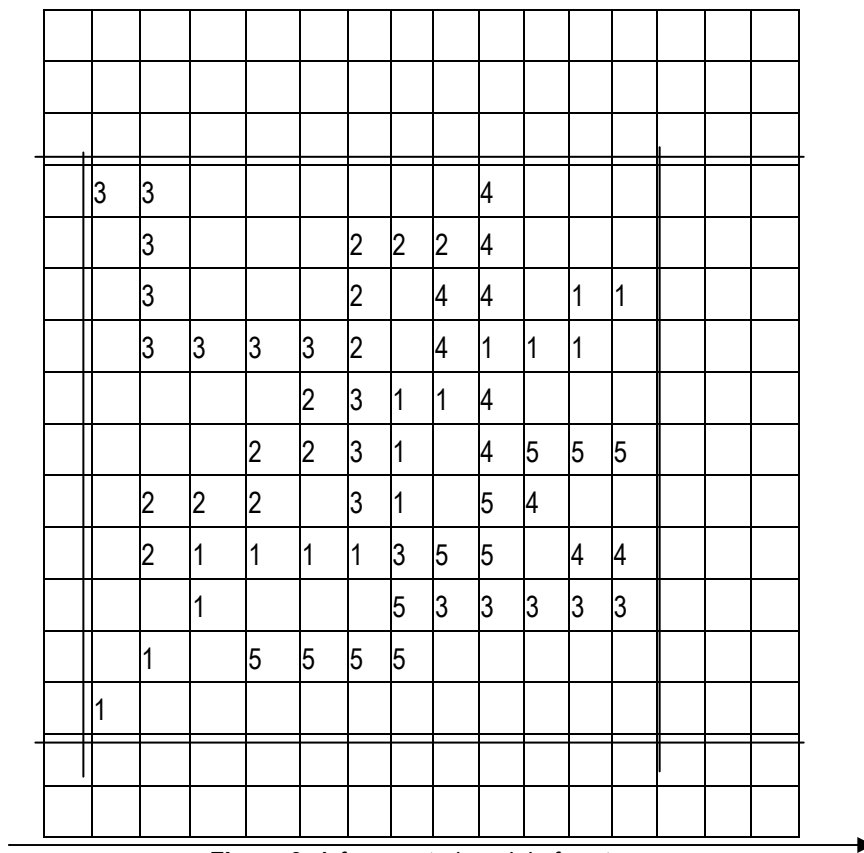


Figure 3. A fragmented model of sector

Conclusions and perspectives

The functioning of natural systems based on the principles of self-organization. Most of them can be used to optimize the activity and artificial systems, including industrial enterprises. Proposed in this paper multiagent technology corresponds both a natural mechanism and functioning elements of the industry. The need to adhere to market principles of the economy makes it necessary decisions for each enterprise according to the performance of businesses and other environmental conditions. The suggested technology allows to the decision maker for the benefit of one company to act in accordance with the market situation: to develop production, eliminate it, improve the competitiveness of products and so on.

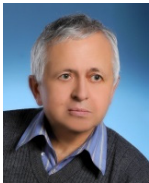
Acknowledgements

The work is published with the financial support of the project ITHEA XXI Institute of Information Theory and Applications FOI ITHEA Bulgaria www.ithea.org Association and the creators and users of intellectual systems ADUIS Ukraine www.aduis.com.ua.

Bibliography

- [Gurevich, 2005] Л.А. Гуревич, А.Н. Вахитов. Мультиагентные системы // Computer Science. – 2005. – С. 116-139.
- [Ivashchenko, 2011] А.В. Иващенко. Мультиагентные технологии для управления производством в реальном времени. Режим доступа: www.smartsolutions-123.ru.
- [Nasluboev, 2011] А.В. Маслубоев, В.В. Быстров, А.В. Горохов. Мультиагентная информационная технология поддержки управления качеством высшего образования // Вестник МГТУ. – 2011. – Том 14, № 4. – С. 854-859.
- [Skobelev, 2003] П.О. Скобелев. Открытые мультиагентные системы для оперативной обработки информации в процессах принятия решений : Дисс. ...докт. техн. наук. – Самара, 2013. – 418.
- [Snytyuk, 2010] В.Е. Снитюк, Б.В. Мыслик. Адаптация концепции «искусственной жизни» к моделированию процессов функционирования производственных предприятий//Журнал передовых технологий. – 2010. – № 4/4(46). – С. 4-8.
- [Voloshyn, 2013] А.Ф. Волошин, М.В. Коробова, Т.В. Колянова. Математическая экономика – Киев, 2013. – 224с.

Authors' Information



Aleksey Voloshyn – Taras Shevchenko National University of Kyiv, Professor, 03680, 4d Glushkov Ave, Kyiv, Ukraine; e-mail: ovoloshyn@ukr.net

Major Fields of Scientific Research: mathematical economics, decision-making theory, Intelligent information systems.



Bogdan Mysnyk – Cherkassy State Technological University, Assistant, 18006, 460 Shevchenko Ave, Cherkassy, Ukraine; e-mail: setne@list.ru .

Major Fields of Scientific Research: Multiagent technologies.



Vitaliy Snytyuk – Taras Shevchenko National University of Kyiv, Professor, 03022, 81 Lomonosov Str., Kyiv, Ukraine; e-mail: Snytyuk@gmail.com

Major Fields of Scientific Research: Decision making under uncertainty.

A HIERARCHICAL APPROACH TO MULTICRITERIA PROBLEMS

Albert Voronin, Yuriy Ziatdinov, Igor Varlamov

Abstract: It is shown, that any multicriteria problem can be represented by a hierarchical system of criteria. Individual properties of the object (alternative) are evaluated at the bottom level of the system, using a criteria vector; and a composition mechanism is used to evaluate the object as a whole at the top level. The problem is solved by the method of nested scalar convolutions of vector-valued criteria. The methodology of the problem solving is based on the complementarity principle by N. Bohr and the theorem of incompleteness by K. Gödel.

Keywords: hierarchical structure, nested scalar convolutions, multicriteria approach, decomposition; composition

Introduction

The problem of decision making in general view can be represented by the scheme

$$\{\{x\}, Y\} \rightarrow x^*$$

where $\{x\}$ is a set of objects (alternatives); Y is the function of choice (rule establishing a prefer ability on a set of alternatives); x^* is the chosen alternatives (one or more).

The function Y is used to solve the problem of analysis and evaluation of alternatives. On results of estimation the choice of one or a few best alternatives from the given set follows. In decision theory, there are two different approaches to evaluating objects (alternatives) subject to choice. One of them is to evaluate an object as a *whole* and to choose an alternative by comparing objects as *gestalts* (holistic images of objects without detailing their properties). The second approach is detailed elaboration and assessment of various object vectors of properties and making decisions after comparing these properties. If a holistic approach implies choosing x^* directly using choice function Y , the *vector approach* requires a mechanism to carry out decomposition of Y into a set (vector) of the choice functions y . By decomposition of the choice function Y is understood its equivalent representation by a certain set of other functions y which composition is the initial choice function Y .

Separation of properties of alternatives on the basis of the analysis is the decomposition leading to the hierarchical structure of properties.

Properties, for which there exist objective numerical characteristics, are called *criteria*. The approach of comparison on separate properties, at all its attraction, derivates a serious problem of return transition to required comparison of alternatives as a whole [Voronin, 2013].

Statement of the Problem

Quality of an alternative is determined by hierarchical system of vectors

$$y^{(j-1)} = \{y_i^{(j-1)}\}_{i=1}^{n^{(j-1)}}, \quad j \in [2, m],$$

where $y^{(j-1)}$ is the vector of criteria on the $(j-1)$ -th level of the hierarchy, by the components of which the quality of properties of alternatives for the j -th level is assessed; m is the amount of levels of the hierarchy; $n^{(j-1)}$ is the amount of estimated properties on $(j-1)$ -th level of the hierarchy. The numerical values of n criteria $y^{(1)} = y$ of the first level of the hierarchy for the alternative are given.

The same criterion on $(j-1)$ -th level can participate in the evaluation of several properties of the j -th level, i.e. in the hierarchy are possible cross-links. It is clear that $n^{(1)} = \sum_{i=1}^{n_1} r_i = n$ and $n^{(m)} = 1$.

Importance (significance) of each of the components of the criterion of $(j-1)$ -th level in the evaluation of properties of k -th level is characterized by a property coefficient of the priority, their set forming the priority vectors system

$$P_{ik}^{(j-1)} = \{p_{ik}^{(j-1)}\}_{k=1}^{n^{(j)}}, j \in [2, m].$$

It is required to find an analytical evaluation y^* and qualitative evaluation of the effectiveness of this given alternative, and from the alternatives available to choose the best.

The Method of Solution

At the study, the approach is used consisting in the creation and simultaneous co-existence of not one but many theoretical models of the same phenomenon, and some of them conceptually contradict each other. However, no one can be neglected, as each describes a property of the phenomenon and none can be taken as a single because it does not express the full range of its properties. Compare the said with the principle of complementarity, introduced into science by Niles Bohr: "... To reproduce the integrity of the phenomenon should be used mutually exclusive "complementary" classes of concepts, each of which can be used in its own, special conditions, but only when taken together, exhaust the definable information". It is the principle of complementarity that allows for separating and then linking these criteria in multicriteria evaluation. Only a full set of individual criteria (vector criterion) enables an adequate assessment of the functioning of a complex system as a manifestation of the contradictory unity of all its properties.

However, this possibility represents only a necessary but not a sufficient condition for the vector evaluation of the entire alternative as a whole.

For a complete evaluation it is necessary to go out from the lower level of the hierarchy and to rise on the following tier, i.e. to carry out an act of criteria composition. Let's compare this with the incompleteness theorem of Kurt Gödel "... In every complex enough not contradictory theory of the first order there is a statement, which by means of the theory is impossible neither to prove, nor to deny. But the self-consistency of a particular theory can be established by means of another, more powerful formal theory of the second order. But then the question of the self-consistency of this second theory arises, and so forth". We can say that Gödel's theorem is a methodological basis for the study of hierarchical structures.

With reference to our problem it means that for an adequate estimation of an alternative as a whole we should solve a task of the criteria composition on levels of hierarchy, consecutively passing from the bottom level up to top.

A scalar convolution of criteria can serve as a tool for the act of composition. The scalar convolution – it is a mathematical technique for data compressing and quantifying its integral properties by a single number.

A scalar convolution on nonlinear compromise scheme for the criteria subject to be minimized is proposed [Voronin, 2014]

$$Y[y(x)] = \sum_{k=1}^s p_k A_k [A_k - y_k(x)]^{-1},$$

applied in cases where the decision-maker considers as the preferred those solutions in which the values of individual criteria $y_k(x)$ are farthest from their limit values, A_k . This convolution has a number of essential advantages, which include flexibility, universality and analyticity.

The choice of a compromises scheme is made by the decision-maker and appears as explicitly conceptual.

Nested Scalar Convolutions

It is proposed for analytical evaluation of hierarchical structures to apply a method of nested scalar convolutions. The composition is performed on the "matryoshka principle": the scalar convolutions of the weighted components of vector criteria of lower level serve as the components of the vectors of higher level criteria. Scalar convolution of criteria obtained at the uppermost level is automatically considered as the expression for the analytical evaluation of effectiveness of the entire hierarchical system.

The algorithm for nested scalar convolutions is represented by an iterative sequence of operations of the weighed scalar convolutions of criteria for each level of the hierarchy from the bottom up, taking into account the priority vectors, based on the selected compromise scheme

$$\{(y^{(j-1)}, p^{(j-1)}) \rightarrow y^{(j)}\}_{j \in [2, m]} \quad (1)$$

and the searching and evaluating of effectiveness of the entire hierarchical system (alternative) as a whole is expressed by the problem of determining the scalar convolution of criteria on the top level of the hierarchy:

$$y^* = y^{(m)}.$$

When using the recurrent formula (1) important is the rational choice of the compromise scheme. For the method of nested scalar convolutions the adequate is a nonlinear compromise scheme. It is established that, without loss of generality, a premise for its use is that all the partial criteria were non-negative, were subject to minimization and were limited:

$$0 \leq y_i \leq A_i, A = \{A_i\}_{i=1}^n,$$

where A is the vector of restrictions on the criteria of the current level of the hierarchy; n is the amount of them.

Preceding from (1) the expression to evaluate k -th property of an alternative for the j -th level of the hierarchy by using the nonlinear compromise scheme looks like

$$y_k^{(j)} = \sum_{i=1}^{n_k^{(j-1)}} p_{ik}^{(j-1)} [1 - y_{0ik}^{(j-1)}]^{-1}, k \in [1, n^{(j)}], \tag{2}$$

where criteria of the $(j-1)$ -th level are normalized (reduced to unity). Thus, $y_{0ik}^{(j-1)}$ are the normalized vector's $y_0^{(j-1)}$ components involved in the evaluation of properties of the k -th alternative on the j -th level of the hierarchy; $n_k^{(j-1)}$ is their amount; $n^{(j)}$ is the amount of evaluated properties of the j -th level.

In the most simple and rather common case the multicriteria problem is formulated and solved without priorities, when decision-makers believe that all the importance parameters for all properties of alternatives are the same. In this case, a simple scalar convolution with the nonlinear trade-offs scheme in a unified form is used.

In order to formula (2) reflected the idea of the nested scalar convolutions method in accordance with the recurrent relation (1), this expression should be normalized, i.e., must be obtained a relative measure such that it were subject to be minimized, and it were the unit for it as the limit value.

The structure of the nonlinear compromise scheme enables normalizing the convolution (2) not to the maximum (which in this case is difficult), but to the minimum value of criteria convolution. Indeed, the ideal values for the criteria that are subject to be minimized are their zero points. Putting in (2)

$$y_{0ik}^{(j-1)} = 0, \forall i \in [1, n_k^{(j-1)}]$$

and taking into account the normalization $\sum_{i=1}^n p_i = 1$, we obtain $y_{k\min}^{(j)} = 1$.

After calculations and normalizing (reducing to unity), the final expression for the recurrent formula for calculating analytical assessments of the alternatives properties at all levels of the hierarchy becomes

$$y_{0k}^{(j)} = 1 - \left\{ \sum_{i=1}^{n_k^{(j-1)}} p_{ik}^{(j-1)} [1 - y_{0ik}^{(j-1)}]^{-1} \right\}^{-1}, k \in [1, n^{(j)}], j \in [2, m].$$

Conclusion

The foregoing leads to the conclusion that any problem of vector assessment of an alternative can be represented by a hierarchical system of criteria, resulting from the decomposition of alternative properties. The lower level of the hierarchy is an object (alternative) assessment on selected properties, using initial criteria vector, and the upper level is obtained through the mechanism of the composition as a whole object evaluation. Central here is the problem of the composition of criteria for levels of the hierarchy to be solved by the method of nested scalar convolutions.

The methodological basis of an alternative properties decomposition to obtain the initial criteria vector is the Bohr's principle of complementarity. This is a *necessary* condition for vector estimation of alternatives.

The methodology of a criteria composition for levels of the hierarchy is based on the Gödel's theorem of incompleteness. This is a *sufficient* condition for vector estimation of alternatives.

We dare say that above inferences about notions of criteria decomposition and composition can be extended on the more general notions of analysis and synthesis.

Bibliography

[Voronin, 2013] Albert Voronin and Yuri Ziatdinov, “Theory and practice of multicriteria decisions: Models, methods, realization”, Lambert Academic Publishing, 2013, [in Russian].

[Voronin, 2014] Albert Voronin, “Multicriteria Decision-Making. Lambert Academic”, Publishing, 2014.

Authors' Information



Albert Voronin – professor, DrSc(Eng), Professor of Chair of Computer Information Technologies of National Aviation University of Ukraine; e-mail: alnv@voliacable.com



Yuri Ziatdinov – professor, DrSc(Eng), Head of Chair of Computer Information Technologies of National Aviation University of Ukraine; e-mail: oberst@nau.edu.ua



Igor Varlamov – PhD, doctoral of National Defence University of Ukraine named after Ivan Cherniakhovsky; e-mail: igor_varlamov0@rambler.ru

SOFTWARE EFFORT ESTIMATION USING RADIAL BASIS FUNCTION NEURAL NETWORKS

Ana Maria Bautista, Angel Castellanos, Tomas San Feliu

Abstract: *One of the biggest challenges that software developers face is to make an accurate estimate of the project effort. Radial basis function neural networks have been used to software effort estimation in this work using NASA dataset. This paper evaluates and compares radial basis function versus a regression model. The results show that radial basis function neural network have obtained less Mean Square Error than the regression method.*

Keywords: *software effort estimation, software repositories, radial basis function and artificial neural networks.*

ACM Classification Keywords: *1.2.6 Artificial Intelligence – Connectionism and neural nets, H.2.7 Database Administration – Data Ware house and repository.*

Introduction

The software projects are complex products of engineering which include many resources and the value of any must be accurate to make project not be defeated.

The reason for an emphasis on software effort estimation is that it provides essential part of the foundation for project management. Without a reasonably effort estimation capability the software projects often experience a lot of problems. An incorrect assumption of the software projects resources may lead the software projects to undesirable results. For any software organization, accurate estimation of effort is crucial for successful management and control of software project. In other words, in any software effort estimation, making an estimate of the person-months and the duration required to complete the project, is very important [Malhotra, 2011].

The goal of effort estimation is the management of software projects and achieving a comprehensive view of the costs of producing software. It is clear that the software projects effort estimation is a basic and key part of the software engineering. So, the software engineering uses the effort estimation and tries to give method for software projects to the project manager. The software project manager must define the success factors making the programming and controlling processes of the project be developed to avoid project be defeated and must utilize the needed limitations for developing the software projects. An important task in software project management is to understand and control critical variables that influence software effort.

In software engineering, effort is used to define the total time that takes members of a development team to perform a project. Software effort estimation is one of the most important processes in software projects development. SEE must be done before coding the software projects. The purpose of SEE to determine the scope of the project, estimate the amount of work required and the program is scheduled to run software projects.

The effort estimation of the software is input information in order to organize software development teams and allocate the project resources. So, the project manager executes techniques to meet the needs taking into

consideration the estimation effort. One of the main factors of software projects management is the accurate information about the time, effort and costs needed for the project execution. Also using the project records is very effective in success of the software projects and the estimation could be done more reliably.

A software engineering (SE) data repository is defined as a set of well-defined, useful, and pertinent real-world data related to software projects, called datasets, which include quantitative and descriptive information about resources, products, processes, techniques, management, etc. Such data are being collected for various purposes by recognized organizations, as well as by individual software organizations and researchers. In most scientific and engineering disciplines, these data are useful for conducting benchmarking, experimental, and empirical studies. While highly varied and widely available in mature disciplines, data repositories are much less frequently found in emerging disciplines, including software engineering, as illustrated by the Guide to the Software Engineering Body of Knowledge [SWEBOK, 2004].

In this paper, we have done empirical study and comparison of some of the models software effort estimation. The models, which we are dealing with, are developed using statistical and neural networks methods in order to verify which model performs the best. Linear Regression and Radial Basis Function Neural Network have been used in this work. These methods have seen an explosion of interest over years and hence it is important to analyze their performance. They have been analyzed using NASA93 dataset of PROMISE repository that collected information about 93 projects.

This paper will be organized as follows: Previous research works section will describe software effort estimation methods and radial basis function neural networks. Next section will explain the proposed model. Later, the results of the proposed will be described and finally, the conclusion and the future works will be presented.

Previous research works

1. Software Effort Estimation Methods

There are numerous Software Effort Estimation Methods such as Algorithmic effort estimation, machine learning, empirical techniques, regression techniques and theory based techniques. For Software estimation methods, there are several models developed, which can be grouped in two major categories:

- Parametric models, which are derived from the statistical or numerical analysis of historical project data;
- Non-parametric models, which are based on a set of artificial intelligence techniques as neural networks, regression trees, genetic algorithms and rule base induction.

1.1. Parametric Models

Different algorithm models for work estimation, scheduling and costs of the software projects are suggested. Boehm defined one of the most known models for estimation of costs in 1981. COCOMO I was presented in 1981 and, COCOMO II was presented in 2000. It is used for getting estimation of the time and costs activities. The successful management of the software projects depends on the accurate estimation of the projects. Project manager must predict the probable problems and give comprehensive solution for them. Also the project manager must estimate the time and the resources needed for the activities in a way that the work force used in an optimized manner.

Different models of Effort Estimation are presented of which we have taken into consideration the following models.

One of the most identified algorithmic models for SEE is the COCOMO model [Boehm, 2000]. The COCOMO model is used for effort estimation of different software projects. The base COCOMO model is identified as equation (1) for SEE:

$$E = a * (\text{Size})^b \tag{1}$$

The main factor in SEE is the effort rate needed for completing the project. In equation (1), the parameters ‘a’ and ‘b’ are the inaccurate estimation of the complexity of the software and ‘Size’ is the number of the lines of the program in KLOC which is the important factor affecting the accuracy and the efficiency of the estimation [Boehm, 1981].

Also, parameter ‘E’ the amount of effort based on units is Man-Months and this value is directly dependent on the size and complexity of the project. Whatever size and complexity of the project, the more would be the effort on the project. In COCOMO model parameters ‘a’ and ‘b’ depends on the size of the project. The different models of COCOMO for effort estimation using different values of ‘a’ and ‘b’ are showed in Table (1).

Table 1. COCOMO basic models for effort estimation

| |
|---|
| $E = 2.4 * (\text{KLOC})^{1.05}$ Organic |
| $E = 3 * (\text{KLOC})^{1.12}$ Semidetached |
| $E = 3.6 * (\text{KLOC})^{1.20}$ Embedded |

COCOMO basic model is a project estimation model that identifies the effort and software projects management using the models of Table (1). Using the COCOMO model it is possible to identify the effort estimation and identify the needed activities for reaching the goals of the project. The main goal of COCOMO model is that all elements of the project get the same view of the goals, stages, organization and the technical and management procedures of the project and the effort of these elements are in direction of the software projects goals.

Some of various models of SEE are presented in Table (2). These models, which are used by the software teams, are the tools for contributing the effort estimation and controlling the software projects. The main goal of the various models of SEE is to be sure of the final results and the costs of the project. The models of Table (2) compare the software projects from financial, technical and human points to the various models of effort estimation and make the techniques and tools be used by the project manager in execution of the project.

Model Name Model Equation

Table 2. Various models of effort estimation

| |
|--|
| $E = 5.2 * (\text{KLOC})^{1.50}$ Halstead [Halstead, 1977] |
| $E = 5.5 + 0.73 * (\text{KLOC})^{1.16}$ Bailey-Basili [Bailey, 1981] |
| $E = 5.288 * (\text{KLOC})^{1.047}$ Doty [Laird, 2006] |

1.2. Non parametric models

To extract information from the Software Repositories different techniques are used. Mohanty et al. classify intelligent techniques in the following [Mohanty, 2010]:

1. Different neural network (NN) architecture including multilayer perceptron (MLP) and cascade correlation NN;
2. Fuzzy logic;
3. Genetic algorithm (GA);
4. Decision tree;
5. Case-based reasoning (CBR);
6. Soft computing (hybrid intelligent systems).

The other techniques:

1. Analogy based;
2. Support vector machine;
3. Self organizing maps (SOM).

Specifically, this work will focus on studying the application of neural networks in existing repositories.

Neural networks are used broadly in the studies we have selected.

There are several neural network methods that have been used in software estimation. The most common neural networks are the following:

- MultiLayer Perceptron;
- Radial basis function (RBF) network;
- Neuron fuzzy networks.

2. Radial Basis Function Networks

In late 80's Radial basis function emerged as a new artificial neural network. Radial basis neural networks (RBF) are a powerful alternative to approximate and classify a pattern set some times better than multilayer perceptron (MLP) neural networks [Minku, 2013].

RBFs differ from MLPs in that the overall input-output map is constructed from local contributions of Gaussian axons, require fewer training samples and train faster than MLP. The most widely used method to estimate centers and widths consist on using an unsupervised technique called the k-nearest neighbor rule (see figure 1). The centers of the clusters give the centers of the RBFs and the distance between the clusters provides the width of the Gaussians. Computation of the centers, used in the kernels function of the RBF neural network, is being the main focus to study in order to achieve more efficient algorithms in the learning process of the pattern set. The choice of adequate centers implies a high performance, concerning the learning times, convergence and

generalization. The activation function for RBFs network is given by $\phi_i = \phi\left(\frac{\|X(n) - C_i\|}{d_i}\right)$ for $i = 1, 2, \dots, m$

where $C_i = (c_{i1}, \dots, c_{ip})$ are the center of the function radial, d_i is standard deviation. The Gaussian function

$\phi(r) = e^{\left(\frac{-r^2}{2}\right)}$ is the most useful in these cases [Moody, 1989].

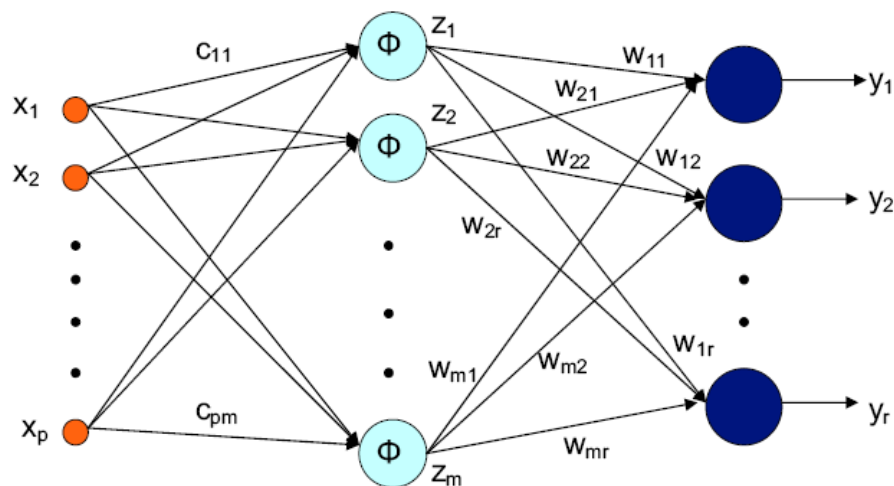


Figure 1. Radial Basis Function Neural Network

Proposed Model

The most important challenge we face in development of the large and complex software projects development is value accurate SEE. First the software projects were small and the costs of producing them included a small percent of the total costs and the error of the effort estimation did not affect the software execution considerably. But by the increase of the number, size and importance of the software projects and the costs of the software development, the software production is the most expensive element in software engineering and the increase of the costs has led the software teams to be defeated in production of the software projects.

The point to be considered in SEE is the method of selecting the suitable model among the estimation models in which the most accurate effort estimation takes place for the development of the software projects.

Also, one of the important goals of studying SEE is studying like costs and utilization of the software are very important. The goal of estimation is to provide the utilization and the control factors of the project and contributes the project manager to define the problem making fields. In the proposed model it is tried to use the RBF and evaluate using NASA project database and find the more accurate value.

In this section we conduct experimental studies and show related results for the experiment. In this experiment, we compare linear regression model and Radial Basis Functional Neural Network.

A. Database

Here we have used COCOMO NASA93 dataset, containing 93 projects. These NASA projects are collected from different NASA centers. These projects were developed during 1980's and 1990's. This database contains size in term of source lines of code (SLOC); the database contains KSLOC value, which means thousand SLOC. It also contains effort in person months and 15 other effort multipliers as described in COCOMO II.

B. Estimation Models

1) Radial Basis Function Neural Network (RBFNN): A radial basis function neural network has been implemented with two input neurons, one hidden layer and eight clusters: KLOC and time to estimate project effort (see figure 2). The net uses a competitive rule with full conscience with one the hidden layer and one output layer with the Tanh function, all the learning process has been performed with the momentum algorithm. Unsupervised learning

stage is based on 100 maximum epochs and the supervised learning control uses as maximum epoch 51938, threshold 0.00001.

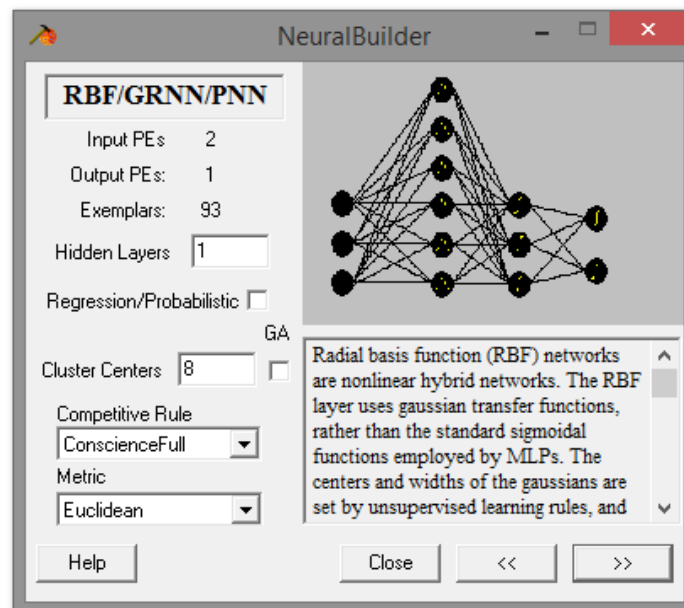


Figure 2. Structure of Radial Basis Function Neural Network Used

2) Regression Analysis Model: This is a traditional prediction method. Here we have used logarithmic function for effort estimation. Linear regression is not suitable in this prediction as effort is not linearly dependent on size (LOC) of the software. So nature log is more appropriate function.

We have performed an initial study using 93 patterns, in training set. Problem under study is prediction of software effort.

Results

The main results of the models studied are presented, first RBFNN and then Nonlinear Regression.

Radial Basis Function Neural Network has approximated in a good manner tested examples, getting a small mean squared error, see Figures 3 and 4 below:

| Active Performance | |
|--------------------|-------------------|
| MSE | 0.003856100042 |
| NMSE | 0.062733542379 |
| r | 0.968670017525 |
| % Error | 65.692399655099 |
| AIC | -386.903203099362 |
| MDL | -369.593719571882 |

Figure 3. Table of mistake obtained

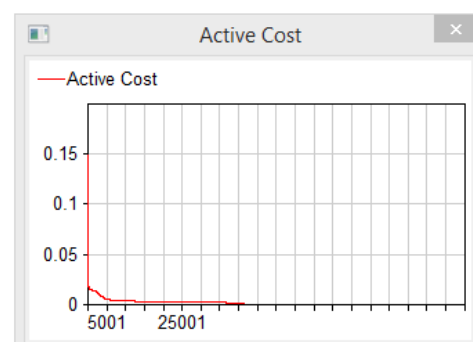


Figure 4. Graph of mistake

After training the network, a study of sensitivity was made. This study shows the influence that each input variable has on the output of the network, see Figure 5.

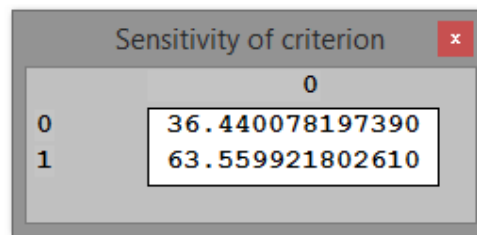


Figure 5. Sensitivity analysis

Now, it will be showed the results of regression model, see Tables 3 and 4. Effort is the dependent variable and Kloc and Time are independent variables. A nonlinear model is used due to the influence of COCOMO exponential models. Function to be estimated is: $a \cdot \text{Kloc}^b \cdot \text{Time}^c$. Marquardt defined estimation method used [Marquardt, 1983]. Estimation stopped due to convergence of residual sum of squares.

Table 3. Estimation Results

| | | | Asymptotic | 95.0% |
|-----------|-----------|----------------|------------|-----------|
| | | | Confidence | Interval |
| Parameter | Estimate | Standard Error | Lower | Upper |
| a | 0.982764 | 0.731052 | -0.469601 | 2.43513 |
| b | -0.473374 | 0.164031 | -0.799251 | -0.147498 |
| c | 2.63606 | 0.386384 | 1.86844 | 3.40368 |

Table 4. Analysis of Variance

| Source | Sum of Squares | Df | Mean Square |
|-----------------------------|----------------|----|-------------|
| Model | 1.02321E8 | 3 | 3.41071E7 |
| Residual | 5.26492E7 | 90 | 584991. |
| Total | 1.5497E8 | 93 | |
| Total (Corr.) | 1.18711E8 | 92 | |
| R-Squared = 55.6491 percent | | | |

The output shows the results of fitting a nonlinear regression model to describe the relationship between Effort and 2 independent variables. The equation of the fitted model is $\text{Effort} = 0.982764 \cdot \text{Kloc}^{-0.473374} \cdot \text{Time}^{2.63606}$.

In performing the fit, the estimation process terminated successfully. The estimated coefficients appeared to converge to the current estimates.

The R-Squared statistic indicates that the model as fitted explains 55.6491% of the variability in Effort. The adjusted R-Squared statistic, which is more suitable for comparing models with different numbers of independent variables, is 54.6636%. The standard error of the estimate shows the standard deviation of the residuals to be 764.847. This value can be used to construct prediction limits for new observations by selecting the Forecasts

option from the text menu. The mean absolute error (MAE) of 397.278 is the average value of the residuals. The Durbin-Watson (DW) statistic tests the residuals to determine if there is any significant correlation based on the order in which they occur in your data file.

Table 5 shows estimated correlations between the coefficients in the fitted model. These correlations can be used to detect the presence of serious multicollinearity, i.e., correlation amongst the predictor variables. In this case, there are 3 correlations with absolute values greater than 0.5.

Table 5. Asymptotic correlation matrix for coefficient estimates

| | a | b | c |
|---|---------|---------|---------|
| a | 1.0000 | 0.7110 | -0.9021 |
| b | 0.7110 | 1.0000 | -0.9430 |
| c | -0.9021 | -0.9430 | 1.0000 |

Using response surface methodology, we explore the relationship between the variables (see Figure 6).

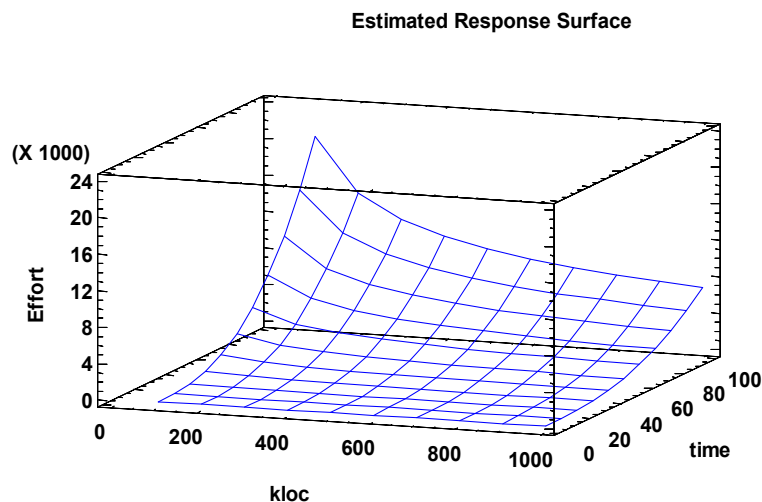


Figure 6. Estimated Response Surface

Conclusion and future work

This paper presents the results about the application of Radial Basis Function Neural Networks on Software Effort Estimation.

Radial basis function neural network learns with only a few patterns are really excellent. The dataset consists of 93 projects. We have obtained less Mean Square Error estimated using RBF than the regression method.

In the future work can further replicate this study using other software repositories in order to contrast the results.

Bibliography

- [Bailey, 1981] Bailey J.W., Basili, V.R. “A meta model for software development resource expenditure,” in Proceedings of the International Conference on Software Engineering, pp. 107–115, 1981
- [Boehm, 1981] Boehm, B., “Software Engineering Economics. Englewood Cliffs”, NJ, Prentice-Hall, 1981
- [Boehm, 2000] Boehm, B. W., Madachy, R., & Steece, B. “Software Cost Estimation with Cocomo II with Cdrom”, Prentice Hall PTR, 2000
- [Halstead, 1977] Halstead, M. H. Elements of Software Science. New York, NJ, Elsevier, 1977
- [Laird, 2006] Laird, L. M., & Brennan, M. C., Software measurement and estimation: a practical approach (Vol. 2). John Wiley & Sons, 2006
- [Malhotra, 2011] Malhotra, R., & Jain, A. Software Effort Prediction using Statistical and Machine Learning Methods. International Journal of Advanced Computer Science and Applications, 2(1), 2011, pp. 1451-1521.
- [Marquardt, 1983] Marquardt, D. W. An algorithm for least-squares estimation of nonlinear parameters. Journal of the Society for Industrial & Applied Mathematics, 11(2), 1963, pp. 431-441.
- [Minku, 2013] Minku, L. L., & Yao, X. Ensembles and locality: Insight on improving software effort estimation. Information and Software Technology, 55(8), 2013, pp. 1512-1528.
- [Mohanty, 2010] Mohanty, R., Ravi, V., & Patra, M. R. The application of intelligent and soft-computing techniques to software engineering problems: a review. International Journal of Information and Decision Sciences, 2(3), 2010, pp. 233-272.
- [Moody, 1989] Moody, J. and Darken C. Fast learning in networks of locally-tuned processing units. Neural Computation, 1, 1989, pp. 281-294
- [Shannon, 1949] Shannon, C.E. The Mathematical theory of communication. In: The Mathematical Theory of Communication. Ed. C.E.Shannon and W.Weaver. University of Illinois Press, Urbana, 1949.
- [SWEBOOK, 2004] IEEE “Guide to the Software Engineering Body Of knowledge- SWEBOOK.” Los Alamitos, California: IEEE Computer Society, 2004, 204 p., <http://www.computer.org/portal/web/swebok> (last accessed on 30/01/2013)

Authors' Information



Ana María Bautista – E.T.S. Ingenieros Informáticos, Universidad Politécnica de Madrid, Madrid; e-mail: am.bautista@alumnos.upm.es

Major Fields of Scientific Research: Artificial Intelligence



Tomas San Felu – E.T.S. Ingenieros Informáticos, Universidad Politécnica de Madrid, Madrid; e-mail: tomas.sanfelu@upm.es

Major Fields of Scientific Research: Software Engineering, Computer Science



Angel Castellanos – Applied Mathematics Department. Universidad Politécnica de Madrid, Madrid; e-mail: angel.castellanos@upm.es

Major Fields of Scientific Research: Artificial Intelligence

MICRORAM: A SIMULATION MODEL OF A COLONY OF BACTERIA EVOLVING INSIDE AN ARTIFICIAL WORLD

Daniel Thai Dam, Rafael Lahoz-Beltra

Abstract: *MICRORAM is a simulation model in which a colony of bacteria evolves inside an artificial world. The model has the flavor of the classical models of the decades of the 80s and 90s in which artificial life was inspired by microbiology. We show how a population of 'bacterial' agents is able to adapt to environmental changes and survive to the attack from an external agent simulated with an 'antibiotic'. The conclusion is that many ideas from the 80s and 90s are still valid, and it is possible to design and simulate agents inspired by natural 'bacterial colonies', with potential applications in bacterial and natural computing.*

Keywords: *artificial life worlds, agent based modeling, bacterial genetic algorithm, conjugation operator.*

ACM Classification Keywords: *I.6 Simulation and Modeling*

Introduction

MICRORAM, an abbreviation for "MICROORGANISMS living in a RAM memory", is a simulation model in which a colony of bacteria evolves inside an artificial world. The model has the flavor of the classical models of the decades of the 80s and 90s in which artificial life was inspired by microbiology [Lahoz-Beltra, 2004; 2008]. We show how a population of 'bacterial' agents is able to adapt to environmental changes and survive to the attack from an external agent simulated with an 'antibiotic'. This was the first model [Thai Dam, 1997] from many others were obtained [Lahoz-Beltra et al., 2014a; Perales-Gravan and Lahoz-Beltra, 2008; Perales-Gravan et al., 2013; Recio Rincon et al., 2014] so that this work has sentimental meaning. According to the prevailing ideas in those decades, the behavior of a *program* can be used to support a complex theory [Partridge and Lopez, 1984], being in some cases the program 'the only irrefutable proposition about such theory'. A good example of this idea is found in a computational model of the chemokinetic behavior in *Paramecium* [Van Houten and Van Houten, 1982]. The development of techniques well known today, as is the case of genetic algorithms and cellular automata, promoted the design of this class of models. On the one hand the ability to simulate the evolution with genetic algorithms leads to models such as the Dewdney's model [Dewdney, 1989] in which the evolution of a population of protozoa in a pond is simulated in a computer. On the other, cellular automata technique allowed in past decades simulate the growth and branching in fungi or conduct the simulation of bacterial colony growth using the BRANCH automata [Ermentrout and Edelstein-Keshet, 1993].

Despite the time elapsed in which were in vogue this class of models, we believe that the design of these models can still provide significant results in biology as well as in bacterial and natural computing [Lahoz-Beltra, 2012; Lahoz-Beltra et al., 2014a]. In the next sections we describe the general features of the MICRORAM model.

Model description

In this section we introduce MICRORAM, that is, the algorithm that results once conjugation as well as other biological features is included in a genetic algorithm. The MICRORAM model consists of three elements (Figure 1):

1. **Agents** - Agents are 'bacteria', capable of performing certain cellular functions, reproduce and evolve over time.
2. **Environment** - It is the artificial world in which the agents inhabit and whose features create a selection pressure. The environment is divided into cells, thus a 2D lattice, it is a finite space but unlimited to be a toroidal shape. We use the word **cell** to refer to these 'slices' of the environment.
3. **Genotype** - The agents have a chromosome being the conjugation the genetic mechanisms responsible for the variability.

The dynamic behavior of the model is due to the interaction of these three elements simultaneously being modified the *genotype* and the *environment* through the *agent*.

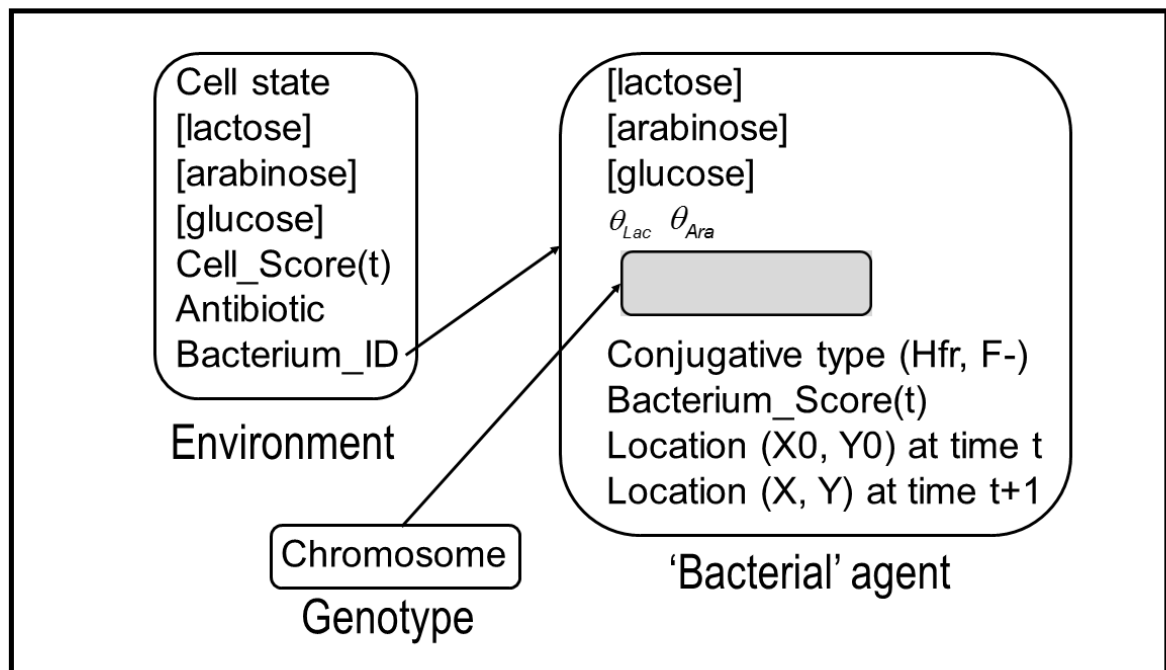


Figure 1. Architecture of a MICRORAM agent, the environment and genotype

Cellular functions of bacterial agents

A 'bacterial' agent performs the following biological tasks:

Nutrition - Bacteria need a source of *energy* (E) for survival, either from three sugars: glucose (E=1), lactose or arabinose (E=2). At the beginning of the simulation MICRORAM randomly assigns specific amounts of sugars in environmental cells and bacteria. The regulation of sugars is through a *model of operon* (cluster of genes under the control of a single promoter). Sugars in the grid cells are not replaced as they are consumed, producing this feature a Darwinian pressure. If the amount of a particular sugar is below a certain threshold then the sugar is not taken up by the bacteria. Furthermore, when glucose in bacteria is exhausted then the conversion of lactose or arabinose in two glucose molecules is triggered. The processes described above can be algorithmically defined as follows. Let $[lactose]_{cell}$, $[arabinose]_{cell}$, $[glucose]_{cell}$ be the amounts of lactose, arabinose and glucose in an environmental cell (Figure 1):

- Compare the value of $[lactose]_{cell}$ with threshold θ_{Lac} :

if $[\text{lactose}]_{\text{cell}} > \theta_{\text{Lac}}$ then activate the 'Lac operon' subtracting a lactose unit to cell register and adding a lactose unit to bacteria register.

if $[\text{lactose}]_{\text{cell}} \leq \theta_{\text{Lac}}$ then disable the 'Lac operon' not updating the lactose records.

- Compare the value of $[\text{arabinose}]_{\text{cell}}$ with threshold θ_{Ara} :

if $[\text{arabinose}]_{\text{cell}} > \theta_{\text{Ara}}$ then activate the 'Ara operon' subtracting an arabinose unit to cell register and adding an arabinose unit to bacteria register.

if $[\text{arabinose}]_{\text{cell}} \leq \theta_{\text{Ara}}$ then disable the 'Ara operon' not updating the arabinose records.

- Compare the value of $[\text{glucose}]_{\text{cell}}$ with threshold θ_{Glu} :

if $[\text{glucose}]_{\text{cell}} > \theta_{\text{Glu}}$ then subtract a glucose unit to cell register and adding a glucose unit to bacteria register. Otherwise, not update the records of glucose.

if $[\text{glucose}]_{\text{cell}} \leq \theta_{\text{Glu}}$ then:

if the bacteria have intracellular lactose then subtract a unit from lactose bacterial record and adding two glucose units in glucose bacterial record. Otherwise, records are not updated.

if the bacteria have intracellular arabinose then subtract a unit from arabinose bacterial record and adding two glucose units in glucose bacterial record. Otherwise, records are not updated.

The adaptability of a bacterium, that is its *fitness*, is given by the amount of glucose accumulated (or equivalently $S(t)$, the bacterial **score**).

Motility - Bacteria move in two different ways. If a bacterium has flagellum (appendage that protrudes the bacterium which role is locomotion) [Lahoz-Beltra, 1997] then it will move attracted to cells with higher sugar content, otherwise it will move erratically. In the case bacterium has flagellum the draw of the cells is done by applying a 'biased roulette' method. Given the initial position (X_0, Y_0) of the bacteria and defining a Moore neighborhood only are selected those cells (X, Y) not occupied by bacteria. Bacterium location is updated synchronously in the population (thus, the bacterial colony). When a bacterium moves occurs an energy expenditure in the form of glucose. In the simulation experiments this value was set equal to $E = 5$.

Cell division - After a certain time, one bacterium is divided into two bacteria. Cell division is simulated as follows: daughter bacterium inherits the parameters of the mother (or alternatively, parental) *bacterium record* (Figure 1). The chromosome and the sugars thresholds are similar in both bacteria, the mother and daughter. However, the bacterial score and intracellular (or bacterial) sugars of the parent bacterium are divided equally between the two bacteria. Mother bacterium will stay in its environmental cell, and the new bacterium will occupy a neighboring cell. If there are no empty cells in the vicinity then bacteria will not be divided.

Bacterial chromosome - The chromosome of a bacterium is a string of 15 bits (Figure 2) such that the first two bits encode the (1-2 positions) Lac operon, the next two bits represent the (3-4 positions) Ara operon, 10 bits are for the antibiotic inhibitor gene (5-14 positions), and the last bit codes for the flagellum gene (15 position). The presence or absence of flagella depends on the value of the last bit encoding this organelle: if the bit value is 1 then the bacterium has flagellum. Otherwise, if the bit value is 0 then the bacterium does not have flagella.

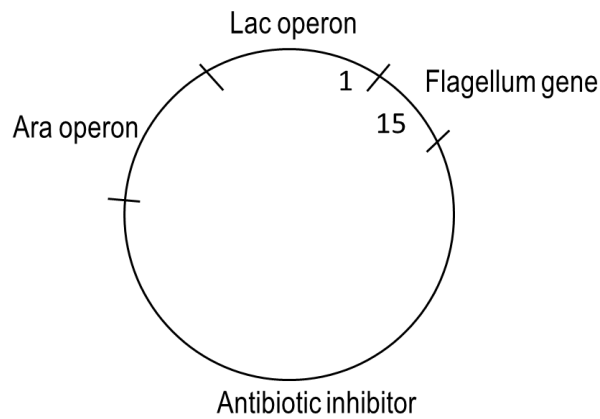


Figure 2. Bacterial chromosome

The operon model (Table I) was simplified to two bits (Figure 3): the first bit (bit controller or regulator) simulates the gene promoter and gene operator, the second bit simulates the structural gene. Its operation is similar to a switch with on/off states, such that a NOT operator is applied to the regulator bit: if the regulator bit is equal to 0 then the structural gene is activated to state 1, capturing sugar (lactose or arabinose) from the environment (environmental cell). Otherwise, no sugar is captured from environment. It is important to note that the operons are regulated by the environment. That is, the regulator bit is modulated by the presence of lactose or arabinose in the environment, activating the operons when the amounts of these sugars are above a threshold value.

Table I- Operon model

| REGULATOR GENE | STRUCTURAL GENE | ACTION |
|----------------|-----------------|----------------------------------|
| 0 | 1 | bacterium captures sugar |
| 1 | 0 | bacterium does not capture sugar |

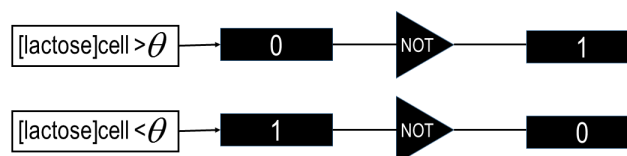


Figure 3. Lac operon model. Depending on lactose concentration in the environment a regulator gene is in 0 or 1 state. A NOT boolean operator is applied to regulator resulting the activation (1) or repression (0) of the structural gene. A similar model is used with arabinose

Antibiotic inhibitor - The antibiotic inhibitor gene is a string of 10 bits length which is coding for a peptide molecule. The peptide binds to an antibiotic molecule according to the key-lock principle. That is, the degree of coupling between the peptide and the antibiotic molecule is simulated according to the Hamming distance, such that 0 simulates a 'valley' and 1 simulates a 'peak' in the molecule conformation. A greater inhibition of antibiotic greater Hamming distance (for example, 1-0 means that a peak of the peptide is coupled to a valley of the antibiotic), surviving the bacterium. Therefore, in the MICRORAM model the degree of inhibition of the antibiotic is simulated through a coefficient k (Table II) that affects to the score of the bacterium: $S(t+1) = k S(t)$. Note that for a Hamming distance less than or equal to 2 the peptide does not inhibit the antibiotic being the $S(t+1)$ equal to 0, thus killing the antibiotic the bacterium.

Table II

| HAMMING DISTANCE | k |
|------------------|------|
| 0, 1, 2 | 0 |
| 3, 4 | 0.25 |
| 5, 6 | 0.5 |
| 7, 8 | 0.75 |
| 9, 10 | 1.0 |

Conjugation - Bacteria exhibit significant phenomena of genetic transfer and crossover between cells. This kind of mechanism belongs to a particular kind of genetic transfer known as horizontal gene transfer. Horizontal, lateral or cross-population gene transfer is any process in which an organism, i.e. a donor bacterium, transfers a genetic segment to another one, a recipient bacterium, which is not its offspring. Conjugation is one of the key genetic mechanisms of horizontal gene transfer between bacteria. In previous papers we introduced a bacterial conjugation operator showing its utility by designing an AM radio receiver [Perales-Gravan and Lahoz-Beltra, 2008] as well as a genetic algorithm including transduction [Perales-Gravan et al., 2013; Lahoz-Beltra et al., 2014b] and which we called as PETRI (*Promoting Evolution Through Reiterated Infection*). Indeed MICRORAM is a model that was the ancestor [Thai Dam, 1997] of this family of models, presented, years later, in this paper.

From a biological point of view we have considered only the simulation of the conjugation of the type Hfr x F-. However, this detail can be omitted without affecting the understanding of the model. If a Boolean variable takes the value True then bacterium is Hfr, being the bacteria F- when the value of the Boolean variable is False. The daughter bacterium inherits the conjugative type from mother bacterium. Conjugation is simulated as follows.

First, a pair of bacteria $\{i, j\}$ are selected according to the next algorithm. The Hfr bacterium searches for the F-bacterium in its neighborhood. In the event that the Hfr bacterium had several candidates randomly select its partner. Second, the transfer of the genome is simulated as follows. We have considered two genetic operators depending on where it is located on chromosome the insertion point of the F plasmid or fertility factor. This factor allows genes to be transferred from one bacterium carrying the factor to another bacterium lacking the factor (<http://en.wikipedia.org/wiki/F-plasmid>). Thus, we introduced two genetic operators, conjugation types I and II:

a) Type I:

1. On chromosome of a (Hfr) donor bacterium, obtain the length l of the strand transferred to the recipient bacterium (F-). The l value has been simulated applying Monte Carlo's method and

assuming DNA lengths exponentially distributed with α parameter [Perales-Gravan and Lahoz-Beltra, 2008];

2. The transferred fragment is inserted into the chromosome of the recipient bacterium, replacing the chromosome segment, between positions 1 and l .

b) Type II:

1. On chromosome of a (Hfr) donor bacterium, obtain a random number U modeling the insertion location of the F plasmid and the length l of the strand transferred to the recipient bacterium (F-);
2. The transferred fragment is inserted into the chromosome of the recipient bacterium, replacing the chromosome segment, between positions $U+1$ and l .

For instance, consider a type II conjugation with $U=2$ and $l=6$ being the bacterial chromosome ‘Lac operon – Ara operon- Antibiotic inhibitor – Flagellum gene’ with gene values 10-11-0000000000-1. In consequence, the strand transferred to the recipient bacterium begins in position 3 and ends in position 8 being the ‘DNA’ fragment 11-0000. In this example, if the recipient chromosome was 00-00-1111111111-0 then after crossover the resulting chromosome will be 00-11-0000111111-0.

Third, the segregation is simulated retaining unchanged the Hfr donor bacterium, updating the chromosome in the recipient bacterium which retains the conjugative type F-. Conjugation operators I and II are similar to COFP (Conjugation Operator with a Fixed Point) and CORP (Conjugation Operator with a Random Point) described in [Perales-Gravan and Lahoz-Beltra, 2008]. The main difference is that the conjugation of type I and II are *bioinspired* operators designed with a *theoretical purpose*, whereas COFP and CORP have a *practical purpose* in the context of genetic algorithms [Davies, 1991; Goldberg, 1989].

Simulation experiments

Before studying the evolution of a bacterial colony or population we compared classical crossover operators [Davies, 1991; Goldberg, 1989] of a genetic algorithm with the bacterial recombination (thus, conjugation types I and II) operators proposed in this paper. Homologous one-point and two-points crossover operators were compared with the conjugation operator, choosing in all experiments a population size $N = 100$ bacteria, chromosomes with length $l = 17$ bits and $\alpha = 0.24$, being the mutation rate equal to 0.08. In the simulation experiments we study the optimization of the following objective function:

$$f(x)=x_{10} \tag{1}$$

where x_{10} is the value in base 10 of the chromosome or binary string x . The study was conducted for a total of 200 generations, conducting the simulation of two kinds of experiments. In a first set of experiments the *environment remains stable*, whereas in another batch of experiments we *changed the environment* after a certain number of generations. Given a certain number of generations, the change of environment was simulated replacing the objective function $f(x)$ by $f'(x)$:

$$f'(x)=131071-f(x) \tag{2}$$

The change of environment was simulated in the generation 5 or 25; the recombination rate was 25%, 50%, 75% and 100%. The simulation experiments performance was evaluated according to the statistical methods described in [Lahoz-Beltra and Perales-Gravan, 2010; 2014]. In MICRORAM model, the evaluation of genetic operators was conducted studying 10 replicate experiments (n), obtaining the average fitness (\bar{f}):

$$\bar{f} = \frac{\sum_{i=1}^N f(x_i)}{N} \tag{3}$$

and the standard error of the mean (SEM):

$$SEM = \frac{\sqrt{\frac{\sum_{i=1}^N (x_i - \bar{f})^2}{N}}}{\sqrt{n}} \quad (4)$$

Concluded this preliminary study, we will describe the experiments carried out with MICRORAM simulator:

Experiment 1. First, we performed a set of experiments in which the bacteria were using homologous recombination instead of conjugation. The experiments were performed with antibiotic present in the environment, studying the antibiotic 'molecule' 1100110011. Experiments without the antibiotic were also conducted. All simulation experiments were performed at different recombination rates, population size $N = 100$, length of chromosome $l = 15$ and mutation rate 0.08.

Experiment 2. Using the same experimental conditions described above, we conducted a batch of experiments in which the bacteria used the conjugation operators, types I and II.

It is important to note that in bacteria with homologous recombination, recombinant bacterium replaces the parental bacteria; while in the bacteria with conjugation, the colony retained both, thus the recombinant bacterium and parental bacterium (donor).

Results

The results of the simulations fit reasonably with observable results in a colony of bacteria, justifying the assumptions of the model and the parameters values. Figure 4 illustrates a sequence of states representative of a population of artificial bacteria simulated with MICRORAM. In Figure 5 we show the results obtained with classic crossover operators in a standard genetic algorithm, showing a convergence of the bacterial colony towards an average fitness value equal to 1200. The SEM value ranged from a 3 minimum to 10 maximum values. In the experiments carried out with conjugation type I no differences were obtained in the minimum SEM value, decreasing the maximum SEM value and therefore the maximum chromosomal variability. However, when the parental bacterium (donor) is removed from the colony then increases the minimum and maximum SEM values, and therefore increases the chromosome population variability (Figure 6a).

In experiments in which a change of environment was simulated the following results were obtained. What was observed when the change of environment occurs in the fifth generation? In conjugation type I experiments the fitness value decreased to [600, 800], except for the case with a recombination rate of 25% in which the population converged to a fitness value equal to 1200. Moreover, in this case the population converges in a stepwise fashion (Figure 6b), also decreasing in the same fashion the SEM value. The results obtained with type II conjugation resemble a standard genetic algorithm in which there is simulated a change of environment (Figure 6c). But what happens when the change of environment arises in generation 25? In this case and for the conjugation of type I with recombination rate of 25% the results are similar to those where the change of environment occurs in the fifth generation. However we highlight the case (Figure 6d) in which the bacterial colony after a decreases in chromosome variability between the 60 and 100 generations, experiences a sharp increases of variability which stabilizes from generation 120.

Finally, in the experiments performed in the presence of antibiotic the following results were obtained. In all experimental situations the antibiotic effect was to decrease the fitness of the population or bacterial colony. When bacteria use the classic one-point crossover operator or alternatively conjugation type I the optimal recombination rate was 75%, while the optimal recombination rate was 50% for the classic two-points crossover operator or conjugation type II.

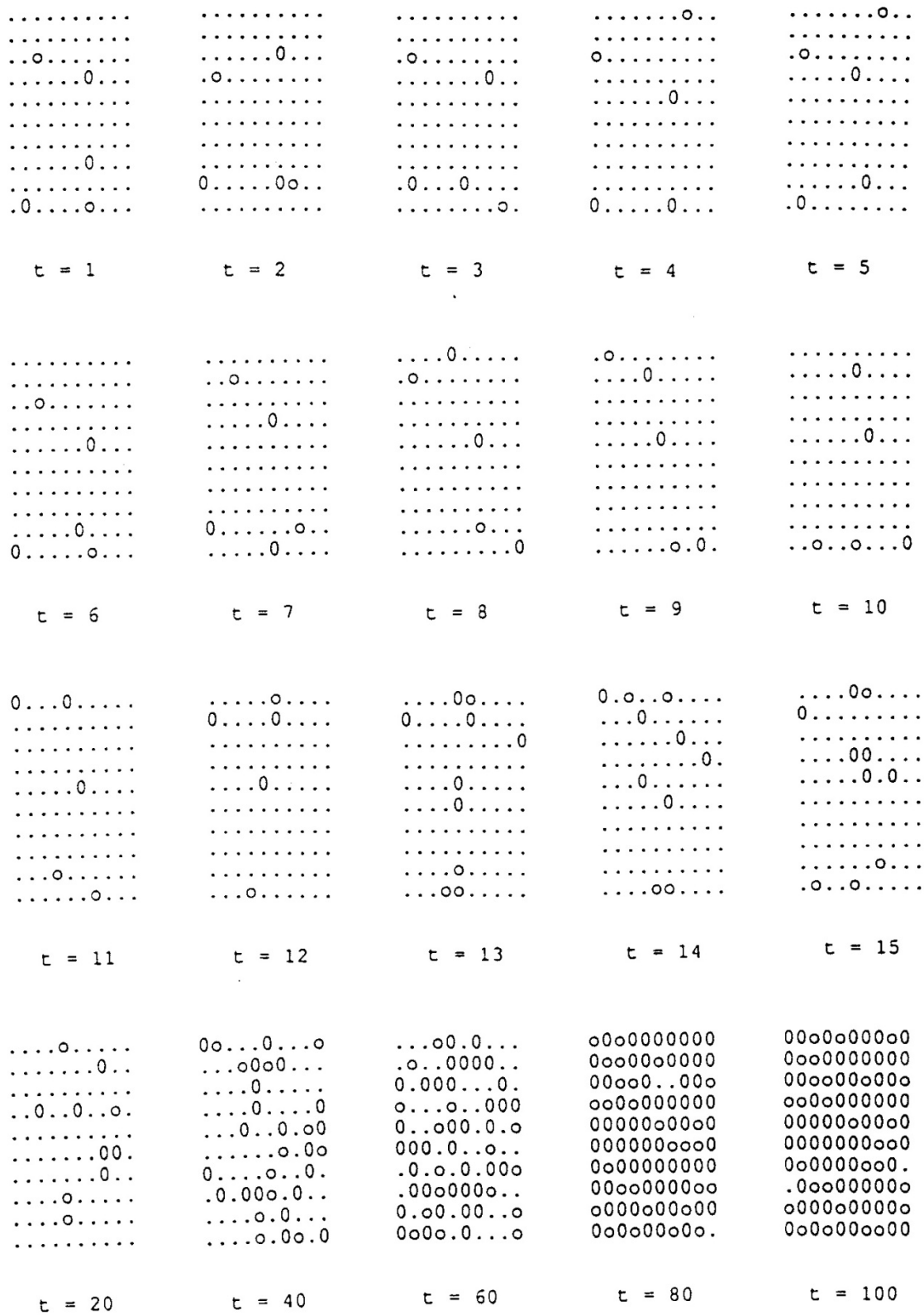


Figure 4. A colony of 'bacterial' agents evolving inside an artificial world (0=Hfr, o=F-)

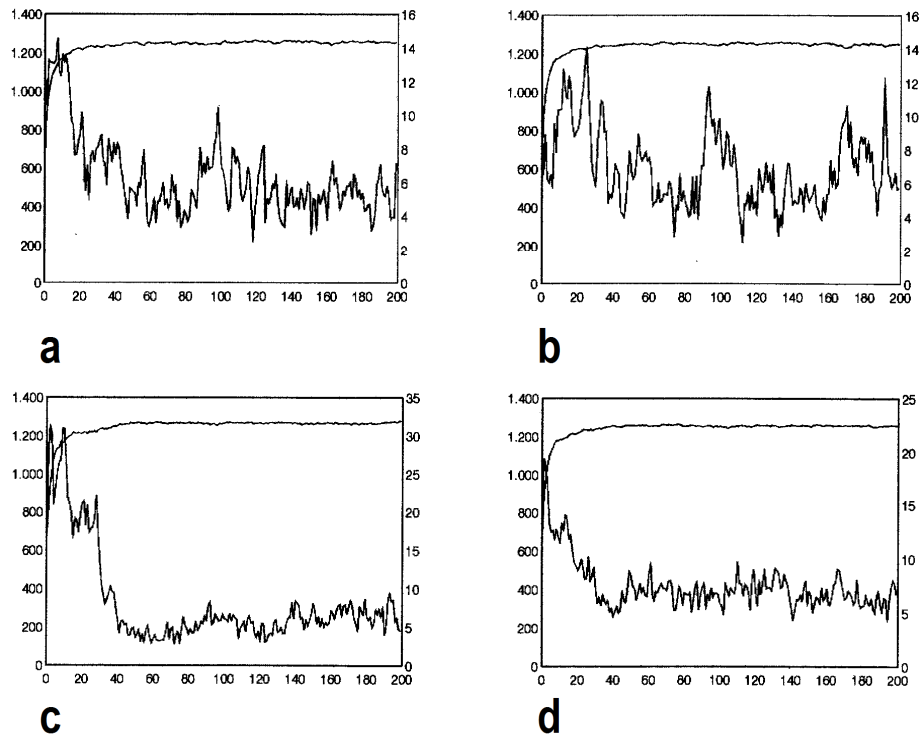


Figure 5. Performance plots showing the average fitness value (continuous line) and SEM (sawtooth curve) vs. generation time. (a) classic one-point crossover (b) classic two-points crossover (c) conjugation type I (d) conjugation type II.

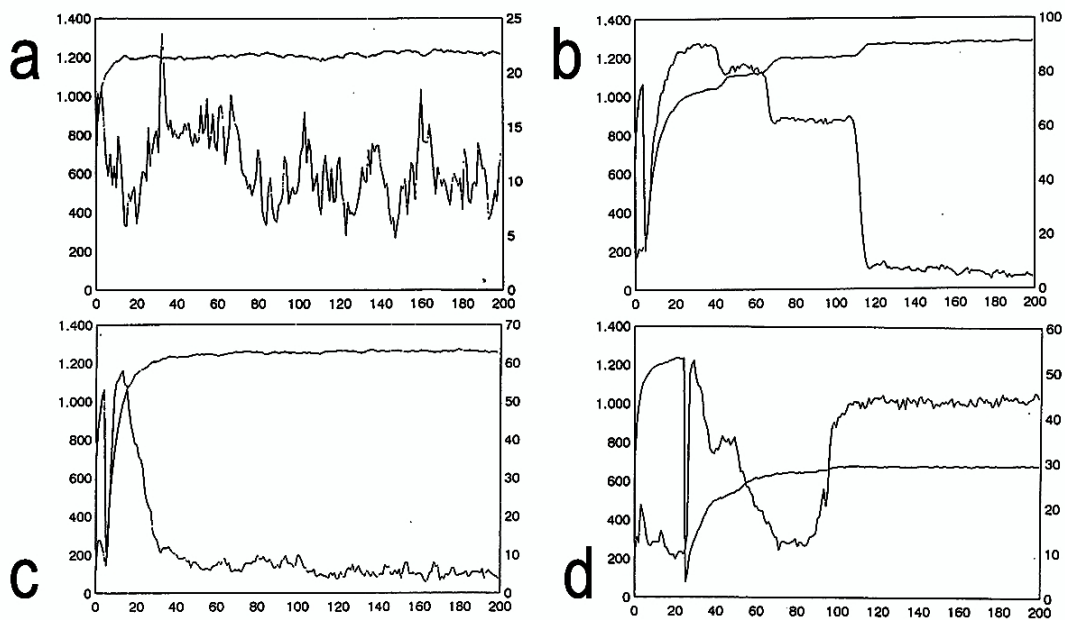


Figure 6. Performance plots showing the average fitness value (continuous line) and SEM (sawtooth curve) vs. generation time. Conjugation type I (a) when bacterium donor is removed and with a (b) change of environment in the 5th generation. (c) Conjugation type II with change of environment in the 5th generation (d) Conjugation type I with a change of environment in generation 25.

Conclusion

We modeled and simulated a population or bacterial colony based on an agent algorithm, exploring the role of conjugation in a genetic algorithm. The simulator has been named MICRORAM, showing how a population of 'bacterial' agents is able to adapt to environmental changes and survive to the attack from an external agent simulated with an 'antibiotic'. This was the first model [Thai Dam, 1997] from many others developed recently, so that this work has a 'sentimental meaning'. The conclusion is that many ideas from the 80s and 90s are still valid, and it is possible to design and simulate agents inspired by natural 'bacterial colonies', with potential applications in bacterial [Lahoz-Beltra, 2012; Lahoz-Beltra et al., 2014a] and natural computing [Recio Rincon et al., 2014].

Bibliography

- [Davies, 1991] L. Davies. Handbook of Genetic Algorithms. Van Nostrand Reinhold.
- [Dewdney, 1989] A. K. Dewdney. 1989. Evolución simulada: Un programa en que los microbios aprenden a cazar bacterias. *Investigación y Ciencia* 154: 96-100 (Spanish edition of Scientific American).
- [Ermentrout and Edelstein-Keshet, 1993] G.B. Ermentrout, L. Edelstein-Keshet. 1993. Cellular automata approaches to biological modelling. *Journal Theoretical Biology* 160: 97-133.
- [Goldberg, 1989] D. E. Goldberg. Genetic algorithms in search, optimization and machine learning. Addison-Wesley.
- [Lahoz-Beltra and Perales-Gravan, 2010] R. Lahoz-Beltra, C. Perales-Gravan. 2010. A survey of nonparametric tests for the statistical analysis of evolutionary computational experiments. *International Journal Information Theories and Applications* 17: 49-61.
- [Lahoz-Beltra and Perales-Gravan, 2014] R. Lahoz-Beltra, C. Perales-Gravan. 2014. Appendix. A survey of nonparametric tests for the statistical analysis of evolutionary computational experiments. figshare. <http://dx.doi.org/10.6084/m9.figshare.1125796>
- [Lahoz-Beltra et al., 2014a] R. Lahoz-Beltra, J. Navarro, P.C. Marijuan. 2014. Bacterial computing: a form of natural computing and its applications. *Frontiers in Microbiology* Article 101. <http://journal.frontiersin.org/Journal/10.3389/fmicb.2014.00101/full>
- [Lahoz-Beltra et al., 2014b] R. Lahoz-Beltra, C. Perales-Gravan, J. de Vicente Buendia, J. Castellanos. 2014. Appendix. Modeling, simulation and application of bacterial transduction in genetic algorithms. figshare. <http://dx.doi.org/10.6084/m9.figshare.1125797>
- [Lahoz-Beltra, 1997] R. Lahoz-Beltra. Molecular automata assembly: principles and simulation of bacterial membrane construction. *BioSystems* 44(3): 209-229.
- [Lahoz-Beltra, 2004] R. Lahoz-Beltra. 2004. Bioinformatica: Simulación, Vida Artificial e Inteligencia Artificial (Transl.: Spanish). Ediciones Díaz de Santos, Madrid, Spain.
- [Lahoz-Beltra, 2008] R. Lahoz-Beltra. 2008. ¿Juega Darwin a los Dados? (Transl.: Spanish). Editorial NIVOLA, Madrid, Spain.
- [Lahoz-Beltra, 2012] R. Lahoz-Beltra. 2012. Cellular computing: towards an artificial cell. *International Journal Information Theories and Applications* 19: 313-318.
- [Partridge and Lopez, 1984] D. Partridge, P.D. Lopez. 1984. Computer programs as theories in biology. *Journal of Theoretical Biology* 108: 539-564.
- [Perales-Gravan and Lahoz-Beltra, 2008] C. Perales-Gravan, R. Lahoz-Beltra. 2008. An AM radio receiver designed with a genetic algorithm based on a bacterial conjugation genetic operator. *IEEE Transactions on Evolutionary Computation* 12(2): 129-142

[Perales-Gravan et al., 2013] C. Perales-Gravan, J. de Vicente Buendia, J. Castellanos, R. Lahoz-Beltra. 2013. Modeling, simulation and application of bacterial transduction in genetic algorithms. International Journal Information Technologies & Knowledge 7:11-22.

[Recio Rincon et al., 2014] C. Recio Rincon, P. Cordero, J. Castellanos, R. Lahoz-Beltra. 2014. A new method for the binary encoding and hardware implementation of metabolic pathways. International Journal Information Theories and Applications 21: 21-30.

[Thai Dam, 1997] D. Thai Dam. 1997. MICRORAM: Un modelo orientado a la simulación de la evolución de una población de bacterias artificiales. Tesina, Facultad de Biología, Universidad Complutense de Madrid. (Transl.: Spanish).

[Van Houten and Van Houten, 1982] J. Van Houten, J.C. Van Houten.1982. Computer simulation of Paramecium chemokinesis behavior. Journal Theoretical Biology 98: 453-468.

Authors' Information



Daniel Thai Dam – *Daniel Thai is a BS in Pharmacy and is currently working as Senior Site Support Engineer at Quintiles, e-mail: dThrak@gmail.com*

Interests: IT processes optimization & synchronization in messy environments



Rafael Lahoz-Beltra – *Department of Applied Mathematics (Biomathematics), Faculty of Biological Sciences, Complutense University of Madrid, 28040 Madrid, Spain, e-mail: lahozraf@ucm.es*

Major Fields of Scientific Research: evolutionary computation, embryo development modeling and the design of bioinspired algorithms

UNIVERSAL AND DETERMINED CONSTRUCTORS OF MULTISSETS OF OBJECTS

Dmytro Terletskyi

Abstract: *This paper contains analysis of creation of sets and multisets as an approach for modeling of some aspects of human thinking. The creation of sets is considered within constructive object-oriented version of set theory (COOST), from different sides, in particular classical set theory, object-oriented programming (OOP) and development of intelligent information systems (IIS). The main feature of COOST in contrast to other versions of set theory is an opportunity to describe essences of objects more precisely, using their properties and methods, which can be applied to them. That is why this version of set theory is object-oriented and close to OOP. Within COOST, the author proposes universal constructor of multisets of objects that gives us a possibility to create arbitrary multisets of objects. In addition, a few determined constructors of multisets of objects, which allow creating multisets, using strictly defined schemas, also are proposed in the paper. Such constructors are very useful in cases of very big cardinalities of multisets, because they give us an opportunity to calculate a multiplicity of each object and cardinality of multiset before its creation. The proposed constructors of multisets of objects allow us to model in a sense corresponding processes of human thought, that in turn give us an opportunity to develop IIS, using these tools.*

Keywords: *constructive object-oriented set theory, class of objects, homogeneous class of objects, inhomogeneous class of objects, set of objects, multiset of objects.*

ACM Classification Keywords: *I.2.0 General – Cognitive simulation, F.4.1 Mathematical Logic – Set theory, D.1.5 Object-oriented Programming, D.3.3 Language Constructs and Features – Abstract data types, Classes and objects, Data types and structures, E.2 Data Storage Representations – Object representation.*

Introduction

Nowadays there are different versions of set theory, such as naive set theory of Cantor [Cantor, 1915], type theory of Russell [Wang, Mc Naughton, 1953], Zermelo-Fraenkel set theory [Fraenkel, Bar-Hillel, 1958; Wang, Mc Naughton, 1953], Von Neumann-Bernays-Gedel set theory [Wang, Mc Naughton, 1953], systems of Quine's set theory [Wang, Mc Naughton, 1953], constructible sets of Mostowski [Mostowski, 1969], alternative set theory of Vopenka [Vopenka, 1979], etc. where definition of set is introduced in different ways. Nevertheless, these definitions just describe the concept of set, and do not explain the origin of particular sets. It means they just declare a fact of existence of sets. That is why questions about the origin of specific sets are arising. Of course, we can conclude that the “new” set can be obtained by set-theoretic operations over “existing” sets, and it is really so. However, the questions about origin of these so-called “existing” sets do not disappear, because if they exist, it means that someone, using some methods (algorithms), created them earlier.

Apart from this, concept of set has important place in human thinking activity during perception, analysis, comparison, retrieval, classification and so on. Really, let us consider situation, when you have bunch of keys and need to open certain lock. If you know how exactly corresponding key looks, you can imagine and distinguish it from other keys from this bunch. In this case, it will be easy and fast. However, into another case you need to

check the keys. It means, you perform certain exhaustive search, and at the same time, you create set of keys, which you have checked. Let us imagine another situation, when you need to count money, which you have in your wallet. During counting, you create at least two sets, set of banknotes and set of coins. In addition, we can consider situation when you want to play chess or checkers, and before starting, you need to make initial arrangement of figures on the chessboard. During figures placement, you create set of white and set of black figures from set of all figures. During the game, you create set of beaten figures and set of unbeaten figures from the set of all figures. These are just a few simple examples from our daily activity. Usually we pay little attention to how do we think, and what concepts do we use during this activity. However, we operate with sets of objects permanently, sometimes it happening consciously sometimes not, but it is so. These facts give us an opportunity to conclude that set is the one of basic constructions of human thinking.

Today, we have an opportunity to use sets in programming, in particular in OOP. As a proof, there are appropriate tools within some OOP-languages for working with such data structure, in particular set in STL for C++ [Musser, Derge, Saini, 2001], HashSet, SortedSet and ISet in C# [Mukherjee, 2012], HashSet in Java [Eckel, 2006], set and frozenset in Python [Summerfield, 2010]. These tools allow sets creation, executing basic set-theoretic operations, membership checking, adding and removing of elements and checking of equivalence between sets, etc.

As we can see, concept of set is very important for mathematics and has some applications in programming, in particular OOP, as practical implementation of some aspects of mathematical set theory. However, set theory and OOP are developed separately, and opportunity to work with sets within OOP is just additional functionality of OOP. It means programmers do not develop set theory, and mathematicians do not develop implementation of set theory within programming languages, very often, these two communities have different interests. Despite this, our target is development of IIS, based on human mechanisms of information analysis, in particular, on manipulation with sets of objects, using OOP. That is why we will try to combine some ideas of set theory and OOP during design and development of such systems. We will consider some constructive version of set theory described in [Terletsy, 2014], which is close to OOP's paradigm, and show its application for simulation of some aspects of human thinking, in particular creation of sets and multisets of objects.

Objects and Classes

We know that each set consists of elements, which form it. Everything, phenomena of our imagination or of our world can be the elements of the set [Cantor, 1915]. From other hand, one of the main postulates of OOP is that real world is created by objects [Pecinovskiy, 2013]. Combining these two ideas, we will call elements of sets – objects. Let us consider such object as natural number. It is clear that every natural number must be integer and positive. These are characteristic properties of natural numbers. It is obvious, that 2 is really a natural number, but -12 and 3.62, for example, are not natural numbers.

Let us consider another object, for instance triangle. We know that triangle is geometrical figure, which has three sides for which the triangle inequality must be satisfied. According to this, geometrical figure, which has sides 3 cm, 5 cm and 7 cm is really a triangle, but figure with sides 2 cm, 4 cm and 7 cm does not triangle. We can conclude that each object has certain properties, which define it as some essence while analyzing these facts. Furthermore, objects and their properties cannot exist separately, because if we assume the opposite, we will have contradiction. On the one hand, object cannot exist separately from its properties, because without properties we cannot imagine and cannot describe it. On the other hand, object's properties cannot exist

separately from object, because without object we cannot see and cannot perceive them. That is why, we cannot consider them separately, and there are few variants of the definitions order. It means that we cannot introduce definition of object without definition of its properties and vice versa. Therefore, we decided to introduce concept of object's properties firstly.

Globally we can divide properties of objects into two types – *quantitative* and *qualitative*. We will define these two types of object's properties formally, but their semantics has intuitive nature.

Definition 1. Quantitative property of object A is a tuple $p_i(A) = (v(p_i(A)), u(p_i(A)))$, where $i = \overline{1, n}$, $v(p_i(A))$ is an quantitative value of $p_i(A)$, and $u(p_i(A))$ are units of measure of quantitative value of $p_i(A)$

Example 1. Suppose we have an apple, and one of its properties is weight. We can present this property as follows $p_w(A) = (v(p_w(A)), u(p_w(A)))$, and if weight of our apple is 0.2 kg, then property $p_w(A)$ will be the following $p_w(A) = (0.2, kg)$. ♠

Definition 2. Two quantitative properties $p_i(A)$ and $p_j(B)$, where $i = \overline{1, n}$, $j = \overline{1, m}$, are equivalent, i.e. $Eq(p_i(A), p_j(B)) = 1$, if and only if $u(p_i(A)) = u(p_j(B))$.

Definition 3. Qualitative property of object A is a verification function $p_i(A) = vf_i(A)$, $i = \overline{1, n}$, which defines as a mapping $vf_i(A) : p_i(A) \rightarrow [0, 1]$.

Example 2. Let us consider such object as a triangle. One of its properties is triangle inequality, which must be satisfied for its sides. We can present this property as follows $p_{ti}(T) = vf_{ti}(T)$, where $vf_{ti}(T)$ is verification function of property $p_{ti}(T)$. In this case, function $vf_{ti}(T) : p_{ti}(T) \rightarrow \{0, 1\}$, and it is a particular case of verification function – predicate or Boolean-valued function. ♠

We can conclude that, such approach gives an opportunity to combine description of property and its verification in the one function, i.e. verification function is a verification function and a description of property at the same time. Therefore, different algorithms can be verifiers and descriptors of properties simultaneously.

Definition 4. Two qualitative properties $p_i(A)$ and $p_j(B)$, where $i = \overline{1, n}$, $j = \overline{1, m}$, are equivalent, i.e. $Eq(p_i(A), p_j(B)) = 1$, if and only if $(vf_i(A) = vf_j(A)) \wedge (vf_i(B) = vf_j(B))$.

Definition 5. Specification of object A is a vector $P(A) = (p_1(A), \dots, p_n(A))$, where $p_i(A)$, $i = \overline{1, n}$ is quantitative or qualitative property of object A .

Definition 6. Dimension of object A is number of properties of object A , i.e. $D(A) = |P(A)|$.

Now, we can formulate the definition of object.

Definition 7. Object is a pair $A / P(A)$, where A is object's identifier and $P(A)$ – specification of object.

Essentially, object is a carrier of some properties, which define it as some essence.

Definition 8. Two objects A and B are similar, if and only if $P(A) \equiv P(B)$.

In general, we can divide objects on concrete and abstract, and does not matter when or how someone created each particular object. It is material implementation of its abstract image – a *prototype*. This prototype is essentially an abstract specification for creation the future real objects. Besides properties of objects, we should allocate operations (methods) which we can apply to objects, considering the features of their specifications. Really, we can apply some *operations (methods)* to objects for their changing and for operating with them. That is why, it will be useful to define concept of object's operation (method).

Definition 9. Operation (method) of object A is a function $f(A)$, which we can apply to object A considering the features of its specification.

Example 3. For such objects as natural numbers n , m we can define operations "+" and ".". ♠

In OOP, programmers consider specifications and methods of objects without objects, and they call it a type or a class of objects, which consists of fields and methods [Weisfeld, 2008; Pecinovsky, 2013]. Fields of class, essentially, are specification of class. Methods are functions, which we can apply to objects of this class for their changing and for operating with them. For convenience, we will also use word "signature" for methods of class. Let us define concept of object's signature.

Definition 10. Signature of object A is a vector $F(A) = (f_1(A), \dots, f_m(A))$, where $f_i(A)$, $i = \overline{1, m}$ is an operation (method) of object A .

Generally, signature of particular object can consist of different quantity of operations, but in practice, especially in programming, usually we are considering finite signatures of objects.

According to definition of object, every object has some specification, which defines it as some essence. There are some objects, which have similar specifications. It means that we can apply the same methods to them. Let us define similar objects.

Definition 11. Objects A and B are similar objects, if and only if, they have the same dimension and equivalent specifications.

If certain two objects are similar, we can conclude that these objects have the same type or class. Now we can introduce concept of object's class.

Definition 12. Object's class T is a tuple $T = (P(T), F(T))$, where $P(T)$ is abstract specification of some quantity of objects, and $F(T)$ is their signature.

When we talk about class of objects, we mean properties of these objects and methods, which we can apply to them. Class of objects is a generalized form of consideration of objects and operations on them, without these objects.

Example 4. Let us describe type *Int* in programming language C++, using concept of similar objects and object's class. Let us set the next specification for the class $P(Int) = (p_1(Int), p_2(Int))$, where property $p_1(Int)$ means “integer number”, property $p_2(Int)$ means “number not bigger then 2147336147 and not smaller than -2147336148”. It is obvious, that all numbers which have properties $p_1(Int)$ and $p_2(Int)$ are objects of class *Int*. Let define the methods of class *Int* in the following way $F(Int) = (f_1(Int), f_2(Int))$, where $f_1(Int) = "+"$ and $f_2(Int) = "*" . ♠$

As we know, in OOP, every particular object has the same fields and behavior as its class, i.e. it has the same specification and signature. It means that every class of OOP is homogeneous in a sense. That is why, let us define concept of homogeneous class of objects.

Definition 13. Homogeneous class of objects T is a class of objects, which contains only similar objects.

The simplest examples of homogeneous classes of objects are class of natural numbers, class of letters of English alphabet, class of colors of the rainbow, etc.

Clearly, that every object is a member of at least one class of objects. Furthermore, some objects are members of few classes simultaneously. For example, such objects as natural numbers n_1, \dots, n_m are members of such classes as natural numbers N , integer numbers Z , rational numbers Q and real numbers R . It is obvious that, class R has the biggest cardinality. Furthermore, it consists of groups of objects of different types. It contradicts concept of OO-class, because different objects from one OO-class cannot have different specifications and signatures. According to this, we cannot describe such classes of objects using concept of homogeneous class. That is why we will define concept of inhomogeneous class of objects.

Definition 14. Inhomogeneous class of objects T is a tuple

$$T = (Core(T), pr_1(A_1), \dots, pr_n(A_n)),$$

where $Core(T) = (P(T), F(T))$ is the core of class T , which includes properties and methods similar to specifications $P(A_1), \dots, P(A_n)$ and signatures $F(A_1), \dots, F(A_n)$ respectively and $pr_i(A_i) = (P(A_i), F(A_i))$, $i = \overline{1, n}$ is projection of object A_i , which consists of properties and methods typical only for this object.

The simplest examples of inhomogeneous classes of objects are class of polygons, cars, birds, etc.

Definition 15. Two classes of objects T_1 and T_2 are equivalent, i.e. $Eq(T_1, T_2) = 1$, if and only if $(P(T_1) \equiv P(T_2)) \wedge (F(T_1) \equiv F(T_2))$.

Sets and Multisets of Objects

According to Naive set theory, a set is a gathering together into a whole of definite, distinct objects of our perception or of our thought, which are called elements of the set [Cantor, 1915]. As we can see, this definition just describes concept of set, and does not explain how to gather these objects together. That is why we are going to define union operation on objects, as a method of set creation.

Definition 16. Union \cup of $n \geq 2$ arbitrary objects is a new set of objects S , which is obtained in the following way

$$S = A_1 / T(A_1) \cup \dots \cup A_n / T(A_n) = \{A_1, \dots, A_n\} / T(S),$$

where $\forall A_i, A_j \in S$, $i, j = \overline{1, n}$ and $i \neq j$, $Eq(A_i, A_j) = 0$; $T(A_i)$, $i = \overline{1, n}$ is a class of object A_i and $T(S)$ is a class of new set of objects S and n is its cardinality.

Example 5. Let us consider such geometrical objects as triangle, square and trapeze. It is obvious that these objects belong to different classes of polygons. Let us denote triangle as A , square as B , trapeze as C , and describe their classes as follows

$$T(A) = ((p_1(A), \dots, p_5(A)), (f_1(A), f_2(A))); T(B) = ((p_1(B), \dots, p_4(B)), (f_1(B), f_2(B)));$$

$$T(C) = ((p_1(C), \dots, p_5(C)), (f_1(C), f_2(C))).$$

Properties $p_1(A)$, $p_1(B)$, $p_1(C)$ are quantities of sides of figures, properties $p_2(A)$, $p_2(B)$, $p_2(C)$, are sizes of sides of figures, properties $p_3(A)$, $p_3(B)$, $p_3(C)$ are quantities of angles of figures, properties $p_4(A)$, $p_4(B)$, $p_4(C)$ are sizes of angles of figures, property $p_5(A)$ is triangle inequality and property $p_5(C)$ is parallelism of two sides of figure. Methods $f_1(A)$, $f_1(B)$, $f_1(C)$ are functions of perimeter calculation of figures, and methods $f_2(A)$, $f_2(B)$, $f_2(C)$ are functions of area calculation of figures.

Of course, specifications and signatures of these objects can include more properties and methods, than we have presented in this example, but everything depends on level of detail. Let us define specifications and signatures of these objects (see Table 1).

Table 1. Specifications and signatures of triangle A , square B and trapeze C

| | p_1 | p_2 | p_3 | p_4 | p_5 | f_1 | f_2 |
|-----|-------|------------------------------------|-------|-------------------------|----------|------------------------|-------------------------------|
| A | 3 | 3.6 cm, 3.6 cm, 5.9 cm | 3 | 35°, 35°, 110° | 1 | $P = \sum_{i=1}^3 a_i$ | $S = \sqrt{p(p-a)(p-b)(p-c)}$ |
| B | 4 | 2 cm, 2 cm, 2 cm, 2 cm | 4 | 90°, 90°, 90°, 90° | \times | $P = \sum_{i=1}^4 a_i$ | $S = a^2$ |
| C | 4 | 3.6 cm, 5.9 cm, 3.6 cm, 11.8 cm | 4 | 35°, 145°, 145°, 35° | 1 | $P = \sum_{i=1}^4 a_i$ | $S = \frac{(a+b)h}{2}$ |

Analyzing Table 1, we can see that property p_5 specified as just value of verification function for particular object. All these functions can be simply implemented using OOP language. Furthermore, there are variety of their implementations that is why we will not consider them within this example.

Now, let us apply the union operation to these objects and create a new set of objects.

$$S = A / T(A) \cup B / T(B) \cup C / T(C) = \{A, B, C\} / T(S)$$

We have obtained a new set of objects S and a new class of objects

$$T(S) = (Core(S), pr_1(A), pr_2(B), pr_3(C)),$$

Where $Core(S) = (p_1(S), p_2(S), p_3(S), p_4(S), f_1(S))$, property $p_1(S)$ is quantity of sides of figures, property $p_2(S)$ means sizes of sides of figures, property $p_3(S)$ is quantity of angles of figures, property $p_4(S)$ means sizes of angles of figures, method $f_1(S)$ is a function of perimeter calculation of figures,

$$pr_1(A) = (p_5(A), f_2(A)), pr_2(B) = (f_2(B)), pr_3(C) = (p_5(C), f_2(C)).$$

Essentially, the set of objects S is the set of triangles of class $T(A)$, squares of class $T(B)$ and trapezes of class $T(C)$ and class of set of objects $T(S)$ describes these three types of geometrical figures. ♣

Therefore, we can create sets of object, applying union operation to objects and not only. According to classical set theory, we can do it, applying union operation to sets of objects. However, this operation does not consider concept of class of objects that is why we need to redefine it.

Definition 17. Union \cup of $m \geq 2$ arbitrary sets of objects is a new set of objects S , which is obtained in the following way

$$S = S_1 / T(S_1) \cup \dots \cup S_m / T(S_m) = \{A_1, \dots, A_n\} / T(S),$$

where $\forall A_i, A_j \in S, i, j = \overline{1, n}$ and $i \neq j, Eq(A_i, A_j) = 0; T(S_i), i = \overline{1, m}$ is a class of set of objects S_i and $T(S)$ is a class of a new set of objects S and n is its cardinality.

Example 6. Let us consider such objects as triangle A , square B and trapeze C , which belong to classes $T(A)$, $T(B)$ and $T(C)$, described in the Example 5, respectively. Let us create two sets of objects S_1 and S_2 using Definition 16, i.e.

$$S_1 = A / T(A) \cup B / T(B) = \{A, B\} / T(S_1); S_2 = A / T(A) \cup C / T(C) = \{A, C\} / T(S_2).$$

As the result we have obtain new sets of objects S_1, S_2 and new classes of objects $T(S_1), T(S_2)$, that have following structures

$$T(S_1) = (Core(S_1), pr_1(A), pr_2(B)); T(S_2) = (Core(S_2), pr_1(A), pr_2(C)).$$

In the case cores of both classes are the same, it means

$$Core(S_1) = Core(S_2) = (p_1(S), p_2(S), p_3(S), p_4(S), f_1(S)),$$

where property $p_1(S)$ is quantity of sides of figures, property $p_2(S)$ means sizes of sides of figures, property $p_3(S)$ is quantity of angles of figures, property $p_4(S)$ means sizes of angles of figures, method $f_1(S)$ is a

function of perimeter calculation of figures. Concerning projections of these classes, then they have following structures $pr_1(A) = (p_1(A), f_2(A))$, $pr_2(B) = (f_2(B))$, $pr_2(C) = (p_5(C), f_2(C))$.

Now, let us calculate union of S_1 and S_2 .

$$S = S_1 / T(S_1) \cup S_2 / T(S_2) = \{A, B\} / T(S_1) \cup \{A, C\} / T(S_2) = \{A, B, C\} / T(S)$$

As we can see, we have obtained the same result, as in the case of union of objects A , B and C , which we considered in the previous example. ♠

Consequently, we have considered two ways of set creation, however we can also obtain a set of objects, combining these two approaches.

Definition 18. Union \cup of $n \geq 1$ arbitrary objects and $m \geq 1$ arbitrary sets of objects is a new set of objects S , which is obtained in the following way

$$S = A_1 / T(A_1) \cup \dots \cup A_n / T(A_n) \cup S_1 / T(S_1) \cup \dots \cup S_m / T(S_m) = \{A_1, \dots, A_k\} / T(S),$$

where $\forall A_i, A_j \in S$, $i, j = \overline{1, k}$ and $i \neq j$, $Eq(A_i, A_j) = 0$; $T(A_v)$, $v = \overline{1, n}$ is a class of object A_v , $T(S_w)$, $w = \overline{1, m}$ is a class of set of objects S_w and $T(S)$ is a class of new set of objects S and k is its cardinality.

Example7. Let us consider objects A , B , C and sets of objects S_1 , S_2 which were described above, and calculate their union.

$$\begin{aligned} S &= A / T(A) \cup B / T(B) \cup C / T(C) \cup S_1 / T(S_1) \cup S_2 / T(S_2) = \\ &= A / T(A) \cup B / T(B) \cup C / T(C) \cup \{A, B\} / T(S_1) \cup \{A, C\} / T(S_2) = \{A, B, C\} / T(S). \end{aligned}$$

As we can see, we have obtained the same result, as in the previous example. ♠

Let us define a concept of set of objects based on methods of set creation, which were considered above.

Definition 19. The set of objects S is a union, which satisfies one of the following schemes:

$$S1: O_1 / T(O_1) \cup \dots \cup O_n / T(O_n) = S / T(S);$$

$$S2: S_1 / T(S_1) \cup \dots \cup S_m / T(S_m) = S / T(S);$$

$$S3: O_1 / T(O_1) \cup \dots \cup O_n / T(O_n) \cup S_1 / T(S_1) \cup \dots \cup S_m / T(S_m) = S / T(S);$$

where O_1, \dots, O_n are arbitrary objects, S_1, \dots, S_m are arbitrary sets of objects, and $T(S)$ is a class of a new set of objects S .

According to types of objects, which form a set of objects, we can obtain different types of sets of objects, in particular set of objects, which consists of only objects, that belong to the same class of objects.

Let us define concept of homogeneous set of objects, based on concept of homogeneous class of objects.

Definition 20. Set of objects $S = \{A_1, \dots, A_n\}$ is homogeneous, if and only if $\forall A_i, A_j \in S$, $i, j = \overline{1, n}$ and $i \neq j$, $Eq(T(A_i), T(A_j)) = 1$.

As we know, multiset is a generalization of the notion of set in which members are allowed to appear more than once [Syropoulos, 2001]. Formally multiset can be defined as a 2-tuple (A, m) , where A is the set, and m is the function that puts a natural number, which is called the multiplicity of the element, in accordance to each element of the set A i.e. $m : A \rightarrow N$. However, this definition does not explain how to create a multiset of objects that is why we are going to define multiset of objects using concept of set of objects.

Definition 21. The multiset of objects is a set of objects $S = \{A_1, \dots, A_n\}$, such that $\exists A_i, A_j \in S$, where $i, j = \overline{1, n}$ and $i \neq j$, $Eq(A_i, A_j) = 1$.

We can obtain a multiset of objects in the same way as a set of objects.

Example 8. Let us consider objects A, B, C and sets of objects S_1, S_2 from Example 5 and Example 6.

Union of objects.

$$S = A / T(A) \cup A / T(A) \cup B / T(B) \cup B / T(B) \cup C / T(C) = \{A, A, B, B, C\} / T(S)$$

Union of sets of objects.

$$S = S_1 / T(S_1) \cup S_2 / T(S_2) = \{A, B\} / T(S_1) \cup \{A, C\} / T(S_2) = \{A, A, B, C\} / T(S)$$

Union of objects and sets of objects.

$$S = A / T(A) \cup S_2 / T(S_2) = A / T(A) \cup \{A, C\} / T(S_2) = \{A, A, C\} / T(S)$$

Using three different ways of creation, we have obtained three different multisets of objects. ♠

Let us define some auxiliary definitions connected with multisets of objects.

Definition 22. Cardinality of multiset of objects $S = \{A_1, \dots, A_n\}$ is a quantity of objects, which it contains, i.e. $|S| = n$.

Definition 23. Basic set of multiset of objects $S = \{A_1, \dots, A_n\}$ is a set of objects S_b , which is defining as follows

$$S_b = bs(S) = \{A_1, \dots, A_m\},$$

where $m \leq n$, $\forall A_i, A_j \in S_b$, $i, j = \overline{1, m}$, $i \neq j$, $Eq(A_i, A_j) = 0$ and $\forall A_w \in S$, $A_w \in S_b$.

Example 9. If we have set of objects $S = \{A, A, B, B, B, C, D, D\}$, then $bs(S) = S_b = \{A, B, C, D\}$. ♠

Universal Constructor of Multisets of Objects

As we can see, a multiset of objects can be obtained similarly to sets of objects. However, sometimes we need to recognize or identify particular copy of some elements, which have multiplicity $m \geq 2$. That is why, we will consider universal constructor of multisets of objects, which was presented in [Terletskyi, 2014]. After that, we will show its generality, i.e. we can create arbitrary multiset of object, using this constructor. However, firstly we are going to define cloning operation on objects.

Definition 24. Clone of the arbitrary object A is the object $Clone_k(A) = A_{i+k} / P(A_i)$, where $P(A)$ is a specification of object A , i is a number of its copy and k is a clone's number of A_i . If the object A is not a clone, then $i = 0$.

The main idea of universal constructor of multisets of objects is superposition of union and cloning operation of objects.

Example 10. Let us consider triangle A from previous section. Using cloning operation, we can create the clones of A , for example

$$Clone_1(A) = A_1 / (p_1(A), \dots, p_5(A)); Clone_2(A) = A_2 / (p_1(A), \dots, p_5(A));$$

Clearly, that triangle A and its clones A_1, A_2 are similar triangles. After this, we can apply union operation to them, and in such a way to create the multiset of triangles S , i.e.

$$S = A / T(A) \cup A_1 / T(A) \cup A_2 / T(A) = \{A, A_1, A_2\} / T(S).$$

Thus, when we have done it, we have also created a new class of multiset of objects $T(S)$, but in this case, it is equivalent to class $T(A)$, i.e. $P(S) = P(A) = (p_1(A), \dots, p_5(A))$ and $F(S) = F(A) = (f_1(A), f_2(A))$. That is why, S is a homogeneous multiset of objects. ♠

Considering this example, we can conclude that

$$S = A \cup \left(\bigcup_{i=1}^2 Clone_i(A) \right).$$

It means that we can create any multisets of objects, using arbitrary superposition of union and cloning operations of objects. According to this, we can define our universal constructor of multisets of objects (UCM) as follows

$$UCM(A, m) = A \cup \left(\bigcup_{i=1}^m Clone_i(A) \right),$$

where m is a multiplicity of object A .

Example 11. Let us extend this constructor to inhomogeneous objects and consider for it the square B and the trapeze C , which were defined in the previous section. Using cloning operation, we can create the clones of object B and of object C , for example

$$Clone_1(B) = B_1 / (p_1(B), \dots, p_4(B)); Clone_1(C) = C_1 / (p_1(C), \dots, p_5(C));$$

$$Clone_2(C) = C_2 / (p_1(C), \dots, p_5(C)).$$

Clearly, that object B and its clone B_1 are similar squares. We have the same situation in case of trapeze C and its clones C_1, C_2 . After this, we can apply union operation to objects B, C and their clones B_1, C_1, C_2 , and in such a way to create a multiset of squares and trapezes S , i.e.

$$S = B / T(B) \cup B_1 / T(B) \cup C / T(C) \cup C_1 / T(C) \cup C_2 / T(C) = \{B, B_1, C, C_1, C_2\} / T(S).$$

Thus, when we have done it, we have also created a new class $T(S)$ with the following specification

$$T(S) = (Core(S), pr_1(B), pr_2(C)) = ((p_1(S), p_2(S), p_3(S), p_4(S), p_5(S), f_1(S)), (f_2(B)), (p_5(C), f_2(C))).$$

Clearly, that S is the inhomogeneous multiset of objects, because $B \equiv B_1$ and $C \equiv C_1 \equiv C_2$, but B and C are objects of different classes. ♠

Considering this example, we can conclude that

$$UCM((A_1, m_1), \dots, (A_n, m_n)) = \bigcup_{i=1}^n \left(A_i \cup \left(\bigcup_{j=1}^{m_i} Clone_j(A_i) \right) \right),$$

where m_1, \dots, m_n are multiplicities of objects A_1, \dots, A_n respectively.

Theorem 1. Any multiset of objects can be created using UCM.

Proof. Our proof consists of two parts, in which we are going to prove that any multiset of objects can be created using UCM(1) and arbitrary multiset of objects can be reduced to input data of UCM (2).

First condition follows from UCM's definition. Really, superposition of union and cloning operation in UCM guarantees multisets of objects in the result. Type of resultant multiset of objects depends on types of objects, which are parameters for UCM and their multiplicities.

Second condition follows from that fact, that every multiset of object can be presented in accordance with the formal definition of multiset, i.e. if $S = \{A_1, \dots, A_k\}$ is multiset of objects, then it can be presented as follows $S = ((A_1, m_1), \dots, (A_n, m_n))$, where $m_1 + \dots + m_n = k$, what is an input data for UCM. It means that we can create exactly the same multiset of objects, using tuple form of presentation of multiset as an input data for UCM, i.e. $UCM((A_1, m_1), \dots, (A_n, m_n)) = \{A_1, \dots, A_k\} = S$. □

As we can see, this constructor is quite general and gives us an opportunity to create different types of multisets of objects, in particular homogeneous and inhomogeneous. Clearly that this constructor is determined if and only if m_1, \dots, m_n are strictly defined. In addition, we are going to define a few other determined constructors of multisets of objects, which strictly define the multiplicity of each element, using for it their own schema.

CP Constructor

This constructor of multisets of objects based on the idea of Cartesian product of two arbitrary sets, that is why we call it CP constructor. We use the idea of Cartesian product of sets. However, in contrast to classical definition of CP we define pairs of CP as sets of objects.

Example 12. Let us consider situation that we need to construct electric garland, and we have green, yellow, orange, blue, purple and rosy light bulbs for it. Before we will make our garland, we need to create color scheme for it. It means we need to decide which colors and how many light bulbs of every color we want to use. It is convenient for us to denote every type of light bulbs according to first letter of its color. Let us assume that we want to use all colors, which we have, and each of them can be used more than once. Let us randomly divide all colors on two sets, for instance $S_1 = \{G, Y, O\}$, $S_2 = \{B, P, R\}$, and build all possible sets of objects which consist of elements of Cartesian product pairs, i.e.

$$S_1 = \{G, B\}, S_2 = \{G, P\}, S_3 = \{G, R\}, S_4 = \{Y, B\}, S_5 = \{Y, P\},$$

$$S_6 = \{Y, R\}, S_7 = \{O, B\}, S_8 = \{O, P\}, S_9 = \{O, R\}.$$

Let us apply union operation to these sets of objects, i.e.

$$\begin{aligned} S &= \{G, B\} / T(S_1) \cup \{G, P\} / T(S_2) \cup \{G, R\} / T(S_3) \cup \{Y, B\} / T(S_4) \cup \{Y, P\} / T(S_5) \cup \\ &\cup \{Y, R\} / T(S_6) \cup \{O, B\} / T(S_7) \cup \{O, P\} / T(S_8) \cup \{O, R\} / T(S_9) = \\ &= \{G, B, G, P, G, R, Y, B, Y, P, Y, R, O, B, O, P, O, R\} / T(S), \end{aligned}$$

where S is multiset of objects and $T(S)$ is its class. Clearly, that all objects are similar, that is why class $T(S)$ is homogeneous. As the result, we have obtained multiset of objects, which consists of six objects G, Y, O, B, P, R and we can consider S as one of possible projects of future electric garland. ♠

Generally we can represent this scheme as follows $S = \{(G, 3), (Y, 3), (O, 3), (B, 3), (P, 3), (R, 3)\}$, because S is a multiset of colors. Such form of presentation gives us quantity of each type of light bulbs. However, order of colors is very important aspects of garland's creation. It is obvious that different orders of the same quantity of colors and placement of particular light bulbs give us different perception of garland. According to it, we can vary different combinations of light bulbs for finding needed combination.

Sometimes we need to identify each light bulb of each color, for example for substitute. That is why we are going to improve our constructor in this aspect, via indexation operation.

Definition25. Indexation of object A_i is a redefining of its index i , i.e. $Ind(A_i) = A_{i+w} / (p_1(A), \dots, p_n(A))$, where i is an index of object A and w is its increase.

According to this, the result of the Example 12 is the following

$$\begin{aligned} S &= \{G_1, B_1\} / T(S_1) \cup \{G_2, P_1\} / T(S_2) \cup \{G_3, R_1\} / T(S_3) \cup \{Y_1, B_2\} / T(S_4) \cup \{Y_2, P_2\} / T(S_5) \cup \\ &\cup \{Y_3, R_2\} / T(S_6) \cup \{O_1, B_3\} / T(S_7) \cup \{O_2, P_3\} / T(S_8) \cup \{O_3, R_3\} / T(S_9) = \\ &= \{G_1, B_1, G_2, P_1, G_3, R_1, Y_1, B_2, Y_2, P_2, Y_3, R_2, O_1, B_3, O_2, P_3, O_3, R_3\} / T(S), \end{aligned}$$

where S is a multiset of objects and $T(S)$ is its class. According to this, we can represent our CP constructor as follows

$$CP(S_1, S_2) = \bigcup_{i=1}^n \bigcup_{j=1}^m (Ind_j(A_i) \cup Ind_i(B_j)),$$

where S_1, S_2 are basic sets of objects for multiset of objects S , $A_i \in S_1, B_j \in S_2, n = |S_1|$ and $m = |S_2|$.

As we can see, CP constructor gives us determined scheme for creation of multiset of objects. We also can calculate multiplicity of each object and cardinality of multiset before its creation. As a proof of these facts, we can formulate and prove following two theorems.

Theorem 2. Cardinality of each multiset of objects S , which is obtained using CP constructor, can be calculated by the following formula

$$|S| = 2nm,$$

where $n = |S_1|$, $m = |S_2|$.

Proof. As we know, cardinality of Cartesian product of two sets can be calculated as follows

$$|S_1 \times S_2| = |S_1| \cdot |S_2| = nm.$$

According to the fact, that elements of Cartesian product are pairs, we can conclude that $|CP(S_1, S_2)| = 2nm$. □

Theorem 3. Multiplicity of each object A_i from multiset of objects S , which is obtained using CP constructor, can be calculated by the following formula

$$m(A_i) = \begin{cases} |S_2|, & \exists B_j \in S_1 \mid A_i = B_j; \\ |S_1|, & \exists C_k \in S_2 \mid A_i = C_k; \end{cases}$$

where $i = \overline{1, 2nm}$, $j = \overline{1, n}$, $k = \overline{1, m}$ and S_1, S_2 are basic sets of objects for multisets of objects S .

Proof. Proof follows from the definition of Cartesian product of sets. □

RCL Constructor

The basic principle of this constructor is recursive cloning of set of objects that is why we call this constructor RCL constructor. We will combine the idea of object's cloning with the idea of direct recursion, within RCL constructor, but firstly we need to define cloning operation for set of objects.

Definition 26. Clone of the arbitrary set of objects $S = \{A_1, \dots, A_n\} / T(S)$ is the set of objects

$$Clone_i(S) = \{A_{1+i}, \dots, A_{n+i}\} / T(S),$$

where $T(S)$ is a class of set of objects S and i is a number of its clone.

Example 13. Let us consider Example 12 and imagine that we have only green, yellow and red light bulbs. It means, that we have set of colors $S_1 = \{G, Y, R\}$. Let us clone it once, and apply union operation to it and to the result of its cloning, i.e.

$$\begin{aligned} S_2 &= S_1 / T(S_1) \cup Clone_1(S_1 / T(S_1)) = \{G, Y, R\} / T(S_1) \cup \{G_1, Y_1, R_1\} / T(S_1) = \\ &= \{G, Y, R, G_1, Y_1, R_1\} / T(S_1). \end{aligned}$$

As the result, we have obtained the multiset of colors S_2 . Let us repeat the same procedure for it.

$$\begin{aligned} S_3 &= S_2 / T(S_1) \cup Clone_1(S_2 / T(S_1)) = \{G, Y, R, G_1, Y_1, R_1\} / T(S_1) \cup \{G_2, Y_2, R_2, G_3, Y_3, R_3\} / T(S_1) = \\ &= \{G, Y, R, G_1, Y_1, R_1, G_2, Y_2, R_2, G_3, Y_3, R_3\} / T(S_1). \end{aligned}$$

where S_3 is a multiset of objects, and $T(S_1)$ is its class. As the result we have obtained multiset of objects which consists of three objects G, Y, R and their copies, that can be accurately identified and we can consider S as one of possible projects of future electric garland. ♠

Using a scheme of creation of multiset of objects from Example 12, we can represent our RCL constructor as follows

$$RCL^n(S) = \begin{cases} S, & n = 0; \\ S \cup Clone_{2^{n-1}}(S), & n = 1; \\ RCL^{n-1}(S) \cup Clone_{2^{n-1}}(RCL^{n-1}(S)), & n \geq 2. \end{cases}$$

where S is a basic set of objects for multiset of objects $RCL^n(S)$ and n is a recursion depth.

As we can see, RCL constructor gives us defined order of colors. We also can calculate cardinality of garland and quantity of light bulbs of each color before garland's creation. As a proof of these facts, we can formulate and prove following two theorems.

Theorem 4. *Cardinality of each multiset of objects S , which is obtained using RCL constructor, can be calculated by the following formula*

$$|S| = n2^i,$$

where n is a cardinality of basic set of objects and i is recursion depth.

Proof. According to the scheme of RCL constructor, on each step we will make a union of two sets of objects, which have equal cardinality. It means, if $|S_b| = n$, then on the step $i = 1$ we have a multiset of objects which cardinality is calculated as follows $n + n = 2n = n2^1$. On the step $i = 2$ we have a multiset of object with cardinality $2n + 2n = 4n$, i.e. $n2^1 + n2^1 = n2^2$, on the step $i = 3$ we have $4n + 4n = 8n$, i.e. $n2^2 + n2^2 = n2^3$, etc. It means that on the step $i = k$ we will have $n2^{k-1} + n2^{k-1} = n2^k$, that is why we can conclude that cardinality of resultant multiset of objects will be equal $n2^i$, where i is recursion depth (step).□

Theorem 5. *Multiplicity of each object A_j from multiset of objects S , which is obtained using RCL constructor, can be calculated by the following formula*

$$m(A_j) = 2^i,$$

where i is a recursion depth of RCL constructor.

Proof. We know, that on each step i RCL constructor will equally increase the multiplicity of all objects from set of objects S_i , it follows from the scheme of RCL constructor. According to Theorem 4, cardinality of resultant multiset of objects $|S| = n2^i$, where i is recursion depth of RCL constructor. Combining these two facts, we can conclude that $m(A_j) = n2^i / n = 2^i$.□

PS Constructor

First version of this constructor was presented in [Terletskyi, 2014] and now we are going to introduce its extension, which give us new abilities of its application. This constructor of multisets of objects is based on the idea of powerset of some set, which is why we will call it PS constructor.

Example 14. Let us consider again Example 12 and build all possible subsets of colors according to Definition 19 for set of colors $S = \{G, Y, R\}$, i.e.

$$S_1 = \{G, Y\}, S_2 = \{G, R\}, S_3 = \{Y, R\}, S_4 = \{G, Y, R\}.$$

Let us apply union operation to sets of objects S_1, \dots, S_4 , i.e.

$$\begin{aligned} S &= \{G, Y\} / T(S_1) \cup \{G, R\} / T(S_2) \cup \{Y, R\} / T(S_3) \cup \{G, Y, R\} / T(S_4) = \\ &= \{G, Y, G, R, Y, R, G, Y, R\} / T(S), \end{aligned}$$

where S is a multiset of objects, and $T(S)$ is its class. However, such form of PS constructor does not provide indexation of light bulbs of the same color that is why we will improve it in this direction. As we can see, PS constructor consists of two parts, first of them is selection of subsets from basic set of objects, and second one is union of these subsets. Clearly, that we need to select all possible subsets of objects from basic set of objects in such way, that all copies of each object have unique index. That is why, we will organize selection procedure of subsets marking a choice of every object from set of objects S , during selection of every its subset using increase indexation of chosen objects from set of objects S i.e.

$$\begin{aligned} S_1 &= \{G, R\}, S = \{Ind_1(G), Ind_1(Y), R\} = \{G_1, Y_1, R\}; \\ S_2 &= \{G_1, R\}, S = \{Ind_1(G_1), Y_1, Ind_1(R)\} = \{G_2, Y_1, R_1\}; \\ S_3 &= \{Y_1, R_1\}, S = \{G_2, Ind_1(Y_1), Ind_1(R_1)\} = \{G_2, Y_2, R_2\}; \\ S_4 &= \{G_2, Y_2, R_2\}, S = \{Ind_1(G_2), Ind_1(Y_2), Ind_1(R_2)\} = \{G_3, Y_3, R_3\}. \end{aligned}$$

Let us apply union operation to S_1, \dots, S_4 and create new multiset of objects S , i.e.

$$\begin{aligned} S &= \{G, Y\} / T(S_1) \cup \{G_1, R\} / T(S_2) \cup \{Y_1, R_1\} / T(S_3) \cup \{G_2, Y_2, R_2\} / T(S_4) = \\ &= \{G, Y, G_1, R, Y_1, R_1, G_2, Y_2, R_2\} / T(S), \end{aligned}$$

where S is a multiset of objects, and $T(S)$ is its class. As the result we have obtained a multiset of objects, which consists of three objects G, Y, R and their copies, which can be accurately identified and we can consider S as one of possible projects of future electric garland. ♠

Now we can formulate and prove the following proposition.

Proposition 1. *The quantity of all possible subsets of sets of objects S can be calculated by the following formula*

$$q(S_w) = 2^n - n - 1,$$

where $n = |S|$.

Proof. As we know, powerset of any set A is the set of all subsets of A , including the empty set \emptyset and A itself, and it is denoted like

$$P(A) = \{P \mid P \subseteq A\}.$$

i.e. for the set $A = \{a, b, c\}$

$$P(A) = \{\{\emptyset\}, \{a\}, \{b\}, \{c\}, \{a, b\}, \{a, c\}, \{b, c\}, \{a, b, c\}\}.$$

We also know that cardinality of powerset $|P(A)|$ of a set A can be calculated using the following formula

$$|P(A)| = 2^n,$$

where $n = |A|$. However, according to the Definition 19, $\{\emptyset\}$, $\{a\}$, $\{b\}$, $\{c\}$ are not sets and set cannot be an element of another set. That is why in case of sets of objects previous formula can be rewritten as follows

$$q(S_w) = 2^n - n - 1,$$

where $S_w \subseteq S$, $q(S_w)$ is a quantity of all possible S_w . \square

Using Proposition 1 and scheme of creation of multiset of objects from Example 14, we can represent our PS constructor as follows

$$PS(S) = \bigcup_{w=1}^{2^n - n - 1} S_w,$$

where S is a basic set of objects for multiset of objects $PS(S)$ and $S_w \subseteq S$.

As we can see, PS constructor gives us determined scheme for creation of multiset of objects. We also can calculate multiplicity of every object and cardinality of multiset before its creation. As a proof of these facts, we can formulate and prove two following theorems.

Theorem 6. *The cardinality of each multiset of objects S , which is obtained using PS Constructor, can be calculated by the following formula*

$$|S| = \frac{n2^n}{2} - n,$$

where $n = |S_b|$.

Proof. Let us consider the set $S_1 = \{A, B, C\}$ and build a powerset for it, i.e.

$$P(S_1) = \{\{\emptyset\}, \{A\}, \{B\}, \{C\}, \{A, B\}, \{A, C\}, \{B, C\}, \{A, B, C\}\}.$$

Let us create the multiset M_1 as a union of all elements of $P(S_1)$, i.e.

$$M_1 = \{\emptyset\} \cup \{A\} \cup \{B\} \cup \{C\} \cup \{A, B\} \cup \{A, C\} \cup \{B, C\} \cup \{A, B, C\} = \{A, B, C, A, B, A, C, B, C, A, B, C\}.$$

Clearly that $|M_1| = 12$. Let us consider the set $S_2 = \{A, B, C, D\}$ and build a powerset for it, i.e.

$$P(S_2) = \{\{\emptyset\}, \{A\}, \{B\}, \{C\}, \{D\}, \{A, B\}, \{A, C\}, \{A, D\}, \{B, C\}, \{B, D\}, \{C, D\}, \\ \{A, B, C\}, \{A, B, D\}, \{A, C, D\}, \{B, C, D\}, \{A, B, C, D\}\}.$$

Let us create the multiset M_2 as a union of all elements of $P(S_2)$, i.e.

$$M_2 = \{\emptyset\} \cup \{A\} \cup \{B\} \cup \{C\} \cup \{D\} \cup \{A, B\} \cup \{A, C\} \cup \{A, D\} \cup \{B, C\} \cup \{B, D\} \cup \{C, D\} \cup \\ \cup \{A, B, C\} \cup \{A, B, D\} \cup \{A, C, D\} \cup \{B, C, D\} \cup \{A, B, C, D\} = \\ = \{A, B, C, D, A, B, AC, A, D, B, C, B, D, C, D, A, B, C, A, B, D, A, C, D, B, C, D, A, B, C, D\}.$$

As you can see $|M_2| = 32$. We know that $|P(S)| = 2^n$, where $n = |S|$. Clearly that in the case of S_1 , $|P(S_1)| = 2^3 = 8$ and we can put

$$|M_1| = \frac{3 \cdot 2^3}{2} = 12,$$

as it is really so. In the case of S_2 , $|P(S_2)| = 2^4 = 16$ and similar to the previous case we can put

$$|M_2| = \frac{4 \cdot 2^4}{2} = 32,$$

as we can see it is also true. Based on principle of mathematical induction we can conclude that for $P(S_n)$,

$$|M_n| = \frac{n2^n}{2}.$$

Let us consider the set of objects $S_k = \{A, B, C\}$ and build for it all possible subsets of objects considering Definition 19, i.e.

$$S_1 = \{A, B\}, S_2 = \{A, C\}, S_3 = \{B, C\}, S_4 = \{A, B, C\}.$$

Let us create the multiset of objects M_k as a union of all subset of S_k , i.e.

$$\begin{aligned} M_k &= \{A, B\} / T(S_1) \cup \{A, C\} / T(S_2) \cup \{B, C\} / T(S_3) \cup \{A, B, C\} / T(S_4) = \\ &= \{A, B, A, C, B, C, A, B, C\} / T(M_k). \end{aligned}$$

Clearly that in the case of $|M_k|$, the formula which was used for calculation $|M_n|$ will be changed to

$$|M_k| = \frac{k2^k}{2} - k,$$

where $k = |M_k|$. □

Theorem 7. *The multiplicity of each object A_i from the multiset of objects S , which is obtained using PS constructor, can be calculated by the following formula*

$$m(A_i) = 2^{n-1} - 1,$$

where $n = |S_b|$.

Proof. We know that generating of possible subsets of objects S_1, \dots, S_w for set of objects S_b can be represented as a combination of $k = \overline{2, n}$ different elements from the set of n elements, i.e. C_n^k . During creation of subsets of cardinality 2, we need to combine every object A_i with every object from the set of objects $S_b \setminus A_i$. Clearly, we can create only $n - 1$ such subsets, i.e. C_{n-1}^1 . In the case of subsets of cardinality 3, we will have C_{n-1}^2 , and finally, in the case of subsets of cardinality k we will have C_{n-1}^{k-1} .

According to the scheme of PC constructor, we can conclude that multiplicity of every object A_i from multiset of objects S consists of multiplicities of object A_i in every subset of objects, i.e.

$$m(A_i) = \sum_{w=1}^{2^n - n - 1} m_w(A_i),$$

where $m_w(A_i)$ is the multiplicity of object A_i in the subset of objects $S_w \subseteq S_b$. It follows from $S_1 \cup \dots \cup S_w$, where $w = 2^n - n - 1$. Using this fact, we can conclude that

$$m(A_i) = C_{n-1}^1 + \dots + C_{n-1}^{n-1},$$

where $n = |S_b|$. It means that every object $A_i \in S$ has the same multiplicity. Using this fact, we can conclude that

$$m(A_i) = \frac{|S|}{|S_b|} = \frac{\frac{n2^n}{2} - n}{n} = \left(\frac{n2^n}{2} - n \right) \frac{1}{n} = \frac{n2^n}{2n} - \frac{n}{n} = \frac{2^n}{2} - 1 = 2^{n-1} - 1,$$

where $n = |S_b|$. □

Let us consider proof of Theorem 7. It shows that multiplicity of every object A_i from multiset of objects S can be calculated as a sum of appropriate binomial coefficients. Using this fact, we can build a part of Pascal's triangle. However, in contrast to original Pascal's triangle, we will combine its part with results of Theorem 6 and Theorem 7. It is convenient to formulate it as a following corollary.

Corollary 7.1. *We can calculate cardinality, multiplicity of every object from the multiset of objects, which was created using PS constructor, and quantity of subsets of objects which were used for its creation, using the following matrix*

| | | | | | | | | | |
|----------|-------|----------|---------|-----|-----|-----|-----|-----|-----|
| $m(A_k)$ | $ S $ | $q(S_w)$ | $ S_b $ | 2 | 3 | 4 | 5 | 6 | ... |
| 1 | 2 | 1 | 2 | 1 | | | | | |
| 3 | 9 | 4 | 3 | 3 | 1 | | | | |
| 7 | 28 | 11 | 4 | 6 | 4 | 1 | | | , |
| 15 | 75 | 26 | 5 | 10 | 10 | 5 | 1 | | |
| 31 | 186 | 57 | 6 | 15 | 20 | 15 | 6 | 1 | |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |

where column $m(A_k)$ reflects multiplicity of object A_k in multiset of objects S ; column $|S|$ reflects cardinality of multiset of objects S ; column $q(S_w)$ reflects quantity of $S_w \subseteq S_b$ that was used for establishing S ; column $|S_b|$ reflects cardinality of basic sets of objects; first row starting with 5-th column reflects quantity of subsets of objects of certain cardinality, where cardinality coincides with the value of $a_{1,j \geq 5}$.

The elements of column $m(A_k)$ can be calculated using Theorem 7 or using the following formula

$$a_{i \geq 2, 1} = \begin{cases} 1, & i = 2; \\ \sum_{i \geq 2, j \geq 4} a_{i-1, j}, & i \geq 2. \end{cases}$$

The elements of column $|S|$ can be calculated using Theorem 6 or using the following formula

$$a_{i \geq 2, 2} = \sum_{i \geq 2, j \geq 5} a_{i, j} \cdot a_{1, j}.$$

The elements of column $q(S_w)$ can be calculated using Proposition 1 or using the following formula

$$a_{i \geq 2, 3} = \sum_{i \geq 2, j \geq 5} a_{i, j}.$$

The element $a_{i \geq 2, j \geq 5}$ of the matrix can be calculated in such a way

$$a_{i \geq 2, j \geq 5} = \begin{cases} 1, & j - i = 1; \\ a_{i-1, j-1} + a_{i-1, j}, & j - i < 3. \end{cases}$$

or using the following formula

$$a_{i \geq 2, j \geq 5} = \frac{a_{i, 4}!}{a_{1, j}!(a_{i, 4} - a_{1, j})!}.$$

D2 Constructor

Similarly, to PS constructor, the first version of this constructor was also presented in [Terletskyi, 2014] and now we introduce its extension, which give us new abilities of its application. This constructor of multisets of objects is based on decomposition of basic set of objects on two disjoint subsets such, that in the result of their union we will obtain initial (basic) set of objects. That is why we call this constructor as D2 constructor.

Example15. Let us consider Example 12 and imagine that we have light bulbs of green, yellow, red and blue colors, it means that we have set of colors $S = \{G, Y, R, B\}$. Let us perform D2 decomposition of it and find all possible variants of such decomposition, i.e.

$$S_1 = \{G, Y\}, S_2 = \{R, B\}; S_3 = \{G, R\}, S_4 = \{Y, B\}; S_5 = \{G, B\}, S_6 = \{Y, R\}.$$

Let us apply union operation to these sets of objects, i.e.

$$S = \{G, Y\} / T(S_1) \cup \{R, B\} / T(S_2) \cup \{G, R\} / T(S_3) \cup \{Y, B\} / T(S_4) \cup \\ \cup \{G, B\} / T(S_5) \cup \{Y, R\} / T(S_6) = \{G, Y, R, B, G, R, Y, B, G, B, Y, R\} / T(S),$$

where S is a multiset of objects, and $T(S)$ is its class. However, such form of D2 constructor does not provide indexation of light bulbs of same color that is why we will improve it in this direction.

As we can see, as the result of D2 decomposition of set of objects, we have obtained sets of objects S_1, \dots, S_6 . It means that in this case, there are three possible variants of such decomposition. Each variant of decomposition consists of pair of sets of objects. Let us change indexes of objects of these sets into accordance with number of decomposition's variant, using indexation operation, i.e.

$$S_1 = \{Ind_1(G), Ind_1(Y)\} = \{G_1, Y_1\}, S_2 = \{Ind_1(R), Ind_1(B)\} = \{R_1, B_1\}, \\ S_3 = \{Ind_2(G), Ind_2(R)\} = \{G_2, R_2\}, S_4 = \{Ind_2(Y), Ind_2(B)\} = \{Y_2, B_2\}, \\ S_5 = \{Ind_3(G), Ind_3(B)\} = \{G_3, B_3\}, S_6 = \{Ind_3(Y), Ind_3(R)\} = \{Y_3, R_3\}.$$

Now, let us apply union operation to these sets and create new multiset of objects S , i.e.

$$S = \{G_1, Y_1\} / T(S_1) \cup \{R_1, B_1\} / T(S_2) \cup \{G_2, R_2\} / T(S_3) \cup \{Y_2, B_2\} / T(S_4) \cup \{G_3, B_3\} / T(S_5) \cup \\ \cup \{Y_3, R_3\} / T(S_6) = \{G_1, Y_1, R_1, B_1, G_2, R_2, Y_2, B_2, G_3, B_3, Y_3, R_3\} / T(S),$$

where S is a multiset of objects, and $T(S)$ is its class. As the result we obtained multiset of objects S which consists of four objects G, Y, R, B and their copies, which can be accurately identified. We can consider S as one of possible projects of future electric garland. ♠

Now we can formulate and prove the following proposition.

Proposition 2. *The quantity of all possible subsets of sets of objects S , which were obtained using D2 decomposition, can be calculated by the following formula*

$$q(S_w) = 2^n - 2n - 2,$$

where $n = |S|$.

Proof. From the previous section, we know that the quantity of all possible subsets of set of objects can be calculated as $q(S_w) = 2^n - n - 1$, where $n = |S|$. However, we can observe that the result of D2 decomposition of set of objects $S = \{G, Y, R, B\}$ does not contain subsets of cardinality 3 and 4, i.e. $n-1$ and n . It is true for any set of objects, because only sets of cardinality n and $n-1$ cannot be divided according to principle of D2 decomposition. Clearly, that for each set of objects S of cardinality n , only one subset of cardinality n exists. Concerning subsets of cardinality $n-1$, their quantity can be calculated as

$$C_n^{n-1} = \frac{n!}{(n-1)!(n-(n-1))!} = \frac{n!}{(n-1)!1!} = \frac{n!}{(n-1)!} = n,$$

it follows from the proof of Theorem 7. Considering all these facts, we can conclude that

$$q(S_w) = 2^n - n - 1 - n - 1 = 2^n - 2n - 2,$$

where $n = |S|$. □

Using Proposition 2 and the scheme of creation of multiset of objects from Example 15, we can represent our D2 constructor as follows

$$D2(S) = \bigcup_{w=1}^{2^n - 2n - 2} (S_1 \cup S_2),$$

where $n = |S|$ and $S_1, S_2 \subseteq S$ are disjoint sets of objects, such that $S_1 \cup S_2 = S$.

As we can see, D2 constructor gives us determined scheme for creation of multiset of objects. We also can calculate multiplicity of every object and cardinality of the multiset before its creation. As a proof of these facts, we can formulate and prove two following theorems.

Theorem 8. *The cardinality of each multiset of objects S , which is obtained using D2 constructor, can be calculated by the following formula*

$$|S| = \frac{n2^n}{2} - n^2 - n,$$

where $n = |S_b|$.

Proof. According to Theorem 6, cardinality of each multiset of objects S , which is obtained using PS Constructor, can be calculated by the following formula

$$|S| = \frac{n2^n}{2} - n,$$

where $n = |S_b|$. From proof of Proposition 2, we know that result of D2 decomposition of set of objects S does not contain subsets of cardinality $n-1$, n and quantity of such subsets of objects will be equal n and 1 respectively. That is why we can conclude that

$$|S| = \frac{n2^n}{2} - n(n-1) - n - n = \frac{n2^n}{2} - n^2 + n - 2n = \frac{n2^n}{2} - n^2 - n,$$

where $n = |S_b|$. □

Theorem 9. Multiplicity of each object A_i from multiset of objects S , which is obtained using D2 constructor, can be calculated by the following formula

$$m(A_i) = 2^{n-1} - n - 1,$$

where $n = |S_b|$.

Proof. From proof of Theorem 7 we know that it is possible to build only C_{n-1}^{k-1} subsets of cardinality k for set of objects S_b , where $|S_b| = n$. In addition, we know that each object $A_i \in S$ has the same multiplicity. Using these facts, we can conclude that

$$\begin{aligned} m(A_i) &= \frac{|S|}{|S_b|} = \frac{\frac{n2^n}{2} - n^2 - n}{n} = \left(\frac{n2^n}{2} - n^2 - n \right) \frac{1}{n} = \frac{n2^n}{2n} - \frac{n^2}{n} - \frac{n}{n} = \frac{2^n}{2} - n - 1 = \\ &= \frac{2^n - 2n - 2}{2} = \frac{2(2^{n-1} - n - 1)}{2} = 2^{n-1} - n - 1, \end{aligned}$$

where $n = |S_b|$. □

Let us consider proof of Theorem 9. It shows that multiplicity of every object A_i from multiset of objects S can be calculated as a sum of appropriate binomial coefficients. Using this fact, we can build a part of Pascal's triangle. However, in contrast to original Pascal's triangle, we will combine its part with results of Theorem 8 and Theorem 9. It is convenient to formulate this as following corollary.

Corollary 9.1. We can calculate cardinality, multiplicity of every object from the multiset of objects, which was created using D2 constructor, and quantity of subsets of objects which were used for its creation, using the following matrix

| $m(A_k)$ | $ S $ | $q(S_w)$ | $ S_b $ | 2 | 3 | 4 | 5 | 6 | ... |
|----------|-------|----------|---------|-----|-----|-----|-----|-----|-----|
| 3 | 12 | 6 | 4 | 6 | | | | | |
| 10 | 50 | 20 | 5 | 10 | 10 | | | | |
| 25 | 150 | 50 | 6 | 15 | 20 | 15 | | | |
| 56 | 392 | 112 | 7 | 21 | 35 | 35 | 21 | | |
| 119 | 952 | 238 | 8 | 28 | 56 | 70 | 56 | 28 | |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |

where column $m(A_k)$ reflects multiplicity of object A_k in multiset of objects S ; column $|S|$ reflects cardinality of multiset of objects S ; column $q(S_w)$ reflects quantity of $S_w \subseteq S_b$ that was used for obtaining S ; column $|S_b|$ reflects cardinality of basic sets of objects; first row starting with 5-th column reflects quantity of subsets of objects of certain cardinality, where cardinality coincides with the value of $a_{1,j \geq 5}$.

The elements of column $m(A_k)$ can be calculated using Theorem 9 or using the following formula

$$a_{i \geq 2, 1} = \begin{cases} 3, & i = 2; \\ \sum_{i \geq 2, j \geq 4} a_{i-1, j}, & i > 2. \end{cases}$$

The elements of column $|S|$ can be calculated using Theorem 8 or using the following formula

$$a_{i \geq 2, 2} = \sum_{i \geq 2, j \geq 5} a_{i, j} \cdot a_{1, j}.$$

The elements of column $q(S_w)$ can be calculated using Proposition 2 or using the following formula

$$a_{i \geq 2, 3} = \sum_{i \geq 2, j \geq 5} a_{i, j}.$$

The element $a_{i \geq 2, j \geq 5}$ of matrix can be calculated in such a way

$$a_{i \geq 2, j \geq 5} = \begin{cases} 6, & i = 2, j = 5; \\ a_{i-1, j-1} + a_{1, j+1}, & j > 5, j - i = 3; \\ a_{i-1, j-1} + a_{i-1, j}, & j - i < 3; \end{cases}$$

or using the following formula

$$a_{i \geq 2, j \geq 5} = \frac{a_{i, 4}!}{a_{1, j}! (a_{i, 4} - a_{1, j})!}.$$

Conclusions

This paper presents certain approach for modeling of some aspects of human thinking, in particular creation of sets and multisets of objects, within constructive object-oriented version of set theory, which was proposed in [Terletskyi, 2014]. The creation of sets and multisets of objects is considered from different sides, in particular classical set theory, object-oriented programming and development of intelligent information systems.

The paper also presents universal constructor of multisets of objects that gives us a possibility to create arbitrary multisets of objects and to recognize (identify) every copy of particular object, which have multiplicity $m \geq 2$. In addition, a few determined constructors of multisets of objects, which allow to create multisets, using strictly

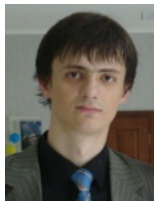
defined schemas, are also presented in the paper. The author proposed methods for calculation multiplicity of each object and cardinality of multiset before its creation for each constructor. That makes them very useful in cases of very big cardinalities of multisets.

The proposed approach for modeling of creation of sets and multisets of objects allows not only creation (generation) of sets and multisets of objects, but also their classification. It gives us an opportunity to consider the problem of object classification and identification in another way. The presented constructors of multisets of objects allow us to model corresponding processes of human thought, that in turn give us an opportunity to develop intelligent information systems, using these tools.

Bibliography

- [Cantor, 1915] G. Cantor. Contributions to the Founding of the Theory of Transfinite Numbers. New York: Dover Publications, Inc., 1915.
- [Eckel, 2006] B. Eckel. Thinking in Java: 4-th Edition. Prentice Hall, 2006.
- [Fraenkel, Bar-Hillel, 1958] A. A. Fraenkel, Ye. Bar-Hillel. Foundations of set theory. North-Holland Publishing Company, 1958.
- [Mostowski, 1969] A. Mostowski. Constructible Sets with Applications. North-Holland Publishing Company, 1969.
- [Mukherjee, 2012] S. Mukherjee. .NET 4.0 Generics: Beginner's Guide. Packt Publishing Ltd. 2012.
- [Musser, Derge, Saini, 2001] D. R. Musser, G. J. Derge, A. Saini. STL Tutorial and Reference Guide: 2-nd edition. C++ Programming with the Standard Template Library, Addison-Wesley Professional, 2001.
- [Pecinovskiy, 2013] R. Pecinovskiy. OOP – Learn Object Oriented Thinking and Programming. Tomas Bruckner, Repin-Zivonin, 2013.
- [Summerfield, 2010] M. Summerfield. Programming in Python 3. A Complete Introduction to the Python Language: 2-nd edition. Pearson Education, Inc. 2010.
- [Syropoulos, 2001] A. Syropoulos. Mathematics of multisets. In Proceedings of the “Workshop on Multiset Processing: Multiset Processing, Mathematical, Computer Science, and Molecular Computing Points of View”, Springer-Verlag Berlin Heidelberg, 2001, pp. 347-358.
- [Terletskiy, 2014] D. O. Terletskiy. Constructors of Sets and Multisets of Objects. Scientific Journal Problems in Programing, N 1, 2014, pp. 18-30 (In Ukrainian).
- [Vopenka, 1979] P. Vopenka. Mathematics in the alternative set theory. Leipzig: BSB B.G. Teubner, 1979.
- [Wang, Mc Naughton, 1953] H. Wang and R. Mc Naughton. Les Systemes Axiomatiques de la Theorie des Ensembles. Paris: Gauthier-Villars, 1953.
- [Weisfeld, 2008] M. Weisfeld. The Object-Oriented Thought Process. Third Edition. Addison-Wesley Professional, 2008.

Authors' Information



Dmytro Terletskiy – Postgraduate student, Department of Information Systems, Faculty of Cybernetics, Taras Shevchenko National University of Kyiv, 03680, 4d Glushkov Avenue, Kyiv, Ukraine; e-mail: dmytro.terletskiy@gmail.com
Major Fields of Scientific Research: Artificial Intelligence, Discrete Mathematics, Programming, Software Engineering.

WORDARM - A SYSTEM FOR STORING DICTIONARIES AND THESAURUSES BY NATURAL LANGUAGE ADDRESSING

Krassimira Ivanova

Abstract: *The main features of WordArM system for storing dictionaries and thesauruses by means of Natural Language Addressing are outlined in the paper. Experiments with WordArM have shown that the NL-addressing is suitable for dynamic processes of creating and further development of datasets due to avoiding recompilation of the database index structures and high speed access to every data element.*

Keywords: *Natural Language Addressing*

ACM Classification Keywords: *H.2 Database Management; H.2.8 Database Applications*

Introduction

Let remember, that the idea of Natural Language Addressing (NLA) [Ivanova et al, 2012a; 2012b; Ivanova et al, 2013a; 2013b; 2013c; 2013d; 2013e; Ivanova, 2013; Ivanova, 2014] consists in using the computer encoding of name's (concept's) letters as logical address of connected to it information stored in a multi-dimensional numbered information spaces [Markov, 1984; Markov, 2004; Markov, 2004a]. This way no indexes are needed and high speed direct access to the text elements is available. It is similar to the natural order addressing in a dictionary where no explicit index is used but the concept by itself locates the definition.

In this paper we present program system WordArM based on NLA Access Method and corresponded NLA Archive Manager called NL-ArM [Ivanova, 2014]. Below we will present main features of WordArM.

WordArM

WordArM is a system for storing dictionaries and thesauruses by means of Natural Language Addressing.

WordArM is upgrade over Natural Language Addressing Access Method and corresponded Archive Manager called **NL-ArM**, realized in [Ivanova, 2014]. WordArM is aimed to store libraries of terms and their definitions. WordArM concepts are organized in multi-layer hash tables (information spaces with variable size). The definition of each term is stored in a container located by appropriate path - mapping of the natural language word or phrase, which presents the concept.

There is no limit on the number of terms in a WordArM archive, but their total length plus internal hash indexes could not exceed the file length (4G, 8G, etc.) which is enough space for several millions of concepts' definitions. There is no limit on the number of files in the data base, as well as their location, including the Internet. This permits to store unlimited number of concepts' definitions.

WordArM has two modes of operation: Automated and Manual.

- *The automated mode* supports reading the input information from file (concepts with definitions to be stored in the archive or only list of concepts to receive their definitions from the archive). The result is storing the definitions in the WordArM archive or exporting definitions from the WordArM archive in a file;
- *The manual mode* does the same but only for one concept which is entered manually from the corresponded screen panel.

To support these modes, WordArM has two main operations – information storing (NLA-Write) and information reading (NLA-Read), which have two variants – for automatic input and output of data from and to files, and for manually performing these operations.

WordArM automated mode functions

The WordArM panel for working in automated mode is shown on Figure 1.

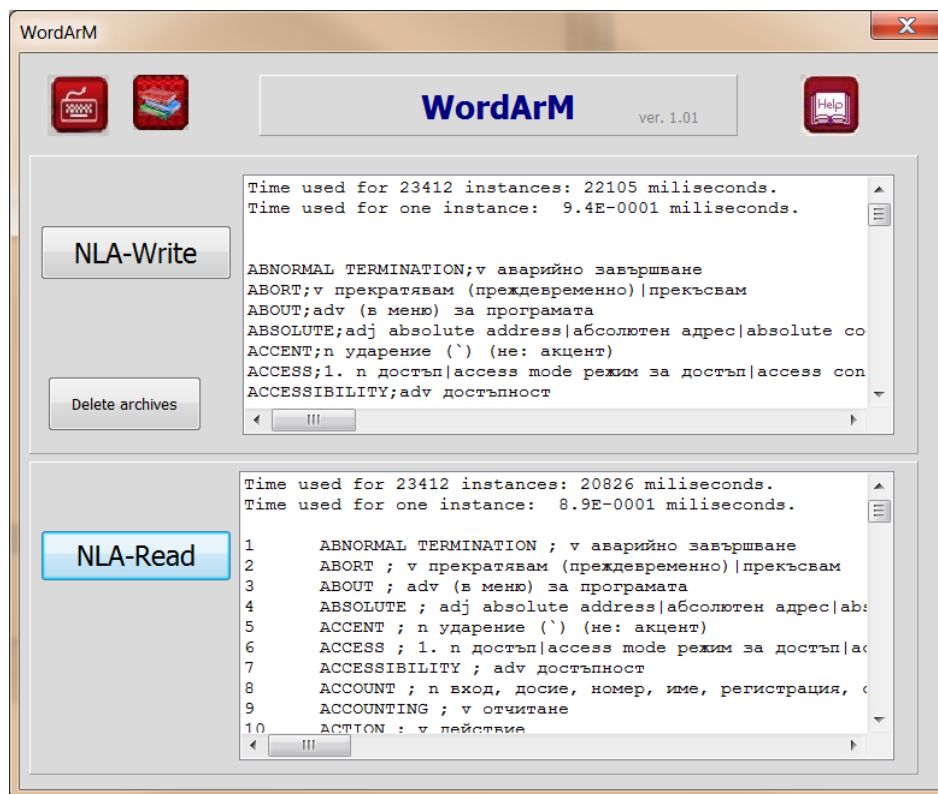


Figure 1. Screenshot of WordArM panel for working in automated mode

By “NLA-Write” button the function for storing definitions from a file can be activated. Each concept and its definition occupy one record in the file. There is no limit for the number of records in the file. After pressing the “NLA-Write” button, the system reads records sequentially from the file and for each of them:

- Transform the concept into path;
- Store the definition of this concept in the container located by the path.

The input file is in CSV file format. Its records have the next format: **<word/words>;<definition><CR>**.

After storing the concepts' definitions, WordArM displays the contents of the input file in the field near to the “NLA-Write” button. Before the information from the file, two informative lines are shown (Figure 1):

- Total time used for storing all instances from the file;
- Average time used for storing of one instance

in milliseconds.

In the same panel (Figure 1) corresponded button enables deleting the work archive of the WordArM (ArmDict.dat, which in this version for test control is stored on the hard disk but not in the computer memory). WordArM is completed with compressing program and after storing the information prepares small archive for long time storage.

By "NLA-Read" button, the function for reading definitions from the WordArM archive can be activated. In the automated mode, NLA-Read uses as input a file with concepts (each on a separate line) and extract from the archive theirs definitions. If any definition does not exist, the output is empty definition.

Each concept and its definition occupy one record in the output file. There is no limit to the number of records in the file. After pressing the "NLA-Read" button, the system reads concepts sequentially from the input file and for each of them:

- Transform the concept into path;
- Extract the definition of this concept from the container located by the path.




The output file is in CSV file format: **<word/words>;<definition><CR>**.

The content of the output file is displayed in the field next to the NLA-Read button. Before the information from the file, two informative lines are shown (Figure 1):

- Total time used for extracting of all instances;
- Average time used for extracting of one instance,

in milliseconds.

Finally, the form has three service buttons:

- The first () serves as a transition to the form for manual input and output of data to/from the system archive;
- The second () is connected to the module for adjusting the environment of the system – archives, input and output information, etc.;
- The third () activates the help text (user guide) of the system.

The exit from the system can be done by the conventional way for Windows - by clicking on the cross in the upper right corner of the form.

WordArM manual mode functions

The WordArM panel for working in manual mode is shown on Figure 2. The NL-addressing supports multi-language work. In other words, in the same archive we may have definitions of the concepts from different languages.

By "NLA-Write" button the function for storing definitions from the form can be activated. Each concept and its definition can be given in corresponded fields on the screen form. After pressing the "NLA-Write" button, the system reads information from the fields and:

- Transform the concept into path;
- Store the definition of this concept in the container located by the path.

By "NLA-Read" button the function for reading a definition from the WordArM archive can be activated. In the manual mode, NLA-Read uses as input the concept given in the screen field and extracts from the archive its definition. If the definition does not exist, the output is empty definition. After pressing the "NLA-Read" button, the system reads concept from the screen field and:

- Transform the concept into path;
- Extract the definition of this concept from the container located by the path.

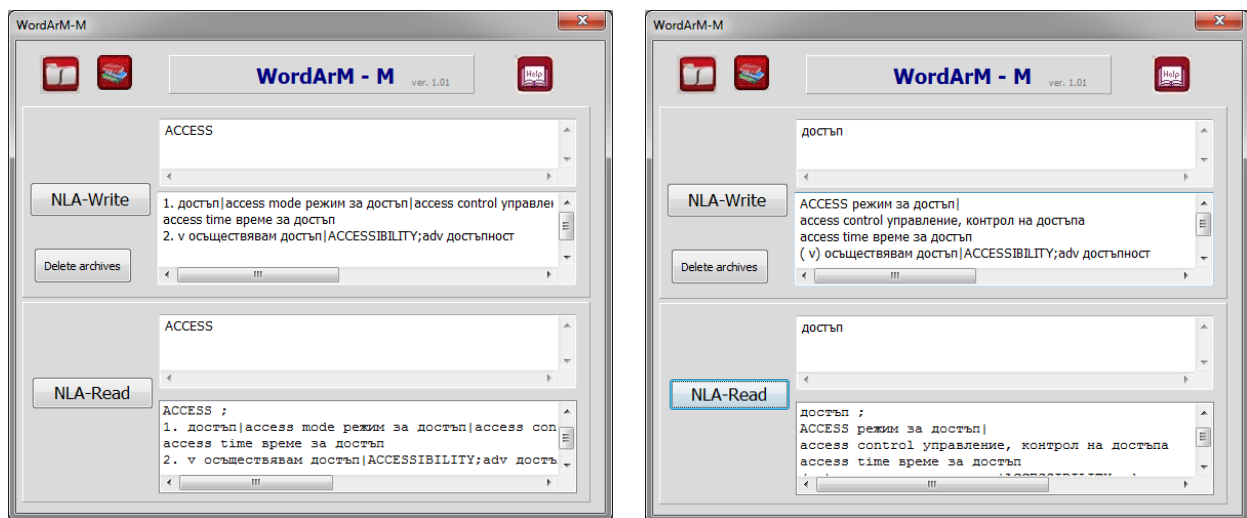


Figure 2. Screenshots of WordArM panel for working in manual mode showing simultaneous work with concepts defined in different languages

The fields for manual work allow copy/past options (Figure 3a and Figure 3b).

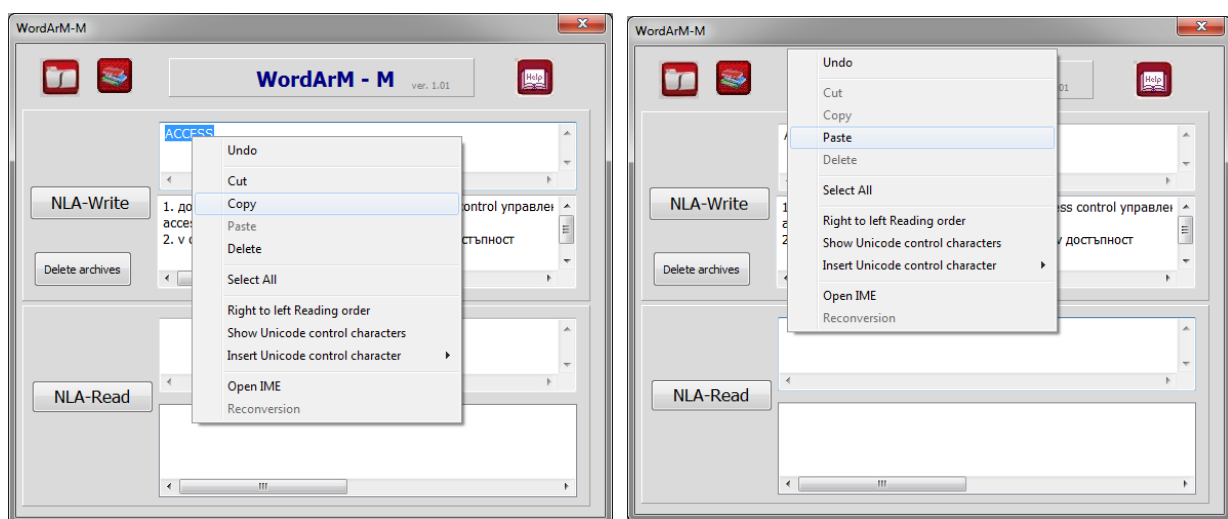


Figure 3a. Copy from the input field

Figure 3b. Past in the field for reading

The service buttons has similar functions as the same in the automated mode.

Experiments with WordArM

Time measured during the experiments presented below is highly dependent on the possibilities of operational environment and speed of computer hardware. We provided experiments on the next computer configuration:

- Processor: Intel Core2 Duo T9550 2.66GHz; CPU Launched: 2009;
- Physical Memory: 4.00 GB;
- Hard Disk: 100 GB data partition; 2 GB swap;
- Operating System: 64-bit operating system Windows 7 Ultimate SP1.

Theoretical background of WordArM was presented in [Ivanova et al, 2013c]. Below we will remember main results from it.

➤ NL-storing dictionaries

Our first experiment was to realize a small multi-language dictionary based on NL-addressing. We have taken data from the popular in Bulgaria "SA Dictionary" [Angelov, 2012]. SA Dictionary is a computer dictionary, which translates words from Bulgarian language to English and vice versa.

For experiments we used a list of **23 412** words in English and Bulgarian with their definitions in Bulgarian, stored in a sequential file with size of 2 410 KB.

For storing dictionaries we used simple model: the words (concepts) are used as paths to theirs definitions stored in corresponded terminal containers.

The speed for storing, accessing, and size of the work memory and permanent archives are given in Table 1.

Work memory is the memory taken for storing hash tables and service information. Usually it has to be part of main computer memory. To analyze its real size in our experiments, work memory is allocated as file.

Permanent archives are static copies of work memory (zipped files), aimed for long storing the information. They have to be of small size and converting to and from expanded work memory structures has to be quick (usually several seconds or minutes). For compressing of work memory we use a separate archiving program.

Table 1. Experimental data for NL-storing of a dictionary

| operation | number of instances | total time in milliseconds | average time for one instance | work memory | permanent archive |
|------------|---------------------|----------------------------|-------------------------------|-------------|-------------------|
| NL-writing | 23 412 | 22 105 | 0.94 ms | 80 898 KB | 5 938 KB |
| NL-reading | 23 412 | 20 826 | 0.89 ms | | |

The work memory taken during the work was **80 898** KB.

After finishing the work, occupied permanent compressed archive is **5 938** KB. This means that the NL-indexing takes 5 102 KB additional compressed disk memory (the sequential file with initial data is 2 410 KB and compressed by WinZip it is 836 KB).

To analyze work of the system, work memory was chosen to be in a file but not in the main memory. In further realizations of WordArM, it may be realized as a part of main memory of computer as:

- Dynamically allocated memory;
- File mapped in memory.

In this case, the speed of storing and accessing will be accelerated and used hard disk space will be reduced.

The analysis of the results in Table 1 shows that the NL-addressing in this realization permits access practically equal for writing and reading for all data. The speed is more than a thousand instances per second. *Reading is possible immediately after writing and no search indexes are created.*

➤ NL-storing thesauruses

We have used NL-addressing to realize the WordNet lexical database [WordNet, 2012]. The WordNet database organization has the following important disadvantages: 1) Relative addressing is convenient for the computer processing, but it is difficult to be used by the customer; 2) Manual creating of numerical addresses is impossible, and their use can be done only by the special program; 3) The end user has access only to the static ("compiled") version of the database, which couldn't be extended and further developed; 4) Building the WordNet database requires the use of the "Grinder" program and the processing of all lexicographers' source files at the same time; 5) Using the current format is not only cumbersome and error-prone, but also limits what can be expressed in a WordNet resource.

The main source information of WordNet is published as lexicographer files. The total number of instances (file records) is 117 871, but 206 instances contain service information (not concepts' definitions), so we have 117 665 instances for experiments, distributed in 45 thematically organized lexicographer files. It is important to note that there is equal synsets in several lexicographer files. This has matter when we integrate the 45 files in one source file for representing a thesaurus, i.e. for experiment we have used all 45 files concatenated in one big file as thesaurus with more than one hundred thousands of concepts. The results are given in Table 2. A screenshot from the WordArM for the case of WordNet as thesaurus is shown at Figure 4.

We receive practically the same results as for storing dictionaries.

The analysis of the results in Table 2 shows that the NL-addressing permits access practically equal for writing and reading for all data. The speed is more than a thousand instances per second. Reading is possible immediately after writing and no search indexes are created.

The work memory for hash tables and their containers taken during the work of WordArM was 385 538 KB. To analyze work of the system, work memory was chosen to be in a file in the external memory. In further realizations, to accelerate the speed and reduce of used disk space, the work memory may be realized as part of main memory (as dynamically allocated memory or as file mapped in memory).

After finishing the work, occupied permanent archive for compressed archive is 15 603 KB, i.e. in this case the NL-indexing takes 14 270 KB additional compressed memory (the sequential file with initial data is 1 333 KB).

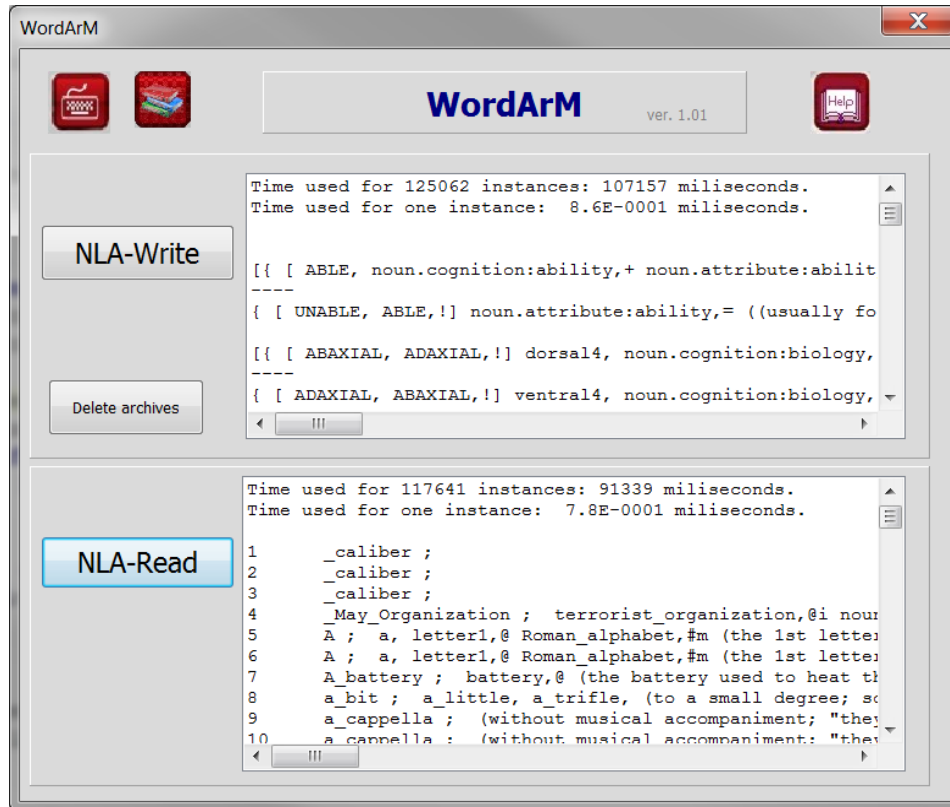


Figure 4. WordArM results for the case of WordNet as thesaurus

Table 2. Experimental data for storing WordNet as thesaurus

| operation | number of instances | total time in milliseconds | average time for one instance |
|--|---------------------|----------------------------|-------------------------------|
| writing | 125 062 | 107 157 | 0.86 ms |
| reading | 117 641 | 91 339 | 0.78 ms |
| work memory: 385 538 KB; permanent archive: 15 603 KB; source text: 1 333 KB | | | |

Conclusion

In this paper we presented program system WordArM based on NLA Access Method and corresponded NLA Archive Manager called NL-ArM [Ivanova, 2014]. Main features of WordArM were outlined.

WordArM is aimed to support experiments. Because of this it was realized with two modes – automatic and manual. In addition, work memory of the system was realized as disk files for analyzing its behavior during the experiments.

Analyzing results from the experiment with a real dictionary data we may conclude that it is possible to use NL-addressing for storing such information. Next experiment was aimed to answer to question: "What we gain and loss using NL-Addressing for storing thesauruses?"

The loss is additional memory for storing hash structures which serve NL-addressing. But the same if no great losses we will have if we will build balanced search trees or other kind in external indexing. It is difficult to compare with other systems because such information practically is not published. The benefit is in two main achievements:

- High speed for storing and accessing the information;
- The possibility to access the information immediately after storing without recompilation the database and rebuilding the indexes.

Main conclusion is that for static structured datasets it is more convenient to use standard utilities and complicated indexes. NL-addressing is suitable for dynamic processes of creating and further development of datasets due to avoiding recompilation of the database index structures and high speed access to every data element.

Bibliography

[Angelov, 2012] St. Angelov. SA Dictionary <http://www.thediction.com/> (accessed: 11.01.2013)

[Ivanova et al, 2012a] Krassimira Ivanova, Vitalii Velychko, Krassimir Markov. "About NL-addressing" (К вопросу о естественно-языковой адрессации) In: V. Velychko et al (ed.), Problems of Computer in Intellectualization. ITHEA® 2012, Kiev, Ukraine - Sofia, Bulgaria, ISBN: 978-954-16-0061 0 (printed), ISBN: 978-954-16-0062-7 (online), pp. 77-83 (in Russian).

[Ivanova et al, 2012b] Krassimira Ivanova, Vitalii Velychko, Krassimir Markov. "Storing RDF Graphs using NL-addressing", In: G. Setlak, M. Alexandrov, K. Markov (ed.), Artificial Intelligence Methods and Techniques for Business and Engineering Applications. ITHEA® 2012, Rzeszow, Poland; Sofia, Bulgaria, ISBN: 978-954-16-0057-3 (printed), ISBN: 978-954-16-0058-0 (online), pp. 84 – 98.

[Ivanova et al, 2013a] Krassimira B. Ivanova, Koen Vanhoof, Krassimir Markov, Vitalii Velychko, "Introduction to the Natural Language Addressing", International Journal "Information Technologies & Knowledge" Vol.7, Number 2, 2013, ISSN 1313-0455 (printed), 1313-048X (online), pp. 139–146.

[Ivanova et al, 2013b] Krassimira B. Ivanova, Koen Vanhoof, Krassimir Markov, Vitalii Velychko, "Introduction to Storing Graphs by NL-Addressing", International Journal "Information Theories and Applications", Vol. 20, Number 3, 2013, ISSN 1310-0513 (printed), 1313-0463 (online), pp. 263 – 284.

[Ivanova et al, 2013c] Krassimira B. Ivanova, Koen Vanhoof, Krassimir Markov, Vitalii Velychko, "Storing Dictionaries and Thesauruses Using NL-Addressing", International Journal "Information Models and Analyses" Vol.2, Number 3, 2013, ISSN 1314-6416 (printed), 1314-6432(online), pp. 239 - 251.

[Ivanova et al, 2013d] Krassimira B. Ivanova, Koen Vanhoof, Krassimir Markov, Vitalii Velychko, "The Natural Language Addressing Approach", International Scientific Conference "Modern Informatics: Problems, Achievements, and Prospects of Development", devoted to the 90th anniversary of academician V. M. Glushkov. Kiev, Ukraine, 2013, ISBN 978-966-02-6928-6, pp. 214 - 215.

[Ivanova et al, 2013e] Krassimira B. Ivanova, Koen Vanhoof, Krassimir Markov, Vitalii Velychko, "Storing Ontologies by NL-Addressing", IVth All-Russian Conference "Knowledge-Ontology-Theory" (KONT-13), Novosibirsk, Russia, 2013, ISSN 0568 661X, pp. 175 - 184.

[Ivanova, 2013] Krassimira Ivanova, "Informational and Information models", In Proceedings of 3rd International conference "Knowledge Management and Competitive Intelligence" in the frame of 17th International Forum of Young Scientists "Radio Electronics and Youth in the XXI Century", Kharkov National University of Radio Electronics (KNURE), Kharkov, Ukraine, Vol.9, 2013, pp 6-7.

- [Ivanova, 2014] Krasimira Ivanova, "Storing Data using Natural Language Addressing", PhD Thesis, Hasselt University, Belgium, 2014
- [Markov, 1984] Krassimir Markov, "A Multi-domain Access Method", Proceedings of the International Conference on Computer Based Scientific Research, PLOVDIV, 1984, pp. 558 - 563.
- [Markov, 2004] Krassimir Markov, "Multi-domain information model", Int. J. Information Theories and Applications, 11/4, 2004, pp. 303 - 308
- [Markov, 2004a] Krassimir Markov, "Co-ordinate based physical organization for computer representation of information spaces", (Координатно базирана физическа организация за компютърно представяне на информационни пространства) Proceedings of the Second International Conference "Information Research, Applications and Education" i.TECH 2004, Varna, Bulgaria, Sofia, FOI-COMMERCE – 2004, стр. 163 - 172 (in Bulgarian).
- [WordNet, 2012] Princeton University "About WordNet", WordNet, Princeton University, 2010 <http://WordNet.princeton.edu> (accessed: 23.07.2012)

Authors' Information



Ivanova Krassimira – *University of National and World Economy, Sofia, Bulgaria;*
e-mail: krazy78@mail.bg

*Major Fields of Scientific Research: Software Engineering, Business Informatics, Data Mining,
Multidimensional multi-layer data structures in self-structured systems*

AN ALGORITHM FOR FACTORING COMPOSITE POLYNOMIAL $P(x^p - x - \delta)$

Sergey Abrahamyan, Knarik Kyureghyan

Abstract: Let $P(x) = x^n + a_{n-1}x^{n-1} + \dots + a_1x + a_0$ be an irreducible polynomial over F_q . In [Cao, 2012, Varshamov, 1973, Lidl, 1987] the factorization of the composite polynomial $P(x^p - ax - \delta)$, when $a = 1$ and $\text{Tr}_{F_q/F_p}(nb - a_{n-1}) = 0$ is considered. The result of factorization of polynomial $P(x^p - x - \delta)$ is a p irreducible polynomials of degree n over F_q . In this paper we propose an algorithm for factoring composite polynomial $P(x^p - x - \delta)$ over F_q and give a explicit view of each factor.

Keywords: finite field, polynomial factorization, polynomial composition

ACM Classification Keywords: I.1.2. Algorithms

Introduction

Construction of irreducible polynomials from given irreducible polynomial is a classic problem of finite field theory and computer algebra. One of methods to construct irreducible polynomials is the polynomial composition method. Such methods have been studied by several authors including Varshamov [Varshamov, 1984], Cohen [Cohen, 1992], Meyn [Meyn, 1990], Kyureghyan [Kyureghyan, 2011].

Let F_q be the Galois field of order $q = p^s$, where p is a prime and s is a natural number and F_q^* be its multiplicative group. Let $P(x) = x^n + a_{n-1}x^{n-1} + \dots + a_1x + a_0$ be an irreducible polynomial over F_q . Varshamov proved that for $a = 1$ the composite polynomial $P(x^p - ax - \delta)$ is irreducible over F_q if and only if $\text{Tr}_{F_q/F_p}(n\delta - a_{n-1}) \neq 0$. In [Lidl, 1987, Varshamov, 1973] the problem of factorization of the composite polynomial $P(x^p - x - \delta)$, when $\text{Tr}_{F_q/F_p}(n\delta - a_{n-1}) = 0$ is considered. Also, in [Cao, 2012] a short proof of above-mentioned problem is given. For constructing p irreducible polynomials from the given irreducible polynomial we need compute the composition $P(x^p - x - \delta)$, and then factorize $P(x^p - x - \delta)$. In this paper we show how factors of polynomial $P(x^p - x - b)$ are connected each other. Also, we propose a probabilistic algorithm based on Cantor Zassenhaus's algorithm for finding one of factors of polynomial $P(x^p - x - \delta)$.

Factorization of composite polynomial $P(x^p - x - \delta)$

Recall that the trace function of F_{q^n} over F_q is

$$\text{Tr}_{q^n/q}(\alpha) = \sum_{i=0}^{n-1} \alpha^{q^i}, \quad \alpha \in F_{q^n}.$$

Define $\text{Tr}_{q^n/q}^{(i)}(\alpha)$ the following way

$$\text{Tr}_{q^n/q}^{(i)}(\alpha) = \sum_{0 \leq j_1 < \dots < j_i \leq n-1} \alpha^{q^{j_1}} \alpha^{q^{j_2}} \dots \alpha^{q^{j_i}},$$

here $\text{Tr}_{q^n/q}^{(1)}(\alpha) = \text{Tr}_{q^n/q}(\alpha)$.

Let $f(x) = \sum_{i=0}^{n-1} g_i x^i$ be a minimal polynomial of α . It is easy to see that

$$g_i = (-1)^{n-i} Tr_{q^n/q}^{(n-i)}(\alpha). \quad (1)$$

In this section based on Proposition 1 (introduced below) we show how connected factors of polynomial $P(x^p - x - \delta)$ over F_q .

Proposition 1. (Theorem 2.1 [Cao, 2012]) Let $g(x) = x^n + a_{n-1}x^{n-1} + \dots + a_0$ be an irreducible polynomial over $F_q = F_{p^s}$ of degree n . Let $\delta \in F_q$ and $Tr_{q/p}(n\delta - a_{n-1}) = 0$. Then $g(x^p - x - \delta)$ decomposes as a product of p irreducible polynomials over F_q of degree n . Let $g(x^p - x - \delta) = u_0(x)u_1(x) \dots u_{p-1}(x)$. Then via a suitable assignment of the indexes of the factors, $u_k(x) = u_0(x + k)$ for $k = 0, 1, \dots, p-1, \dots$

In our proof we will need the following proposition.

Proposition 2. (Theorem 2.25 [Lidl, 1987]) Let F be a finite extension of K . Then for $\alpha \in F$ we have $Tr_{F/K}(\alpha) = 0$ if and only if $\alpha = \beta^q - \beta$ for some $\beta \in F$.

Theorem 1. Let $q = p^s$, where p is a prime. $P(x) = \sum_{u=0}^n a_u x^u$ be an monic irreducible polynomial of degree n over F_q and $Tr_{q/p}(n\delta - a_{n-1}) = 0$. Then the polynomial $F(x) = P(x^p - x - \delta)$, $\delta \in F_q$ factors to p irreducible polynomials of degree n over F_q as follows: $F(x) = G_0(x)G_1(x) \dots G_{p-1}(x)$, where

$$G_0(x) = x^n + g_{n-1}x^{n-1} + \dots + g_1x + g_0,$$

$$G_k(x) = x^n + g_{n-1}^{(k)}x^{n-1} + \dots + g_1^{(k)}x + g_0^{(k)} \quad k = 1, 2, \dots, p-1$$

$$\text{and } g_i^{(k)} = \sum_{v=0}^{n-i} (-1)^{n+v-i} k^{n-v-i} \binom{n-v}{i} g_{n-v} \quad i = 0, 1, 2, \dots, n.$$

Proof 1. Let $\alpha \in F_{q^n}$ be a root of $P(x)$. Then we have

$$P(x) = \prod_{i=0}^{n-1} (x - \alpha^{q^i}) \quad (2)$$

Substituting $x^p - x - \delta$ for x in (2), we will derive

$$F(x) = P(x^p - x - \delta) = \prod_{i=0}^{n-1} (x^p - x - \delta - \alpha^{q^i}) = \prod_{i=0}^{n-1} (x^p - x - (\delta + \alpha)^{q^i}) \quad (3)$$

Let us consider the polynomial $l(x) = x^p - x - (\delta + \alpha)$.

By proposition 2 $l(x)$ has a root in F_{q^n} if and only if $Tr_{q^n/p}(\delta + \alpha) = 0$.

Now we compute $Tr_{q^n/p}(\delta + \alpha)$.

$$Tr_{q^n/p}(\delta + \alpha) = Tr_{q^n/p}(Tr_{q^n/q}(\delta + \alpha)) = Tr_{q/p}(n\delta + Tr_{q^n/q}(\alpha)) = Tr_{q/p}(n\delta - a_{n-1})$$

which is equal to 0 by condition of theorem. So we have that $l(x)$ has a root in F_{q^n} .

Let $\gamma \in F_{q^n}$ be a root of $l(x)$, that is

$$\gamma^p - \gamma - (\delta + \alpha) = 0.$$

Considering that $\alpha = \gamma^p - \gamma - \delta$ one can see that $F_q(\gamma) \supseteq F_q(\alpha) = F_{q^n}$, therefore γ is proper element of F_{q^n} . It is easy to see that p roots of $x^p - x - (\delta + \alpha)$ are $\gamma + k$, $k = 0, 1, \dots, p-1$. Clearly, $\gamma^{q^i} + k$, $k \in F_p$ are all the roots of $x^p - x - (\delta + \alpha)^{q^i}$.

Hence from (3) we have

$$F(x) = \prod_{i=0}^{n-1} \prod_{k=0}^{p-1} (x - \gamma^{q^i} - k) = \prod_{k=0}^{p-1} \left(\prod_{i=0}^{n-1} (x - \gamma^{q^i} - k) \right).$$

Denote

$$G_k(x) = \prod_{i=0}^{n-1} (x - \gamma^{q^i} - k).$$

It is obvious $G_k(x)$ is the minimal polynomial of $\gamma + k$, where $k = 0, 1, \dots, p - 1$ and $G_k(x) = G_0(x - k)$. Thus $G_k(x)$ is a irreducible polynomial over F_q .

Let $G_0(x) = x^n + g_{n-1}x^{n-1} + \dots + g_1x + g_0$ and $G_k(x) = x^n + g_{n-1}^{(k)}x^{n-1} + \dots + g_1^{(k)}x + g_0^{(k)}$.

From (1) we have

$$g_i^{(k)} = (-1)^{n-i} Tr_{q^n/q}^{(n-i)}(\gamma + k) = (-1)^{n-i} \sum_{0 \leq j_1 < \dots < j_{n-i} \leq n-1} (\gamma + k)^{q^{j_1}} (\gamma + k)^{q^{j_2}} \dots (\gamma + k)^{q^{j_{n-i}}}.$$

Let us compute $g_i^{(k)} = (-1)^{n-i} Tr_{q^n/q}^{(n-i)}(\gamma + k)$.

$$g_i^{(k)} = (-1)^{n-i} \sum_{0 \leq j_1 < \dots < j_{n-i} \leq n-1} \left(k^{n-i} + k^{n-i-1} \sum_{\substack{j_1 \leq u_1 \leq j_{n-i} \\ u_1 \in \{j_1 \dots j_{n-i}\}}} \gamma^{q^{u_1}} \right. \\ \left. + k^{n-i-2} \sum_{\substack{j_1 \leq u_1 < u_2 \leq j_{n-i} \\ u_1, u_2 \in \{j_1 \dots j_{n-i}\}}} \gamma^{q^{u_1}} \gamma^{q^{u_2}} + \dots + k \sum_{\substack{j_1 \leq u_1 < \dots < u_{n-i-1} \leq j_{n-i-1} \\ u_1, \dots, u_{n-i-1} \in \{j_1 \dots j_{n-i}\}}} \gamma^{q^{u_1}} \gamma^{q^{u_2}} \dots \gamma^{q^{u_{n-i-1}}} \right. \\ \left. + \gamma^{q^{j_1}} \gamma^{q^{j_2}} \dots \gamma^{q^{j_{n-i}}} \right) \tag{4}$$

Now we compute the following double sum

$$\sum_{0 \leq j_1 < \dots < j_{n-i} \leq n-1} \sum_{\substack{j_1 \leq u_1 < \dots < u_r \leq j_{n-i} \\ u_1, \dots, u_r \in \{j_1 \dots j_{n-i}\}}} \gamma^{q^{u_1}} \gamma^{q^{u_2}} \dots \gamma^{q^{u_r}} \quad r = 1, \dots, n - i - 1 \tag{5}$$

In the first and the second sums we have correspondingly $\binom{n-i}{n-i}$ and $\binom{n-i}{r}$ terms, and totally $\binom{n-i}{n-i} \cdot \binom{n-i}{r}$ terms. It is easy to see that in (5) each term is repeated equal times. On the other hand the sum

$$\sum_{0 \leq u_1 < u_2 < \dots < u_r \leq n-1} \gamma^{q^{u_1}} \gamma^{q^{u_2}} \dots \gamma^{q^{u_r}} \quad r = 1, \dots, n - i - 1 \tag{6}$$

contains the same terms found in (5) without any repetition, whereas in (6) contains $\binom{n-i}{r}$ terms.

So, one may conclude that

$$\sum_{0 \leq j_1 < \dots < j_{n-i} \leq n-1} \sum_{\substack{j_1 \leq u_1 < \dots < u_r \leq j_{n-i} \\ u_1, \dots, u_r \in \{j_1 \dots j_{n-i}\}}} \gamma^{q^{u_1}} \gamma^{q^{u_2}} \dots \gamma^{q^{u_r}} \\ = \frac{\binom{n-i}{n-i} \cdot \binom{n-i}{r}}{\binom{n-i}{r}} \sum_{0 \leq u_1 < \dots < u_r \leq n-1} \gamma^{q^{u_1}} \gamma^{q^{u_2}} \dots \gamma^{q^{u_r}}$$

$$= (-1)^r \binom{n-r}{i} g_{n-r} \quad r = 1, \dots, n-i-1 \quad (7)$$

Opening brackets in (4) and substituting (7) in (4) we get

$$g_i^{(k)} = \sum_{v=0}^{n-i} (-1)^{n+v-i} k^{n-v-i} \binom{n-v}{i} g_{n-v} \quad (8)$$

where $0 \leq i \leq n, 0 \leq k \leq p-1$.

So, for obtaining the polynomial $P(x^p - x - \delta)$ factors we need a single factor only. Rest factors may be computed by (8).

An algorithm for factoring polynomial $P(x^p - x - \delta)$

As seen from the proof of Theorem 1 a polynomial $P(x^p - x - \delta)$ has no repeated factors. Below we propose an equal degree factorization algorithm based on Cantor and Zassenhaus's algorithm [Cantor, 1981].

Let f be a monic square-free univariate polynomial over a finite field F_q of degree n with $r \geq 2$ irreducible factors f_1, \dots, f_r each of degree d . Since f_1, \dots, f_r are pairwise relatively prime, the Chinese Remainder Theorem provides the isomorphism:

$$\chi: F_q[x]/(f) \rightarrow F_q[x]/(f_1) \times \dots \times F_q[x]/(f_r),$$

$$h \pmod f \mapsto (h \pmod{f_1}, \dots, h \pmod{f_r}).$$

Let us write $R = F_q[x]/(f)$, and $R_i = F_q[x]/(f_i)$ for $1 \leq i \leq r$. Then R_i is a field with q^d elements and so contains F_q

$$F_q \subseteq F_q[x]/(f_i) = R_i \cong F_{q^d} \quad \text{for} \quad 1 \leq i \leq r.$$

Now f_i divides $h \in F_q[x]$ if and only if $h \equiv 0 \pmod{f_i}$, that is, if and only if the i th component of $\chi(h \pmod f)$ is zero. Thus if $h \in F_q[x]$ is such that $(h \pmod{f_1}, \dots, h \pmod{f_r})$ has some zero components and some nonzero components, i.e. $h \pmod f$ is a nonzero zerodivisor in R , then $\gcd(h, f)$ is a nontrivial factor of f , and we call h a "splitting polynomial". Therefore, we look for polynomials with this property.

Now assume q to be odd (the algorithm can be generalized to characteristic 2 fields). We take $m = (q^d - 1)/2$ and an r -tuple (h_1, \dots, h_r) with each $h_i \in R_i^\times = F_{q^d}^\times = F_{q^d} \setminus \{0\}$. In $F_{q^d}^\times$, half of the values are quadratic residues and the other half are quadratic nonresidues. Thus, $h_i^m = \pm 1$, with the same probability for both values when h_i is chosen randomly. Now, choose at random (uniformly) a polynomial $h \in F_q[x]$, with $\deg h < n$, and let us assume that $\gcd(h, f) = 1$ (otherwise we have already found a partial factorization). The components (h_1, \dots, h_r) of its image under the Chinese remainder isomorphism are independently and uniformly distributed random elements in $R_i^\times = F_{q^d}^\times$. Since $h_i^m = 1$ with probability $\frac{1}{2}$, the probability that $\gcd(h^m - 1, f)$ is not a proper factor of f , i.e. all the components in $(h_1^m - 1, \dots, h_r^m - 1)$ are equal, is $2 \cdot 2^{-r} = 2^{-r+1} \leq \frac{1}{2}$. Running the algorithm l times ensures a probability of failure at most 2^{-l} . Producing factorization $f = g_1 g_2$ we can repeat it for g_1 (or for g_2 if $\deg(g_2) < \deg(g_1)$). The process is interrupted when $\deg(g)$ is equal to n .

ALGORITHM:

Input: Polynomial $F(x) = P(x^p - x - \delta) \in F_q[x]$ of degree $m = np$.

Output: Monic irreducible factor of $F(x)$ of degree n .

```

1: while  $\deg(F) \neq n$ , do
2:   Choose  $h \in F_q[x]$  with  $\deg(h) < \deg(F)$  at random;
3:    $g = \gcd(h, F)$ 
4:   if  $g = 1$ , then  $g = h^{(q^n - 1)/2} - 1 \pmod{F}$ 
5:     if  $\gcd(g, F) \neq 1$ , then  $g_1 = \gcd(g, F)$ ,  $g_2 = \frac{F}{\gcd(g, F)}$ 
6:
7:     
$$F = \min_{\deg} \{g_1, g_2\};$$

8:   endif;
9:   else:  $g_2 = \frac{F}{g}$ ,  $g_1 = g$ ;
10:   $F = \min_{\deg} \{g_1, g_2\}$ 
11: endif;
12: endwhile

```

For making the proposed algorithm more understandable, we will compare between ours and that of Cantor-Zassenhaus algorithm. Using Cantor-Zassenhaus algorithm we can split the polynomial into two proper factors. The remaining thing to do is to recursively call the algorithm on every splitting polynomial unless it is already irreducible. Using our algorithm we will also be able to split the polynomial into two proper factors. After that we are recursively call our algorithm only for one splitted polynomial, unless find one polynomial of degree n .

Theoretical computations show that the cost of the proposed algorithm for factoring polynomial $P(x^p - ax - \delta)$ of degree np , where n is a degree of factors, is $O((n \log q + \log n))M(n) \log p$ operations in F_q .

Acknowledgements

This study was supported by the grant ('YSSP-13-22') of the National Foundation of Science and Advanced Technologies (RA) and Young Science Support Program(RA).

Bibliography

- [Cantor, 1981] D. Cantor, H. Zassenhaus. A New Algorithm for Factoring Polynomials over Finite Fields. Mathematics of Computation, Vol. 36, No. 154, April, 1981, 587–592.
- [Cao, 2012] X. Cao, L. Hu. On the reducibility of some composite polynomials over finite fields. Des. Codes Cryptogr, 64, 2012, pp. 229-239.
- [Cohen, 1992] S.D. Cohen. The explicit construction of irreducible polynomials finite fields. Des. Codes Cryptogr, 2, 1992, pp. 169-174.
- [Gathen, 2001] J. V. Z. Gathen, D. Panario. Factoring Polynomials over Finite Fields: A Survey, Academic Press, 2001.
- [Kyuregh, 2011] M. Kyureghyan, G. Kyureghyan. Irreducible Compositions of Polynomials over Finite Fields. Des. Codes Cryptogr, 61(3), 2011, pp. 301-314.
- [Lidl, 1987] R. Lidl, H. Niederreiter. Finite Fields, Cambridge University Press, 1987.

[Meyn, 1990] H.Meyn. On the construction of irreducible self-reciprocal polynomials over finite fields. Appl. Algebra Eng. Commun. Comput, 1 1990, pp. 43-53.

[Varshamov, 1973] R. R. Varshamov, Operation substitution in a Galois field and their applications, Soviet Math. Dokl., 211, 1973, 768 – 771.

[Varshamov, 1984] R. R. Varshamov, A general method of synthesizing irreducible polynomials over Galois fields, Soviet Math. Dokl., 29, 1984, 334 – 336.

Authors' Information



Sergey Abrahamyan P.O. Box: 0014, P. Sevak street 1, Yerevan 0014, Armenia;

e-mail: serj.abrahamyan@gmail.com

Major Fields of Scientific Research: Cryptography, Coding Theory



Knarik Kyuregyan P.O. Box: 0014, P. Sevak street 1, Yerevan 0014, Armenia;

e-mail: knarikyuregyan@gmail.com

Major Fields of Scientific Research: Cryptography, Coding Theory

PHYSICAL PHENOMENON OF STATISTICAL STABILITY

Igor Gorban

Abstract: *The article presents new monograph dedicated to the researching of physical phenomenon of statistical stability and exposure of basics of physical-mathematical theory of hyper-random phenomena, the latter describing physical events, variables and processes with consideration of violation of statistical stability. In contrast to two previous author's monograph devoted to the same subject in which the main attention was paid to the mathematical aspects of the theory of hyper-random phenomena, in the new book the accent is made on physical headwords. The book is oriented on scientists, engineers, and post-graduate students researching in statistical laws of natural physical phenomena as well as developing and using statistical methods for high-precision measuring, prediction and signal processing on long observation intervals. It may also be useful for high-level courses for university students majoring in physical, engineering, and mathematical fields.*

Keywords: *phenomenon of statistical stability, theory of hyper-random phenomena, physical process, violation of convergence.*

ACM Classification Keywords: *G.3 Probability and Statistics*

Introduction

In 2014 it has been published a new monograph [Gorban, 2014 (1)] dedicated to the research of physical phenomenon of statistical stability and exposure of basics of physical-mathematical theory of hyper-random phenomena, the latter describing physical events, variables and processes with consideration of violation of the statistical stability. In contrast to previous monographs [Gorban, 2007 (1), 2011 (1)] mainly devoted to mathematical aspects of the theory of hyper-random phenomena in the new book the accent is made on physical questions.

The aim of the current article is presentation of the new monograph. It is written on the base of the original author's researches published in Russian and English in different scientific issues from 2005 to 2014 [Gorban, 2005-2014].

Questions regarded in the monograph

The phenomenon of statistical stability

One of the surprising physical phenomena is the phenomenon of statistical stability, consists of the *stability of statistics* (that are the functions of the sample), in particular a frequency of mass events, averages, etc. This phenomenon is widespread and therefore *it can be regarded as fundamental natural phenomena*.

The statistical stability phenomenon the first noticed in 1669 John Graunt (who was a cloth merchant) [Graunt, 1939]. Researches of this phenomenon led to the development of the probability theory, widely used now in different areas of science and technology.

Axiomatization of the probability theory

Prior to the beginning of XX century probability theory was regarded as a *physical theory*, which describes the phenomenon of statistical stability.

At the beginning of the last century, the problem of probability theory axiomatization was raised. David Hilbert formulated this problem as a part of axiomatization problem of physics [Hilbert's problems, 1969].

Many famous scientists have made great efforts to solve the problem. Different approaches have been proposed. At present, recognized approach is the set-theoretic one proposed by A.N. Kolmogorov, raised even to the rank of the international standard ISO [International standard, 2006].

The concept of random phenomenon

In accordance with the Kolmogorov's approach a *random event* is described by using the probabilistic space defined by the triad of $(\Omega, \mathfrak{S}, P)$, where Ω is the space of elementary events $\omega \in \Omega$, \mathfrak{S} is the Borel field (σ -algebra of subsets of events) and P is the probability measure (probability) of subsets of events.

A *random variable* is regarded as a measurable function defined on the space Ω of the elementary random events ω , and a *random function* — as a function of the independent argument, the value of which is a random variable when its argument is fixed.

Under a *random phenomenon* is understood the mathematical object (a random event, a variable or a function), which is exhaustively characterized by certain concrete probability distribution law.

In the book, a *phenomenon* or a *mathematical model*, not being described by a concrete distribution law does not consider as a random one. This is an extremely important position that must be taken into account.

The probability concept

In probability theory a concept of probability event is a key one. Draw attention, in Kolmogorov's definition the probability has not physical interpretation.

With more visual statistical determining of probability concept (by R. von Mises [Mises, 1964]), the probability $P(A)$ of a random event A is interpreted as a limit of the event frequency $p_N(A)$ when the experiments are led under the identical statistical conditions and the experiment quantity N tends to infinity: $P(A) = \lim_{N \rightarrow \infty} p_N(A)$.

At small values N the frequency $p_N(A)$ can fluctuate greatly, but with increasing of N it gradually stabilizes and under $N \rightarrow \infty$ tends to a definite limit $P(A)$.

Physical hypotheses of the probability theory

All mathematical theories, including probability theory based on the Kolmogorov's axioms are related to the abstract mathematical concepts which *are not associated with the actual physical world*. In practice, correct application of these theories is possible if some *physical hypotheses* asserting the adequacy of description of real world objects by the relevant mathematical models are accepted.

For probability theory such *physical hypotheses* are the following ones:

- The *hypothesis of perfect statistical stability (ideal statistical predictability)* of parameters and characteristics of real physical phenomena (real events, variables, processes, and fields), signifying the *presence of convergence of any statistic to a constant value*;
- The *hypothesis of adequate describing of real physical phenomena by random (stochastic) models*.

It is assumed that the hypothesis of perfect statistical stability is valid for a lot of physical mass phenomena. In other words, the *stochastic concept of world's building is accepted*.

The hypothesis of perfect statistical stability

One of the main requests to the physical hypotheses is that *they are harmonized with the experimental data*.

For many years, the hypothesis of ideal statistical stability was not doubtful one, although some scholars (even Kolmogorov and such famous scientists as A. A. Markov [Markov, 1924, p. 67] A. V. Skorokhod [Ivanenko, Labkovsky, 1990, p. 4], E. Borel [Borel, 1956, pp. 28-29] V. N. Tutubalin [Tutubalin, 1972 (2), pp. 6-7] and others) noticed that, in the real world this hypothesis is valid only with certain reservations.

Violation of statistical stability in the real world

Experimental researches on large observation intervals of various processes of different physical nature show that the *hypothesis of perfect statistical stability is not confirmed*.

The real world is continuously changing. Changes occur at all levels, including statistical one. Statistical assessments formed at relatively small observation intervals are relatively stable. The stability is manifested in decreasing of fluctuation of statistical assessments when the number of statistical data is raised. This creates the illusion of a perfect statistical stability. However, starting from a certain critical volume, the level of fluctuations practically does not change (sometimes even grows) when the amount of the data is raised. This indicates that the statistical stability is not perfect.

Violation of statistical stability in the real world means that the *probability concept has not physical interpretation*. *The probability is a mathematical abstraction*.

Violation of statistical stability in deterministic and stochastic models

Violation of statistical stability is observed in different models, even deterministic and random ones.

A typical example is a random variable described by the Cauchy distribution. This distribution has not the moments and therefore any assessments of its moments are *statistically unstable (inconsistent)* ones.

Causes of statistical instability in the real world

Violation of statistical stability is called by a lot of reasons. They are inflow in open system matter, energy and (or) information, feeding non-equilibrium processes, various nonlinear transformations, low-frequency linear filtering of special type, etc.

It was founded that *statistical stability of the process was determined by its power spectral density*. Therefore, a broadband stationary statistically stable noise in a result of low-frequency filtration can be transformed into statistically unstable process.

Investigation of violations of statistical stability and looking for effective ways for adequate description of real world phenomena accounting these violations led to building of new *physical-mathematical theory of hyper-random-random phenomena* [Gorban, 2007 (1), 2011 (1), 2014 (1)].

The concept of hyper-random-random phenomenon

In the probability theory the basic mathematical objects (models) are random phenomena (random event, variable, and function); in the theory of hyper-random phenomena such objects are hyper-random phenomena (hyper-random event, variable, and function) representing the set of unlinked random objects regarded in complex as comprehensive whole.

A *hyper-random event* can be described by the tetrad $(\Omega, \mathfrak{S}, G, P_g)$, where Ω is a space of elementary events $\omega \in \Omega$, \mathfrak{S} is a Borel field, G is a set of conditions $g \in G$, P_g is a probability measure of subsets of events, depending on conditions g . Thus, the probability measure is defined for all subsets of events and all possible conditions $g \in G$. Mark, the measure for conditions $g \in G$ does not exist.

Using a statistical approach, hyper-random event A can be interpreted as an event, the frequency $p_N(A)$ of which *does not stabilize with rising of the number N and under $N \rightarrow \infty$ this frequency has not a limit*. So in this case, the property of statistical stability is not intrinsic for event frequency. However, the property of statistical stability may be intrinsic for other statistics, for instance ones describing *bounds of event's frequency oscillation*.

A random phenomenon is exhaustively described by the probability distribution, and a hyper-random phenomenon — by the set of conditional probability distributions.

A random variable X , for example, is completely characterized by the distribution function $F(x)$, and a hyper-random variable $X = \{X / g \in G\}$ — by the *set of conditional distribution functions $F(x / g)$, $g \in G$* .

A hyper-random variable can be presented not only by such set, but also by other characteristics and parameters, in particular by the upper $F_S(x) = \sup_{g \in G} F(x / g)$ and the lower $F_I(x) = \inf_{g \in G} F(x / g)$ bounds of the distribution function, by the central and crude moments of these bounds, by the bounds of the moments, etc.

The link of hyper-random models with others ones

A random variable can be interpreted as the hyper-random variable with coincided bounds of the distribution function: $F_S(x) = F_I(x) = F(x)$.

A *deterministic variable (constant)* can be approximately regarded as a degenerated random (or hyper-random) variable with distribution function $F(x)$ having a single jump at the point x_0 .

An *interval variable* characterized by the borders x_1, x_2 of the interval can be represented by the hyper-random variable with bounds of the distribution function $F_S(x), F_I(x)$ that have unit jumps at the points x_1 and x_2 .

Thus, a *hyper-random variable is a generalization of the concepts of deterministic, random, and interval variables*. Therefore hyper-random models can be used for modeling of different physical phenomena that have various type and degree of uncertainty.

Determinism and uncertainty

For centuries it was believed that the world is based on deterministic principles. Discovery of the phenomenon of statistical stability shook these representations. It turned out that it was essential not only determinism, but also uncertainty too.

An important form of uncertainty is *many-valuedness*. Many-valued mathematical objects are random phenomena, interval variables and functions, as well as hyper-random phenomena. In all of them there is an uncertainty, though of different types. *Uncertainty of random phenomena has a probability measure and interval variables and functions have not such measure. Hyper-random phenomena contain uncertainty of both types*.

Object and subject of investigation of the theory of hyper-random phenomena

The *study object* of the theory of hyper-random phenomena are real physical phenomena — events, quantities, processes and fields. The *study subject* is a violation of statistical stability of characteristics and parameters of real physical phenomena.

General characteristic of the theory of hyper-random phenomena

The theory of hyper-random phenomena has mathematical and physical components. The mathematical component is based on the classical Kolmogorov's axioms of probability theory, the physical one — on two hyper-random physical adequacy hypotheses:

- The *hypothesis of imperfect statistical stability of real events, quantities, processes, and fields*, and also;
- The *hypothesis of adequate description of real physical phenomena by hyper-random models*.

The assumption that these hypotheses are valid for a wide range of mass phenomena leads to *accepting of a new concept of the world's building: it builds on hyper-random principles*. Fundamental role in this concept plays imperfect statistical stability.

From the mathematic point of view the theory of hyper-random phenomena is a *branch of the probability theory*; from the physics point of view it is a *new theory* based on the new concepts of the world's building.

The law of large numbers and the central limit theorem in case of violation of statistical stability

A violation of statistical stability is reflected in the statistical properties of the physical phenomena, in particular ones described by the law of large numbers and the central limit theorem.

Investigations show that *both in the absence and in the presence of violation of statistical stability, the sample mean of a random sample tends to average of the mathematical expectations*. However, in the absence of violation of statistical stability the sampling mean converges to a certain number, and *in violation of stability it tends to infinity (plus or minus) or fluctuates within a certain range*. In general, the sample mean in limit may be a number, a random variable, interval or hyper-random variable with continuous area of uncertainty, bounded by the curves consisting of fragments of Gaussian curves.

Sample mean of a hyper-random sample converges to fixed value (number), to a set of the fixed values (numbers), fluctuates in one or more disjoint intervals or tends to infinity. At the same time the sample mean in limit can be a number, interval, multiinterval, random variable, or hyper-random variable with optional continuous area of uncertainty, bounded by the curves consisting of fragments of Gaussian curves.

Potential accuracy in case of violation of statistical stability

One of the most important questions is the potential accuracy of the measurement.

According to classical conception developed yet by Galileo Galileo the estimated physical quantity can be represented by the *single-valued deterministic volume* and the result of measurement — by the *random variable*. Measurement error has two components: *systematic* and *random* ones.

According to the probability theory when the sample size follows to infinity, the random component tends to zero and the whole error — to the systematic component. However, in practice, as is well known, it does not occur. The cause of that is in violation of statistical stability.

Within the hyper-random paradigm, the *error has hyper-random nature and describes by hyper-random variables*.

In general, it is impossible to divide the error in some components. In one of the simplest cases (when the bounds of the distribution function of hyper-random error differ only on mathematical expectations) the *error can be*

divided in a systematic, a random and an uncertain (an unpredictable) components, the latter is described by the interval value.

While the sample size follows to infinity the *hyper-random error keeps hyper-random character.*

This explains many well-known but for a long time incomprehensible facts, in particular, why the accuracy of any physical measurements is limited, why in case of a large number of experimental data the accuracy does not depend on the volume of data, etc.

How the uncertainty is formed

There are many pathways of uncertainty forming. The simplest of them is a non-linear transformation, generating many-valuedness. The averaging of the determined data in the absence of convergence leads to uncertainty too.

Efficiency of different models

Different models describe the indeterminate properties of the visual environment with various accuracy and in different ways.

Since the concept of probability has not physical interpretation, we must recognize that stochastic models describe these properties approximately. An adequate description can provide *interval and hyper-random models.*

This circumstance, however, does not mean that the stochastic model and other simple ones are useless. Not a complete correspondence the model to simulated object is important only for large sample sizes. Often the sample sizes are small. Then the description error of the real objects by stochastic and by other approximate models *is negligible.* Typically, these models are simpler than the interval and hyper-random models, and therefore in many cases *they are preferred.*

The necessity in more complex interval and hyper-random models arises when *gives evidence the restricted character of the phenomenon of statistical stability.* This usually there is in case of large observation intervals and large sample sizes.

Using scope of hyper-random models

The primary using scope of the hyper-random models is associated with the statistical processing of various physical processes (electrical, magnetic, electromagnetic, acoustic, hydroacoustic, seismic, meteorological, etc.) *of long duration, as well as high-precision measurements of physical quantities and forecasting physical processes on the basis of statistical processing of large data sets.*

Hyper-random models can also be used for simulation of physical events, variables, processes and fields, for which, due to the extreme smallness of the statistical data it is impossible to obtain well assessments of the parameters and characteristics, and it is possible only to point their borders.

The problem for formalization of physical concepts

Using non-stochastic models (in particular, interval and hyper-random models) exacerbates the underlying *problem of correct formalization of physical concepts defined now by using stochastic models,* for instance the concept of entropy.

The difficulty is that the probability has not physical interpretation, and therefore all physical concepts using the concept of probability, are actually uncertain. But this difficulty, as it turns out, can be overcome.

Mathematical analysis of divergent and many-valued functions

The theory of hyper-random phenomena touches a *little-studied field of mathematics concerning violations of convergence and many-valuedness*.

Modern mathematics is built on mathematical analysis deal with single valued sequences and functions which have single valued limits.

The development of the theory of hyper-random phenomena led to the formation of the *foundations of mathematical analysis of divergent and many-valued functions*. Limit concept is extended to the case of diverging (in usual sense) sequences and functions, and the concepts of convergence, continuity, differentiability, primitive, indefinite and definite integrals — to many-valued functions.

These questions are researched in the monograph.

The structure of the book

The monograph consists of five parts. The first part (Chapters 1-8) is devoted to discussion of phenomenon of statistical stability and developing of the methodology for researching of violations of statistical stability, in particular in case of limited amount of data. The second part (Chapters 9-13) contains a description of a set of experimental researches that examine violations of statistical stability of the various processes of different physical nature. The third part (Chapters 14-21) presents a shot description of mathematical foundations of the theory of hyper-random phenomena. The fourth part (Chapters 22-25) is devoted to the mathematical generalization of the results of the theory of hyper-random phenomena and formation of foundations of the mathematical analysis of divergent and many-valued functions. The fifth part (Chapters 26-33) contains theoretical and experimental researches of the statistical regularities in case of violation of statistical stability.

Summaries of the chapters

Chapter 1

The main manifestations of the phenomenon of statistical stability are examined, namely the statistical stability of the event's frequency and the sample averages. Mark that phenomenon of statistical stability has emergent property and it not inherent only to physical phenomena of stochastic type. The hypothesis of perfect (absolute) statistical stability, assuming the convergence of event frequencies and averages is discussed. The examples of statistically unstable processes are presented. The terms "identical statistical conditions" and "unpredictable statistical conditions" are discussed.

Chapter 2

The sixth Hilbert's problem concerning the axiomatization of physics is described. Generally recognized mathematical axiomatization principles of probability theory and mechanics are presented. New approach for solving the sixth problem based on the spreading the set of mathematical axioms by adding the physical adequacy hypotheses establishing a connection between the existing axiomatic mathematical theories and the real world is proposed. Fundamental concepts of probability theory and the theory of hyper-random phenomena are considered. The adequacy hypotheses for the probability theory and the theory of hyper-random phenomena are stated. Attention is drawn that the probability has not physical interpretation in the real world.

Chapter 3

Various conceptual views on the structure of the world from the standpoint of determinism and of uncertainty are examined. A classification of uncertainties is presented. The uniform method for presentation of the models using the distribution function is described. The classification of mathematical models is proposed.

Chapter 4

Random variables and stochastic processes, statistically unstable with respect to different statistics are examined. Various types of non-stationary processes are analyzed in respect to statistical stability.

Chapter 5

The notion of the statistical stability is formalized. The parameters of statistical instability are introduced. Measuring units of the statistical instability parameters are proposed. The concepts of the statistical stability/instability of the processes in narrow and wide senses are introduced. Statistical stability of the several models of the processes is researched.

Chapter 6

Dependence of statistical stability of the process from particularities of its temporal characteristics is researched (in particular the dependence from fluctuation parameters of expectation and correlation samples).

Chapter 7

Wiener — Khinchin transformation is examined. Attention is drawn that there are stochastic processes, which have not the correlation function typical for stationary process and the power spectral density together. Interaction between the statistical stability and power spectral density of the continues process is found. The statistical stability of the process, power spectral density of which is described by power function is investigated.

Chapter 8

Interaction between statistical stability and power spectral density of discrete process is found. The simulation results that confirm the correctness of formulas describing the dependence of the statistical instability parameters from the spectral power density of the process are presented.

Chapter 9

The results of experimental studies of the statistical stability of various physical processes are presented. It is researched the intrinsic noise of the amplifier, hydroacoustic noise of the ship, urban voltage oscillations, height and period of sea heaving, Earth's magnetic field variations, currency fluctuations. Attention is drawn to that in the small observation intervals the violations of statistical stability are not detected, but in the large observation intervals they become explicit.

Chapter 10

The results of experimental studies of statistical stability of air temperature and precipitation in Moscow and Kiev areas, as well as the wind speed in Chernobyl are presented. It is shown that all of these processes are statistically unstable. The degree of their instability is different. It is found that the temperature fluctuations much more unstable than the precipitation oscillations.

Chapter 11

The results of experimental studies on large observation intervals of the statistical stability of temperature and sound speed variations in the Pacific Ocean are presented. A statistical instability of these processes is found.

Chapter 12

The results of experimental studies on large observation intervals of the statistical stability of radiation of the astrophysical objects in the X-ray band are presented. All investigated fluctuations have been found statistically unstable. The most stable oscillation is the intensity of pulsar PSRJ 1012+5307. It is found that in the whole observation interval, its oscillations are statistically stable with respect to the average, but unstable with respect to the standard deviation.

Chapter 13

Different types of noise are researched, in particular, color noise, flicker noise, self-similar (fractal) one. The results of studies of statistical stability of various noises and processes are generalized. The causes of the violation of statistical stability are researched. It is found that statistically unstable processes can be formed in different ways: as a result of revenues from the outside in an open system of matter, energy and (or) information, as a result of nonlinear and even linear transformation.

Chapter 14

The notion of hyper-random event is introduced. To describe the hyper-random event the conditional probabilities and probability borders are used. The properties of these parameters are presented.

Chapter 15

The concept of hyper-random scalar variable is introduced. To describe it, the conditional distribution function (giving an exhaustive description of the hyper-random variable), the bounds of the distribution function, and their moments, as well as the bound of the moments are used. The properties of these characteristics and parameters are presented.

Chapter 16

The notion of hyper-random vector variable is introduced. Methods describing the scalar hyper-random variables are extended to the case of vector hyper-random variables. Properties of the characteristics and parameters of vector hyper-random variables are given.

Chapter 17

The notion of hyper-random scalar function is introduced. Various ways of its presentation are examined. To describe them, the conditional distribution functions (giving the most complete characterization of hyper-random function), the bounds of the distribution function, the density distribution of the borders, the moment borders, and borders of the moments are used.

Chapter 18

The bases of mathematical analysis of random functions are presented. The notions of the convergence of the sequence of the random variables and functions, the derivative and the integral of a random function are determined. It is introduced the concepts of the convergence of a sequence of hyper-random variables and functions, as well as of the concepts of the continuity, the differentiability and the integrability of hyper-random functions.

Chapter 19

Known for stochastic functions concepts such as stationarity and ergodicity are generalized to hyper-random functions. The spectral methods for describing of stationary hyper-random functions are regarded. The properties of stationary and ergodic hyper-random functions are presented.

Chapter 20

Different description methods of hyper-random variables and processes in a respect to appropriateness of their using in different types of transformations are analyzed. Relationships linking the characteristics and the parameters of transformed and primary hyper-random variables and processes are presented. Recommendations on using of different description ways of hyper-random variables in case of linear and nonlinear transformations, as well as of hyper-random processes in case of inertialess and inertial transformations are developed.

Chapter 21

The notion of hyper-random sample and its properties are formalized. The forming methodology of the assessments of the characteristics of the hyper-random variable is described. It is focus on that there is violation of the convergence of real assessments and that hyper-random models give adequate description of such assessments.

Chapter 22

The notion of a limit of a convergent numerical sequence is generalized to the case of divergent sequence and function. Unlike a usual limit necessarily receiving a single value, a generalized limit receives a set of values. For the divergent numerical sequence the concept of the spectrum of the limit points is introduced. The theorem on average sequence is proved.

Chapter 23

For description of divergent sequences and functions the approach based on the tool of distribution function is presented. The theorem concerned the spectrum of value frequency of sequence discharge is proved. The examples of description of the divergent functions are presented.

Chapter 24

Different variants for description of many-valued variables and functions are regarded. Using the mathematical tool developed in the theory of hyper-random phenomena, the notions of many-valued variable and many-valued function are formalized. The link between multiple meaning and the violation of the convergence is found. The concepts of the spectrum and the distribution functions of multi-valued variables and functions are introduced.

Chapter 25

For many-valued functions the concepts of the continuous function, the derivative, indefinite and definite integrals, as well as the spectrum of the principal values of the definite integral are introduced.

Chapter 26

It is established that the law of large numbers, known for the sequence of random variables, is valid both in the presence and absence of the convergence of the sample mean. In the absence of the convergence, the sample average tends to the average of expectations, fluctuating synchronously with it in a certain range. The law of large numbers is generalized to the case of the sequence of hyper-random variables. The particularities of the generalized law of large numbers are investigated.

Chapter 27

Peculiarities of the central limit theorem for a sequence of random variables in case of presence and absence of convergence of the sample mean to the fixed number are researched. The central limit theorem is generalized for a sequence of hyper-random quantities. The experimental results demonstrating the lack of convergence of sample means of real physical processes to fixed numbers are presented.

Chapter 28

Two concepts for evaluation the accuracy of the measurement are analyzed: the error concept and the uncertainty one. A number of measurement models are considered.

Chapter 29

The deterministic — hyper-random measurement model is studied. For the point hyper-random estimates the concepts of unbiased, consistent, effective, and sufficient estimates are introduced; for the interval hyper-random estimates the concepts of confidence interval and bounds of confidence probability are introduced. Theorems defining bounds of upper limits of accuracy of the point estimate and bounds of confidence interval of the interval estimate are proved. It is shown that hyper-random estimates of deterministic variables are not consistent and therefore the accuracy of any measurement is limited.

Chapter 30

The hyper-random — hyper-random measurement model is studied. The formulas describing the error of hyper-random estimate of hyper-random variable in general and particular cases are obtained. The relations that gives possibility to calculate the error of hyper-random estimate in case of indirect measurements of hyper-random variable are obtained.

Chapter 31

For point hyper-random estimates of hyper-random variables the concepts of unbiased, consistent, efficient and sufficient estimates are introduced. The theorems defining the upper limit of the accuracy of the point estimate and bounds of confidence interval of the interval estimate are proved. The fact well known from the practice that the accuracy of any actual physical measurement has a limit that can not be overcome even in case of very large data amount is explained.

Chapter 32

Different definitions of the entropy concept are analyzed. The concept of Shannon entropy for random variables is disseminated on uncertain variables that have no probability measure. The entropy concept for hyper-random and interval variables is introduced.

Chapter 33

Different ways of uncertainty formation are researched. It is found that uncertainty may arise as a result of a certain type of nonlinear transformation and in the process of the averaging the deterministic variables in the absence of convergence. It is explained why the interval, multiinterval, and hyper-random models can adequately reflect the world realities, and the random models are mathematical abstractions.

In **Appendix 1** quotes by famous scientists about the phenomenon of statistical stability are presented, in **Appendix 2** — a practical guidance for studies of statistical stability, and in **Appendix 3** — a brief history of the theory of hyper-random phenomena formation.

Conclusion

Hypothesis of absolute (ideal) statistical stability of the physical phenomena generated the classical theory of probability and mathematical statistics does not find experimental confirmation. Studies show that the *probability is the mathematical abstraction, which has not a physical interpretation. In the real world there is not absolute*

statistical stability. However, *there is a limited statistical stability, manifested in the absence of convergence of statistics (their inconsistency)*.

Looking for the adequate means for description of the real physical events, variables, processes and fields, taking into account the statistical violations of stability led to a new physical-mathematical theory of hyper-random phenomena, offering a new view on the world and new ways for its learning.

Theory of hyper-random phenomena does not cancel the achievements of the classical probability theory and mathematical statistics. It complements these achievements, extending the statements of these disciplines to the case, they are not considered: the lacking of the statistics convergence.

Limitations of statistical stability are manifested at large sample sizes and at the passage to the limit. Since the sample sizes are often small, stochastic models can provide a solution of many practical tasks with acceptable accuracy. Typically, these models are easier than hyper-random models and therefore for not very large sizes of the samples are preferred.

Hyper hyper-random models have clear advantages over stochastic and other relatively simple models in case when it is manifested the violation of phenomenon statistical stability – usually in large observation intervals and large sample sizes.

Therefore, the primary application area of hyper-random models is linked with statistical processing of various long-duration physical processes (electrical, magnetic, electromagnetic, acoustic, hydroacoustical, seismic, meteorological, etc.), as well as with high accuracy measurements of physical quantities and forecasting of physical processes on the basis of statistical processing of large data sets.

It is reasonable to use the hyper-random model for modeling of physical events, variables, processes, and fields, for which due to the extremely small volume of the statistical material it is impossible to obtain high quality estimates of the parameters and characteristics, and it is possible only to specify the boundaries in which these estimates are located.

The theory of hyper-random phenomena touches little-studied *mathematical* sphere concerning convergence violation and multivaluedness. Approaches developed in the monograph can be used for forming the *mathematical analysis of divergent and many-valued functions*. The scope of this new theory seems to be quite broad, going too far the statistics.

Limited character of statistical stability indicates the need to revise a number of statements of *physical disciplines*, in which the concepts of probability and convergence play a key role: it is primarily a *statistical mechanics, statistical physics and quantum mechanics*. Accounting violations of statistical stability may lead to new scientific results interesting for both theory and practice.

Bibliography

[Borel, 1956] Borel, E. (1956). Probability et certitude. Paris: Presses universitaires de France.

[Gorban, 2005 (1)] Gorban, I. I. (2005). Hyper-random phenomena and their description. Acoustic Bulletin, 8(1–2), 16-27. (In Russian).

[Gorban, 2005 (2)] Gorban, I. I. (2005). Description methods for hyper-random variables and functions. Acoustic Bulletin, 8(3), 24-33. (In Russian).

[Gorban, 2005 (3)] Gorban, I. I. (2005). Randomness, hyper-randomness, chaos, and uncertainty. Standardization, Certification, and Quality, (3), 41-48. (In Russian).

- [Gorban, 2006 (1)] Gorban, I. I. (2006). Hyper-random functions and their description. *Radioelectronics and Communications Systems*, 49 (1).
- [Gorban, 2006 (2)] Gorban, I. I. (2006). Mathematical description of physical phenomena in statistically unstable conditions. *Standardization, Certification, and Quality*, (6), 26-33. (In Russian).
- [Gorban, 2006 (3)] Gorban, I. I. (2006). The estimates of characteristics of hyper-random variables. *Mathematical Machines and Systems*, (1), 40-48. (In Russian).
- [Gorban, 2006 (4)] Gorban, I. I. (2006). Stationary and ergodic hyper-random functions. *Radioelectronics and Communications Systems*, 49 (6).
- [Gorban, 2006 (5)] Gorban, I. I. (2006). Point and interval estimate's methods for parameters of hyper-random variables. *Mathematical Machines and Systems*, (2), 3-14. (In Russian).
- [Gorban, 2007 (1)] Gorban, I. I. (2007). Theory of hyper-random phenomena. Kiev: IMMSP, NAS of Ukraine, 181. ISBN 978-966-02-4367-5. From http://www.immsp.kiev.ua/perspages/gorban_i_i/index.html. (In Russian).
- [Gorban, 2007 (2)] Gorban, I. I. (2007). Hyper-random phenomena: definition and description. Proceedings of XIII-th International Conference "Knowledge-Dialogue-Solution", June 18-24, 2007, Varna (Bulgaria), 1. Sofia: ITHEA, 137-147. (In Russian).
- [Gorban, 2007 (3)] Gorban, I. I. (2007). Presentation of physical phenomena by hyper-random models. *Mathematical Machines and Systems*, (1), 34-41. (In Russian).
- [Gorban, 2008 (1)] Gorban, I. I. (2008). Hyper-random phenomena: definition and description. *International Journal "Information Theories & Applications"*, 15 (3), 203-211.
- [Gorban, 2008 (2)] Gorban, I. I. (2008). Value measurement in statistically uncertain conditions. *Radioelectronics and Communications Systems*, 51 (7), 349-363.
- [Gorban, 2008 (3)] Gorban, I. I. (2008). Description of physical phenomena by hyper-random models. International Book Series "Information Science and Computing". Book 1: Algorithmic and Mathematical Foundations of the Artificial Intelligence, 135-141. (In Russian).
- [Gorban, 2008 (4)] Gorban, I. I. (2008). Hyper-random Markov models. International Book Series "Information Science and Computing". Book 7: Artificial Intelligence and Decision Making, 233-242. (In Russian).
- [Gorban, 2009 (1)] Gorban, I. I. (2009) Cognition horizon and the theory of hyper-random phenomena. *International Journal "Information Theories & Applications"*, 16 (1), 5-24.
- [Gorban, 2009 (2)] Gorban, I. I. (2009). The hypothesis of hyper-random world building and cognition possibilities. *Mathematical Machines and Systems*, (3), 44-66. (In Russian).
- [Gorban, 2009 (3)] Gorban, I. I. (2009). The law of large numbers for hyper-random sample. International Book Series "Information Science and Computing". Book 15: Knowledge-Dialogue-Solution, 251-257. (In Russian).
- [Gorban, 2009 (4)] Gorban, I. I. (2009). Description of physical phenomena by hyper-random models. Proceedings of the fifth distant conference "Decision making support systems. Theory and practices", 5-9. (In Russian).
- [Gorban, 2010 (1)] Gorban, I. I. (2010). Violation of statistical stability of the physical processes. *Mathematical Machines and Systems*, (1), 171-184. (In Russian).
- [Gorban, 2010 (2)] Gorban, I. I. (2010). Study of violations of statistical stability of currency rate. Proceedings of the fifth conference "Mathematical and simulation system modeling", 84-86. (In Russian).
- [Gorban, 2010 (3)] Gorban, I. I. (2010). Transformation of hyper-random quantities and processes. *Radioelectronics and Communications Systems*, 53(2), 59-73.
- [Gorban, 2010 (4)] Gorban, I. I. (2010). Statistical instability of magnetic field of the Earth. Proceedings of the sixth distant conference "Decision making support systems. Theory and practices", 189-192. (In Russian).
- [Gorban, 2010 (5)] Gorban, I. I. (2010). Physical-mathematical theory of hyper-random phenomena from general-system position. *Mathematical Machines and Systems*, (2), 3-9. (In Russian).

- [Gorban, 2010 (6)] Gorban, I. I. (2010). Effect of statistical instability in hydrophysics. Proceedings of the X-th All-Russian conference "Applied technologies of hydroacoustics and hydrophysics". St. Petersburg: Science, 199-201. (In Russian).
- [Gorban, 2010 (7)] Gorban, I. I. (2010). Disturbance of statistical stability. In the book "Information Models of Knowledge", 398-410.
- [Gorban, 2011 (1)] Gorban, I. I. (2011). Theory of hyper-random phenomena: physical and mathematical basis. Kiev: Naukova dumka, 318. ISBN 978-966-00-1093-2. From http://www.immsp.kiev.ua/perspages/gorban_i_i/index.html. (In Russian).
- [Gorban, 2011 (10)] Gorban, I. I. and Korovitski, Yu. G. (2011). Estimates of statistical stability of air temperature and precipitation fluctuations in Moscow and Kiev. Proceedings of the VI-fth conference "Mathematical and simulation system modeling", 23-26. (In Russian).
- [Gorban, 2011 (11)] Gorban, I. I. (2011). Researches of statistical stability of air temperature and precipitation fluctuations. Proceedings of the VII-th distant conference "Decision making support systems. Theory and practices", 175-178. (In Russian).
- [Gorban, 2011 (2)] Gorban, I. I. (2011). Disturbance of statistical stability (part II). International Journal "Information Theories & Applications", 18(4), 321-333.
- [Gorban, 2011 (3)] Gorban, I. I. (2011). Statistical instability of physical processes. Radioelectronics and Communications Systems, 54(9), 499-509.
- [Gorban, 2011 (4)] Gorban, I. I. (2011). Peculiarities of the large numbers law in conditions of disturbances of statistical stability. Radioelectronics and Communications Systems, 54(7), 373-383.
- [Gorban, 2011 (5)] Gorban, I. I. (2011). Markov's hyper-random models. Mathematical Machines and Systems, (2), 92-99. (In Russian).
- [Gorban, 2011 (6)] Gorban, I. I. (2011). Statistical stability of air temperature and precipitation fluctuations in Moscow area. Mathematical Machines and Systems, (3), 97-104. (In Russian).
- [Gorban, 2011 (7)] Gorban, I. I. (2011). The law of large numbers in conditions of violation of statistical stability. Mathematical Machines and Systems, (4), 107-115. (In Russian).
- [Gorban, 2011 (8)] Gorban, I. I., Gorban, N. I., Novotriasov, V. V. and Yaroshuk, I. O. (2011). Researches of statistical stability of temperature fluctuations in offshore area in marginal sea. Proceedings of VII All-Russian symposium "Physics of geosphere" Vladivostok, 542-547. (In Russian).
- [Gorban, 2011 (9)] Gorban, I. I. and Yaroshuk, I. O. (2011). Researches of statistical stability of temperature and sound speed in the ocean. Proceedings of the conference "CONSONANS-2011", 99-104. (In Russian).
- [Gorban, 2012 (1)] Gorban, I. I. (2012). Divergent sequences and functions. Mathematical Machines and Systems, (1), 106-118. (In Russian).
- [Gorban, 2012 (10)] Gorban, I. I. (2012). The problem of axiomatization of physico-mathematical theories. Proceedings of the conference "Modern (electronic) education MeL2012", 55-58. (In Russian).
- [Gorban, 2012 (2)] Gorban, I. I. (2012). Many-valued variables, sequences, and functions. Mathematical Machines and Systems, (3), 147-161. (In Russian).
- [Gorban, 2012 (3)] Gorban, I. I. (2012). Many-valued determine variables and functions. Processing of VII scientific-practical conference "Mathematical and simulation system's modeling", 257-260. (In Russian).
- [Gorban, 2012 (4)] Gorban, I. I. (2012). Divergent and multiple-valued sequences and functions. International Book Series "Information Science and Computing". Book 28: Problems of Computer Intellectualization, 358-373.
- [Gorban, 2012 (5)] Gorban, I. I. (2012). Statistically unstable processes: links with flicker, nonequilibrium, fractal, and color noises. Radioelectronics and Communications Systems, 55(3), 99-114.
- [Gorban, 2012 (6)] Gorban, I. I. (2012). Statistical stability of astrophysical object's radiation. Mathematical Machines and Systems, (2), 155-160. (In Russian).

- [Gorban, 2012 (7)] Gorban, I. I. (2012). Criteria and parameters of statistical instability. *Mathematical Machines and Systems*, (4), 106-114. (In Russian).
- [Gorban, 2012 (8)] Gorban, I. I. and Yaroshuk, I. O. (2012). About statistical instability of the temperature fluctuations in the Pacific ocean. *Hydroacoustical Journal*, (9), 11-17. (In Russian).
- [Gorban, 2012 (9)] Gorban, I. I. and Skorbin, A. D. (2012). Research of violation of statistical stability of wind velocity fluctuations in Chernobyl. *Proceedings of the eighth distant conference “Decision making support systems. Theory and practices”*, 39-42. (In Russian).
- [Gorban, 2013 (1)] Gorban, I. I. (2013). The sixth Hilbert’s problem: the role and the sense of physical hypothesis. *Mathematical Machines and Systems*, (1), 14-20. (In Russian).
- [Gorban, 2013 (2)] Gorban, I. I. (2013). The entropy of uncertainty. *Mathematical Machines and Systems*, (2), 105—117. (In Russian).
- [Gorban, 2013 (3)] Gorban, I. I. (2013). Classification of mathematical models. *Processing of the VIII-th scientific-practical conference “Mathematical and simulation system’s modeling”*, 370-373. (In Russian).
- [Gorban, 2013 (4)] Gorban, I. I. (2013). Formation of statistically unstable processes. *Proceedings of the IX-th distant conference “Decision making support systems. Theory and practices”*, 20-23. (In Russian).
- [Gorban, 2013 (5)] Gorban, I. I. (2013). Physico-mathematical theory of hyper-random phenomena. *Processing of the international conference “Modern informatics: problems, advances, and perspectives of development”*, 97-98. (In Russian).
- [Gorban, 2014 (1)] Gorban, I. I. (2014). Phenomenon of statistical stability. *Kiev: Naukova dumka*, 448. From http://www.immsp.kiev.ua/perspages/gorban_i_i/index.html. (In Russian).
- [Gorban, 2014 (2)] Gorban, I. I. (2014). Phenomenon of statistical stability. *Technical Physics*, 59(3), 333-340.
- [Graunt, 1939] Graunt, J. (1939). *Natural and political observations made upon the bills of mortality (1662)*. Baltimore.
- [Hilbert’s Problems, 1969] Hilbert’s Problems. (1969). P.S. Aleksandrov (Ed.). Moscow: Science. (In Russian).
- [International standard, 2006] International standard ISO 3534-1. (2006). *Statistics. Vocabulary and symbols. Part I: General statistical terms and terms used in probability*.
- [Ivanenko, 1990] Ivanenko, V. I. and Labkovsky, V. A. (1990). *Uncertainty problem in the tasks of decision making*. Kiev: Naukova Dumka. (In Russian).
- [Markov, 1924] Markov, A. A. (1924). *Calculus of probability*. Moscow. (In Russian).
- [Mises, 1964] Mises, R. (1964). *Mathematical theory of probability and statistics*. Ed. H.Geiringer. N.Y. & London: Acad. Press.
- [Tutubalin, 1972] Tutubalin, V. N. (1972). *Probability theory*. Moscow: Moscow university. (In Russian).

Author's Information



Igor Gorban – *Principal scientist of the Institute of Mathematical Machines and Systems Problem, National Academy of Sciences of Ukraine, Glushkov ave., 42, Kiev, 03187, Ukraine; e-mail: igor.gorban@yahoo.com*

Major Fields of Scientific Research: Phenomenon of Statistical Stability, Mathematical statistics, Probability theory, Physical-mathematical theory of hyper-random phenomena, Space-time signal processing in complicated dynamic conditions, Fast multichannel space-time signal processing.

DEVELOPMENT AND ANALYSIS OF THE PARALLEL ANT COLONY OPTIMIZATION ALGORITHM FOR SOLVING THE PROTEIN TERTIARY STRUCTURE PREDICTION PROBLEM

Leonid Hulianytskyi, Vitalina Rudyk

Abstract: *Parallel ant colony optimization algorithm for solving protein tertiary structure prediction problem given its amino acid sequence is introduced. The efficiency of developed algorithm is studied and the results of computational experiment on the SCIT supercomputer clusters are discussed.*

Keywords: *combinatorial optimization, protein tertiary structure prediction, ant colony optimization, parallel algorithms, SCIT supercomputer.*

Introduction

Applied science researches often result in the development of models that are convenient to be analyzed with some mathematical methods. Thus at the interfaces between math and biology the branch of computational biology appeared. It includes the set of mathematical models and methods for solving the problems that arise in biology, genetics, pharmacology, medicine. A wide range of statistical methods, data analysis, combinatorial optimization methods are used for that. Combinatorial optimization methods take on special significance in genetics inasmuch as DNA and RNA molecules are encoded as a sequence of genes and the optimization problems on strings arise. They naturally lead to combinatorial optimization problems.

One of computational biology problems is examined, namely the protein tertiary structure prediction problem based on Dill's model [Dill et al, 1995].

Protein Tertiary Structure Prediction Problem

The aim of protein structure prediction is to construct the three-dimensional shape of the molecule (tertiary structure) given its amino acid composition (primary protein structure). Dill's model describes a tertiary structure using some discrete lattice, amino acids that form a protein are placed in its nodes. In order the molecule to remain connected amino acids that are consequent in the primary sequence are placed in the neighboring lattice nodes. All amino acids are labeled as hydrophobic or polar depending on their physical properties. Between closely set hydrophobic amino acids hydrophobic contacts appear. Every structure in a lattice has some nonpositive energy that is equal to the number of hydrophobic contacts in it. It is believed the molecule takes the shape where the minimum of its energy is achieved. Formal statement of this problem appears to be NP-hard combinatorial optimization problem [Berger & Leighton, 1998].

Parallel Ant Colony Optimization Algorithm for Solving the Problem

Ant colony optimization method (ACO) is a approximate scheme developed for solving combinatorial optimization problems that simulates the process of finding the shortest ways to the food by the colony of ants [Dorigo & Stützle, 2004]. The communication between individuals is performed via pheromone trails that contribute to the selection of optimal moving directions. The complexity of the aroused combinatorial optimization problem causes the importance of parallel combinatorial algorithms research and development.

ACO-based algorithms are developed in [Shmygelska & Hoos, 2005; Chu et al, 2005; Fidanova & Lirkov, 2008]. Parallel ACO algorithm we present differs from the ones mentioned above. In particular another pheromone update procedures, heuristic estimations, structure encoding (q-encoding introduced in [Гуляницкий & Рудык, 2013]) are used. The diagram of the algorithm is presented on img.1.

The algorithm is designed to be used on multiprocessor computer systems. One host processor is singled out; all others are treated as subordinated, pairwise interactions are made between host and every subordinated processor. The differentiation of the processors into host and subordinated is conventional, the only requirement to the network architecture is the ability to pass the messages from one of the processors to others and back in a short time. The number of processors involved in calculations is equal to the number of the agents in ant population increased by one.

The first step of the algorithm is the initialization of pheromone trails matrix Φ that is performed on the host processor. The number of rows in Φ coincides with the number of code elements in the lattice, the number of columns is equal to the length of the given protein code. Initially every matrix element is set to some value $\mu_0 > 0$.

The record (in the sense of minimal energy value) structure among the ones generated during the procedure is saved, it is designated as $fold_{rec}$.

After the initialization the iterative process is performed. On every iteration the population (the number of agents in population is the parameter of the algorithm) of new structures is generated based on information that is stored in pheromone matrix. After that every structure modifies the pheromone matrix by adding new pheromone trails that intensify the paths that lead to low-energy structures. The creation of the structure starts from the first element and is processed consecutively. The next amino acid position is chosen from the neighboring free (not occupied) ones with the probability

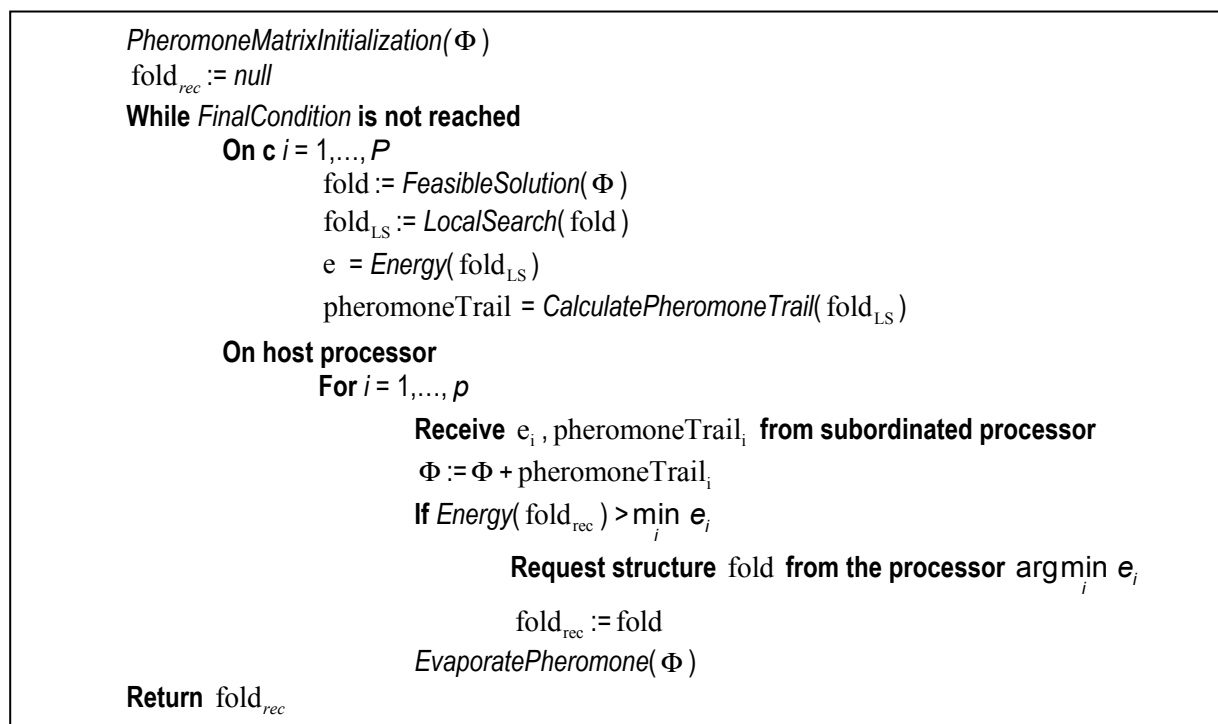
$$P_{i,d} = \frac{[\tau_{i,d}]^\alpha [\eta_{i,d}]^\beta}{\sum_{l \in D} [\tau_{i,l}]^\alpha [\eta_{i,l}]^\beta},$$

where D is a set of q-encoding code elements, that reflects the directions to free neighbors of the current (latest occupied) node in the lattice, $\tau_{i,d}$ is the pheromone matrix element that corresponds to the direction d for amino acid number i with regard to amino acid number $i - 1$, $\eta_{i,d}$ is a heuristic estimation that depends on the constructed part of the structure, α and β are the parameters of the algorithm that describe the degree of influence of pheromone and heuristic information on the structure composition.

If after some step the construction process reached a deadlock, i.e. we have a situation when all neighbor nodes are occupied, a one-step rollback is performed and deadlock direction is saved to tabu-list not to be repeated again. The solution construction is performed on subordinated processors that gets pheromone matrix as an input. To reach the effectiveness increase every of the subordinated processors also performs a local search procedure.

Next a pheromone trail *pheromoneTrail* is evaluated on every subordinate processor. It is represented as a sequence of pairs (matrix row index – the amount of pheromone) $(d, \Delta_{i,d,c})$, where d is the direction to the i -th amino acid in the structure c , and $\Delta_{i,d,c}$ is the relative quality of the structure c taking into account direction d .

To calculate $\Delta_{i,d,c}$ two ways are proposed. The first one treats $\Delta_{i,d,c}$ as the relative quality of structure that depends on its energy. The second one is developed to take into account the fact that it is required to strengthen only the trails that influence the energy of constructed molecule. The "strength" $\Delta_{i,d,c}$ of a certain direction d in structure c is defined as the number of connections influenced by this direction. Such scheme provides taking into account the suitability of this or that part of a molecule structure.



Img. 1 Parallel algorithm diagram

The constructed sequence of pairs together with the value of the best generated solution energy is then passed to the host processor that analyses and aggregates received information: adds the mentioned amount of pheromone to corresponding matrix elements ($\tau_{i,d} = \tau_{i,d} + \Delta_{i,d,c}^\gamma$, γ is the parameter of the algorithm),

compares the received energy value with the saved record one. If lower energy value is found additional request to corresponding processor is performed to get the corresponding structure.

To keep the relevance of pheromone matrix pheromone evaporation procedure is used. It simulates the process of pheromone trails evaporation in nature. Every matrix element is multiplied by some positive parameter ρ that is less than one: $\tau_{i,d} = (1-\rho)\tau_{i,d}$, ρ is an evaporation coefficient, it characterizes the fraction of information gathered on previous steps that is kept.

Pheromone evaporation is also performed on the host processor, after that if final condition is not reached the next iteration is executed. In our implementation the process stops if the record solution $fold_{rec}$ has not been changed during a certain number of iterations.

Parallel Algorithm Efficiency Study

Parallelization of the computations does not always lead to computation time decreasing. So it is important to make an analysis of suggested procedure running time. Let's introduce the following designations:

- p - the number of subordinated processors (that is equal to the number of agents in sequential algorithm);
- t_{ini} - estimated initialization time;
- $t_{slavelter}$ - estimated time consumed by host processor to perform one iteration (it includes solution generation, local search and pheromone trails calculation);
- $t_{phUpdate}$ - estimated time consumed by host processor to update pheromone matrix for one structure;
- t_{vapor} - estimated time consumed on pheromone matrix modification (evaporation);
- t_{exch} - estimated time consumed to exchange the messages between the host processor and one of the subordinated processors (includes pheromone matrix passing from the host processor to the subordinated one and pheromone trail passing back);
- I - the number of algorithm iterations.

Then the estimated parallel algorithm computation time is $t_{parallel} = t_{ini} + (p(t_{exch} + t_{phUpdate}) + t_{slavelter} + t_{vapor}) \times I$, while the estimated time for sequential algorithm is $t_{seq} = t_{ini} + (p(t_{phUpdate} + t_{slavelter}) + t_{vapor}) \times I$.

So the acceleration when using parallel algorithm compared to sequential one is

$$K_p = \frac{t_{ini} + (p(t_{phUpdate} + t_{slavelter}) + t_{vapor}) \times I}{t_{ini} + (p(t_{exch} + t_{phUpdate}) + t_{slavelter} + t_{vapor}) \times I} = 1 + \frac{((p-1)t_{slavelter} - pt_{exch}) \times I}{t_{ini} + (p(t_{exch} + t_{phUpdate}) + t_{slavelter} + t_{vapor}) \times I}$$

If the number of iterations I is big enough the initialization time t_{ini} can be ignored, then

$$k_p \approx 1 + \frac{(p-1)t_{slavelter} - pt_{exch}}{p(t_{exch} + t_{phUpdate}) + t_{slavelter} + t_{vapour}}.$$

For further analysis let's designate the number of neighbor nodes in the lattice under study as n , and the number of amino acids in the given protein as m . Let's estimate the orders of computation time depending on the values of n and m .

For a start let's estimate the message exchange time. The host processor sends to subordinated processor pheromone matrix consisting of $m \times n$ elements. By turn the subordinated processor sends the results of its computations in a format of sequence of pairs (element index – the amount of pheromone) – $m \times 2$ elements in total. So the size of transmitted messages is $O(m \times n + m \times 2) = O(m \times n)$. Supposing the message transition time is linearly dependant on its size we have $t_{exch} = O(m \times n)$.

During the evaporation step every of $m \times n$ pheromone matrix elements is updated, so $t_{vapour} = O(m \times n)$.

The time consumed to update pheromone matrix by one agent is $t_{phUpdate} = O(m)$, as one element in every row is modified (and the number of rows is m).

It remains to analyze time $t_{slavelter}$. It includes two components – time consumed to generate a feasible solution given the pheromone matrix and pheromone trails calculation time given the molecule structure. Let's start from the second one. The procedure that calculates the amount of pheromone looks through all hydrophobic contacts in the structure (their number is linearly dependent on its length m) and for every such contact some amount of pheromone is added for all the elements between those that form a contact (not more than m elements). So the complexity of the procedure is $O(m^2)$.

Let's estimate the time consumed to generate a feasible solution. To remind, a structure is constructed by sequential supplementing the elements, if at some step supplementing is not possible a rollback is performed. In the worst case the rollback procedure can turn into the brute force in some region; it means the worst-case complexity is $O(e^m)$. In practice such situations are rare in three-dimensional lattices, but it is worth pointing out that the problem is critical in two-dimension lattices as the deadlock formations are more probable. Theoretical analysis of the average computational complexity turns into determination the number of self-avoiding paths in given lattice; this problem is supposed to be *NP*-hard [Liśkiewicz et al, 2003]. The computation experiment analysis shows that the procedure of solution generation is the most time-consuming, especially on the last algorithm iterations when the pheromone matrix becomes inhomogeneous.

Let's add up the estimations:

$$k_p \approx 1 + \frac{(p-1)O(e^m) - pO(m \times n)}{p(O(m \times n) + O(m)) + O(e^m) + O(m \times n)}.$$

Two conclusions can be carried out of it. First of all if m is big enough the nominator is greater than zero, so the acceleration when using parallel algorithm compared to sequential one is greater than one. Secondly with the growth of m k_p is growing too and converges to p , that proves the practicability of using multiprocessor computer systems for the high-dimension problems.

Conclusions and Line of Further Investigations

To solve the protein tertiary structure prediction problem using multiprocessor computer systems a parallel ACO-based algorithm was developed and implemented. Acceleration estimations for using parallel algorithm compared to sequential one were calculated.

Line of further investigations is comparison of computed theoretical results with the real ones derived from computational experiment results.

Literature

- [Berger & Leighton, 1998] B. Berger, T. Leighton. Protein folding in the hydrophobic-hydrophilic (HP) model is NP-complete // Journal of Computational Biology, 1998, 5(1), pp. 27-40.
- [Chu et al, 2005] D. Chu, M. Till, A. Zomaya. Parallel Ant Colony Optimization for 3D Protein Structure Prediction using the HP Lattice Model, 19th IEEE International Parallel and Distributed Processing Symposium (IPDPS'05), 2005, 7, pp.193-200.
- [Dill et al, 1995] K. Dill, S.Bromberg, K. Yue, K. M. Fiebig, D. Yee, P. Thomas, H. Chan. Principles of protein folding - a perspective from simple exact models // Protein Science, 1995, 4, pp. 561– 602.
- [Dorigo & Stützle, 2004] M. Dorigo, T. Stützle. Ant Colony Optimization, Cambridge: MIT Press, MA, 2004, 348 p.
- [Fidanova & Lirkov, 2008] S. Fidanova, I. Lirkov. Ant Colony System Approach for Protein Folding, Int. Conf. Multiconference on Computer Science and Information Technology, 2008, pp. 887–891.
- [Liśkiewicz et al, 2003] M. Liśkiewicz, M. Ogihara, S. Todac. The complexity of counting self-avoiding walks in subgraphs of two-dimensional grids and hyper cubes // Theoretical Computer Science, 2003, 304, pp. 129-156.
- [Shmygelska & Hoos, 2005] A. Shmygelska, H. Hoos. An ant colony optimization algorithm for the 2D and 3D hydrophobic polar protein folding problem // BMC Bioinformatics, 2005, 6(30), pp. 30–52
- [Гуляницький & Рудык, 2013] Л. Гуляницький, В. Рудык. Проблема предсказания структуры протеина: формализация с использованием кватернионов // Кибернетика и системный анализ, 2013, 4, с.130-137.

Authors Information



Leonid Hulianytskyi – PhD, head of the department in V.M Glushkov Institute of Cybernetics of National Academy of Sciences of Ukraine, 40 Glushkova ave., Kyiv, Ukraine, 03680; e-mail: leonhul.icyb@gmail.com

Major fields of scientific research: combinatorial optimization; decision making; mathematical modeling and applications.



Vitalina Rudyk – junior research assistant in V.M Glushkov Institute of Cybernetics of National Academy of Sciences of Ukraine, 40 Glushkova ave., Kyiv, Ukraine; e-mail: vitalina.rudyk@gmail.com

Major fields of scientific research: computational biology, combinatorial optimization, protein structure prediction.

TABLE OF CONTENTS OF IJ ITA VOL.21, NO.:1, 2014

| | |
|---|----|
| Analyzing the Collective Intelligence Application Software "Wisdom Professional" For Advertising in (Social) Media, Case Study: Coca-Cola | |
| Elham Fayezioghani, Koen Vanhoof..... | 3 |
| A New Method for the Binary Encoding and Hardware Implementation of Metabolic Pahtways | |
| C. Recio Rincon, P. Cordero, J. Castellanos, R. Lahoz-Beltra..... | 21 |
| Advent of Cloud Computing Technologies in Health Informatics | |
| Omer K. Jasim, Safia Abbas, El-Sayed M. El-Horbaty, Abdel-Badeeh M. Salem..... | 31 |
| Matrix "Feature Vectors" in Grouping Information Problem: Linear Discrimination | |
| Volodymyr Donchenko, Fedir Skotarenko..... | 40 |
| Standardization of Geometrical Characteristics in Gesture Recognition | |
| Andrew Golik..... | 48 |
| Fuzzy Neural Networks for Evaluating the Creditworthiness of the Borrowers | |
| Natalia Shovgun..... | 54 |
| Problem and Mathematical Models for Rescue Technics Acquisition | |
| Vitaliy Snytyuk, Pavlo Kucher..... | 60 |
| Model for Astronomical Dating of the <i>Chronicle</i> of Hydatius: Results for the Interval (600-1000) | |
| Jordan Tabov..... | 65 |
| Essay on Order | |
| Karl Javorszky..... | 76 |
| Estimation of Peak Sustainable Power Consumption for Sequential CMOS Circuits | |
| Liudmila Cheremisinova, Arkadij Zakrevskij | 85 |
| Памяти член-корреспондента НАН Беларуси Аркадия Дмитриевича Закревского..... | 94 |

TABLE OF CONTENTS OF IJ ITA VOL.21, NO.:2, 2014

| | |
|--|-----|
| Programming of Agent-Based Systems | |
| Dmitry Cheremisinov, Liudmila Cheremisinova..... | 103 |
| A Language Using Quantifiers for Description of Assertions about Some Number Total Pascal Functions | |
| Nikolay Kosovskii..... | 120 |
| Worlddyn as the Tool for Study of World Dynamics with Forrester's Model: Theory, Algorithms, Experiments | |
| Olga Proncheva..... | 126 |
| Comparison of Different Wavelet Bases in the Case of Wavelets Expansions of Random Processes | |
| Olga Polosmak..... | 142 |

| | |
|---|-----|
| Wireless Data Transmission Options in Rotary In-Drilling Alignment(R-Ida) Setups for Multilateral Oil Drilling Applications | |
| Zhenhua Wang, Tao Li, Myles McDougall, Dan McCormack, Martin P. Mintchev | 154 |
| Information Technology of Processing Information of the Customs Control | |
| Borys Moroz, Sergii Konovalenko | 162 |
| Geometric Approach for Gaussian-Kernel Bolstered Error Estimation for Linear Classification in Computational Biology | |
| Arsen Arakelyan, Lilit Nerisyan, Aram Gevorgyan, Anna Boyajyan | 170 |
| The Management of Patient Information in Polish Health Care System | |
| Anna Sołtysik-Piorunkiewicz | 182 |
| Памяти Виктора Поликарповича Гладуна | 195 |

TABLE OF CONTENTS OF IJ ITA VOL.21, NO.:3, 2014

| | |
|--|-----|
| Формирование множества связанных концептов для автоматического синтеза онтологий | |
| Лариса Чалая, Антон Чижевский..... | 203 |
| Выбор вычислительного комплекса по интегральному показателю перспективности | |
| Алексей Петровский, Василий Лобанов, Алла Заболева-Зотова, Татьяна Шитова | 213 |
| Исследование критериев оптимизации в муравьином алгоритме решения задачи коммивояжера | |
| Юрий Зайченко, Николай Мурга..... | 224 |
| Анализ финансового состояния и оценка кредитоспособности заемщиков – юридических лиц в условиях неопределенности | |
| Юрий Зайченко, Ови Нафас Агаи аг Гамиш..... | 241 |
| Алгоритм решения многокритериальных задач лексикографической оптимизации с выпуклыми функциями критериев | |
| Наталия Семенова, Мария Ломага, Виктор Семенов | 254 |
| Оценка ожидаемой эффективности интервальных альтернатив | |
| Михаил Стернин, Геннадий Шепелев | 263 |
| Границы применимости аксиоматического подхода к сужению множества парето | |
| Владимир Ногин | 275 |
| Программный комплекс региональных моделей потребления электроэнергии в российской федерации | |
| Галина Старкова, Наталья Фролова..... | 283 |
| Динамический межотраслевой баланс с опережающим аргументом (дискретный случай) | |
| Игорь Ляшенко, Юрий Тадеев..... | 294 |

TABLE OF CONTENTS OF IJ ITA VOL.21, NO.:4, 2014

| | |
|--|-----|
| Intellectual Information Support of Branch Enterprise Executives' Decision Making Processes | |
| Aleksey Voloshyn, Bogdan Mysnyk, Vitaliy Snytyuk | 303 |
| A Hierarchical Approach to Multicriteria Problems | |
| Albert Voronin, Yuriy Ziatdinov, Igor Varlamov | 314 |
| Software Effort Estimation Using Radial Basis Function Neural Networks | |
| Ana Maria Bautista, Angel Castellanos, Tomas San Feliu | 319 |
| MICRORAM: A Simulation Model of a Colony of Bacteria Evolving Inside an Artificial World | |
| Daniel Thai Dam, Rafael Lahoz-Beltra | 328 |
| Universal and Determined Constructors of Multisets of Objects | |
| Dmytro Terletsykyi | 339 |
| WordArM - a System for Storing Dictionaries and Thesauruses by Natural Language Addressing | |
| Krassimira Ivanova | 362 |
| An Algorithm for Factoring Composite Polynomial $P(x^p - x - \delta)$ | |
| Sergey Abrahamyan, Knarik Kyureghyan | 371 |
| Physical Phenomenon of Statistical Stability | |
| Igor Gorban | 377 |
| Development and Analysis of the Parallel Ant Colony Optimization Algorithm for Solving the Protein Tertiary Structure Prediction Problem | |
| Leonid Hulianytskyi, Vitalina Rudyk | 392 |
| Table of contents of IJ ITA Vol.21, No.:1, 2014 | 398 |
| Table of contents of IJ ITA Vol.21, No.:2, 2014 | 398 |
| Table of contents of IJ ITA Vol.21, No.:3, 2014 | 399 |
| Table of contents of IJ ITA Vol.21, No.:4, 2014 | 400 |