КОМПЮТЪРНИ НАУКИ И КОМУНИКАЦИИ



Електронно списание на
Център по информатика и технически науки на
Бургаски Свободен Университет

ТОМ 2, БРОЙ 3 (2013)

ISSN: 1314-7846

Списание „Компютърни науки и комуникации”,

Главен редактор

prof. d-r Angel Toshkov, Bulgaria Burgas, 8001 SanStefano str. 62 Burgas Free University, Bulgaria

Редакционна колегия

1. prof. d-r Andon Lazarov, Burgas Free University, Bulgaria
2. prof. d-r Acad. Ivan Popchev, Bulgarian Academy of Sciences, Sofia, Bulgaria
3. prof. d-r Mihail Iliev, Technical University of Ruse, Ruse, Bulgaria
4. prof. d-r Kiril Konov, Technical University of Sofia, Research Institute of Communications, Sofia, Bulgaria, Bulgaria
5. prof. d-r Ivan Kolev, Technical university of Gabrovo, Gabrovo, Bulgaria
6. prof. d-r Vencislav Valchev, Technical University of Varna, Bulgaria
7. prof. d-r Daniela Orozova, Burgas Free University, Burgas, Bulgaria
8. prof. d-r Михаил Фёдорович Меняев, Доктор педагогических наук, профессор, Академик Международной Академии Информатизации, Москва, Россия
9. assoc.prof. d-r. Veselina Jecheva, 62 San Stefano Str., Burgas
10. assoc.prof. d-r. Evgeniya Petrova Nikolova, 62 San Stefano Str., Burgas
11. assoc. prof. d-r Marcin Paprzycki, Associate Professor in the Systems Research Institute of the Polish Academy of Sciences, Warsaw, Poland
12. d-r Yingjie Yang, School of Computer Science and Informatics Centre for Computational Intelligence (CCI) and De Montfort University Interdisciplinary Group in Intelligent Transport Systems (DIGITS)
13. Michela Tramonti, Università degli Studi "G. Marconi" Rome Area, Italy

ISSN: 1314-7846

Том 2, № 3 (2013)

Юли - Септември

Съдържание

Компютърни науки и комуникации

ЕЛЕКТРОННОТО СПИСАНИЕ НА ЦЕНТЪРА ПО ИНФОРМАТИКА И ТЕХНИЧЕСКИ НАУКИ В БУРГАСКИ СВОБОСЕН УНИВЕРСИТЕТ Е ВКЛЮЧЕНО В БАЗАТА НА ВИНИТИ – РАН PDF

Angel Toshkov 1

АНАЛИЗ НА ЗАГУБИТЕ ОТ МАКРООГЪВАНИЯ В ОПТИЧНИ ВЛАКНА PDF

Angel Toshkov 2-9

ОТ ДЖАДЖИ ДО РОБОТИ С MICROSOFT GADGETEER PDF

Dimitar Minchev 10-15

ГЕНЕТИЧНИТЕ АЛГОРИТМИ КАТО СРЕДСТВО ЗА РЕШАВАНЕ НА ОПТИМИЗАЦИОННИ ЗАДАЧИ PDF

Penka Georgieva 16-23

EVALUATION OF THE EFFECTIVENESS OF ANOMALY IDS BASED ON THE CLUSTERING ALGORITHM AND DATA MINING TECHNIQUES PDF

V. Jecheva, E. Nikolova 24-30

MECHANISMS AND MACHINERY USED IN UTILIZATION OF RENEWABLE ENERGY SOURCES PDF

Радостин Долчинков 31-42

МОДЕЛ НА ПРОЦЕСИТЕ В СКЛАД ОТ ДАННИ ЧРЕЗ СРЕДСТВАТА НА ОБОБЩЕНИТЕ МРЕЖИ PDF

Daniela Orozova 43-52

ISSN: 1314-7846

FROM GADGETS TO ROBOTS WITH MICROSOFT GADGETEER

Assist. Prof. PhD, Dimitar Minchev
Burgas Free University, Faculty of Computer Science and Engineering
E-mail: mitko@bfu.bg

Abstract: *In this publication wirelessly controlled robot based on Microsoft .NET Gadgeteer technology is presented. Socket client-server communication technology principle is used to transmit control commands from computer to the robot and also for displaying images taken from the camera mounted on it. Software for controlling the robot was developed and described.*

Keywords*: Microsoft, Gadgeteer, Embedded devices, Robots.*

ОТ ДЖАДЖИ ДО РОБОТИ С MICROSOFT GADGETEER

гл. ас. д-р Димитър Минчев
Бургаски Свободен Университет, Център по информатика и Технически науки
e-mail: mitko@bfu.bg

Абстракт: *В тази публикация е представен безжично управляем робот базиран на технологията Microsoft .NET Gadgeteer. Използва се принципа на сокет клиент-сървър комуникационна технология за подаване на управляващи команди към робота, както и за визуализиране на изображения, направени от камера монтирана на него. Програмно пбезпечаване за управление на робота е разработено и описано.*

Ключови думи*: Microsoft, Gadgeteer, Вградени устройства, Роботи.*

КАКВО Е НЕОБХОДИМО?

Препоръчително е да се започне с краткото ръководство на Simon Monk, озаглавено "Getting Started with .NET Gadgeteer" [1], което е отлична отправна точка при първи стъпки с тази нова технология. NET Gadgeteer е платформа за вградени устройства (*embedded devices*), която позволява сглобаването на различни видове джаджи на модулен принцип. Тя среща подкрепа от страна на компанията Microsoft, която предоставя Micro .NET Framework, като среда за програмиране на различните устройства.

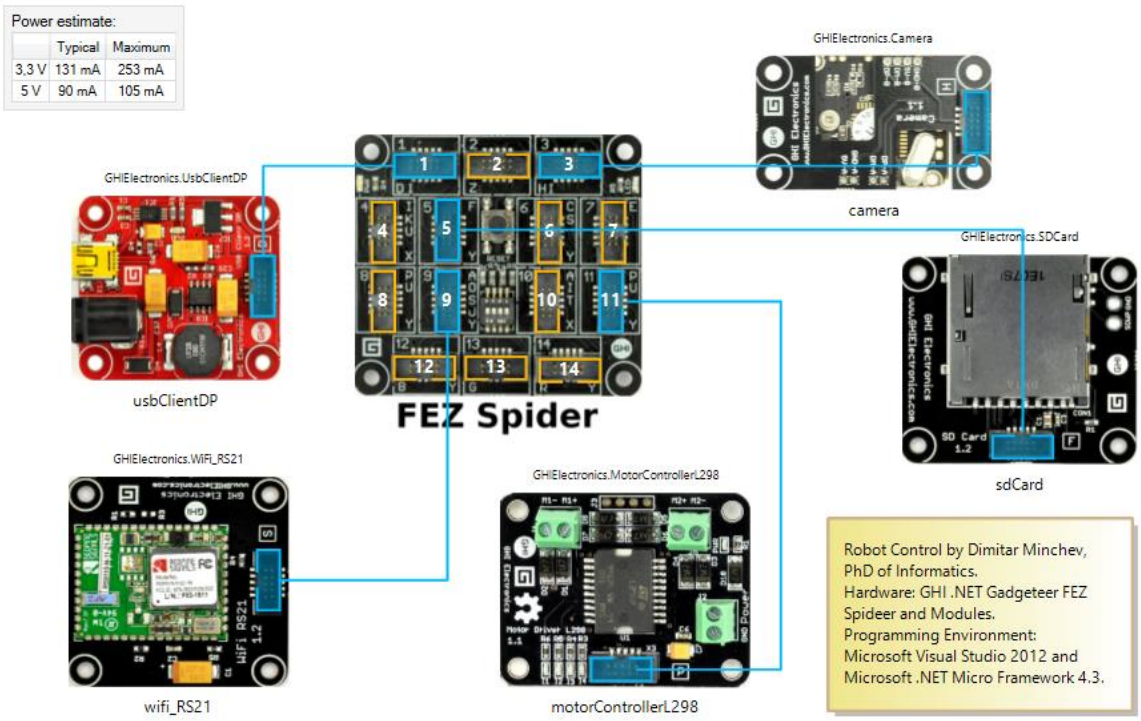
ХАРДУЕР

Примерен проект за робот управляван безжично по WiFi мрежа и стандарт 802.11g от телефон Windows Phone е създаден от Marco Minerva [2]. Авторът използва хардуер на компанията GHI Electronics [3], като е избрал за своето устройсто следните модули: Gadgeteer FEZ Spider [4], WiFi RS21 Module [5], Motor Driver L298 Module [6], Camera Module [7], OLED Display Module [8], Multicolor LED Module [9], USB ClientDP Module [10]. Програмната му реализация използва Socket Server [12] и Client Server [13] и е продължение на предишен проект за конструиране на Bluetooth управляем робот [14] на същия автор.

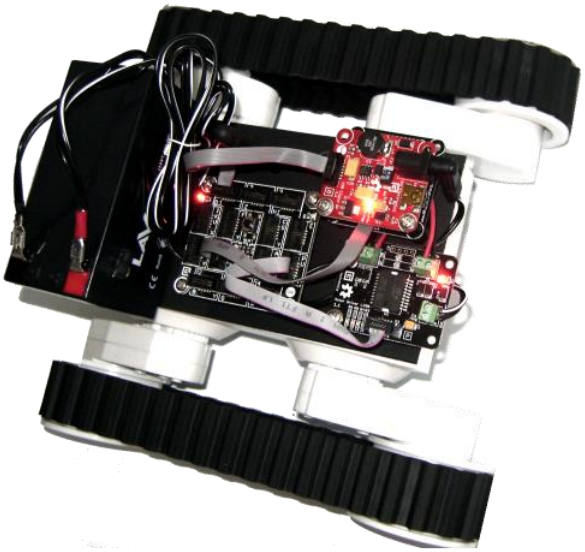
Използвайки успехите на [2] в настоящата публикация е разработен нов безжично управляем робот базиран на технологията Microsoft .NET Gadgeteer [1]. Избран е хардуер на компанията GHI Electronics [3], като в случая са използвани следните

модули: Gadgeteer FEZ Spider [4], WiFi RS21 Module [5], Motor Driver L298 Module [6], Camera Module [7], USB ClientDP Module [10] и SD Card Module [11]. Премахнати са модулите: дисплей и светлинен индикатор, но е добавен модул за Secure Digital карти.

Проектиране на хардуерната част на робота в среда Microsoft Visual Studio 2012 е представена на Фиг.1. За физическото сглобяване на робота е използвано шаси на 4 колела с гумени вериги и със задно задвижване от два електро двигателя. Автономно захранване на робота се осигурва от акумулаторна батерия със следните характеристики: 12V / 1.4Ah / 20hr. Завършен вид на хардуерната част на робота е представен на фиг.2.



Фиг.1. Проектиране на хардуерната част на робота



Фиг.2. Завършен вид на хардуерната част на робота

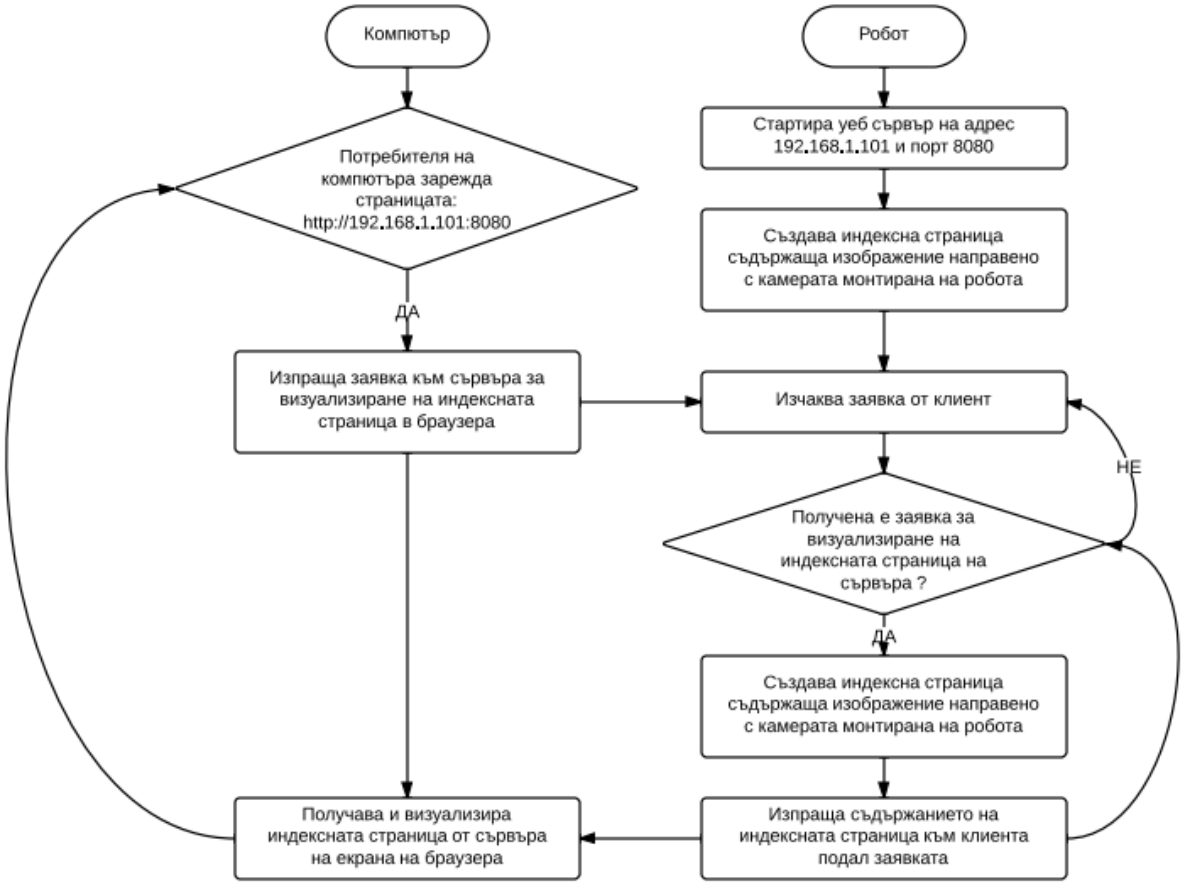
СОФТУЕР

Програмното обезпечаване за робота включва на първо място инсталирането на задължителните компоненти необходими за програмирането му, както следва: Microsoft .NET Micro Framework SDK 4.3 [15], GHI Software Package 4.2 [16] и Microsoft Visual Studio 2012 [17]. На второ и съществено място за функционирането на робота е необходимо неговото програмиране, в противен случай хардуера е само една кофа с електронни части. За осъществяването на последното е използван е езика C#.

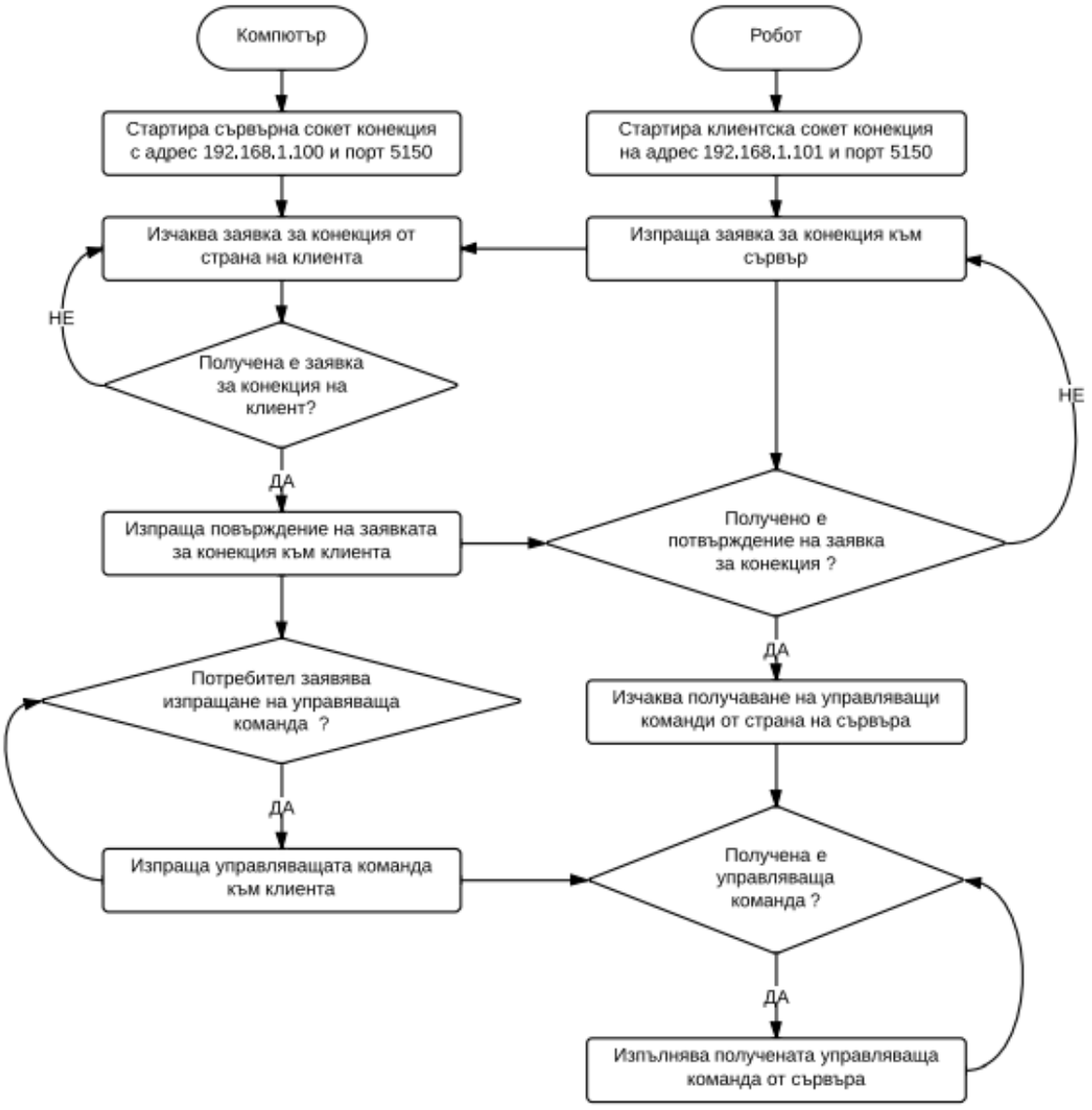
Разработени са два отделни модула, съответно: програмно осигуряване работещо на самия робот, както и програмно осигуряване за компютър, който ще извършва управление на робота. За предаване на информация между робота и компютъра е използвана комуникация тип TCP сокет клиент-сървър, конекция изградена върху безжична мрежа WiFi по стандарт 802.11g, която позволява дистанционен контрол от порядъка до 300 метра.

От една страна в робота е стартиран WEB сървър, който може да се зарежда като Интернет страница от страна на клиент (*в случая компютъра*), като за отправна точка в браузера се използва IP адрес на робота. В робота периодично се създава начална индексна страница съдържаща актуална снимка, която е направена от камерата монтирана на него. От друга страна робота е клиент, който получава комади, а компютърът е сървър, който изпраща управляващи команди.

Блокова схема на комуникация при изпращане на изображения от робота към компютъра е представена на фиг.3. Блокова схема на комуникация при изпращане на управляващи команди от компютъра към робота е представена на фиг.4.



Фиг.3. Блокова схема на комуникация при изпращане на изображения от робота към компютъра



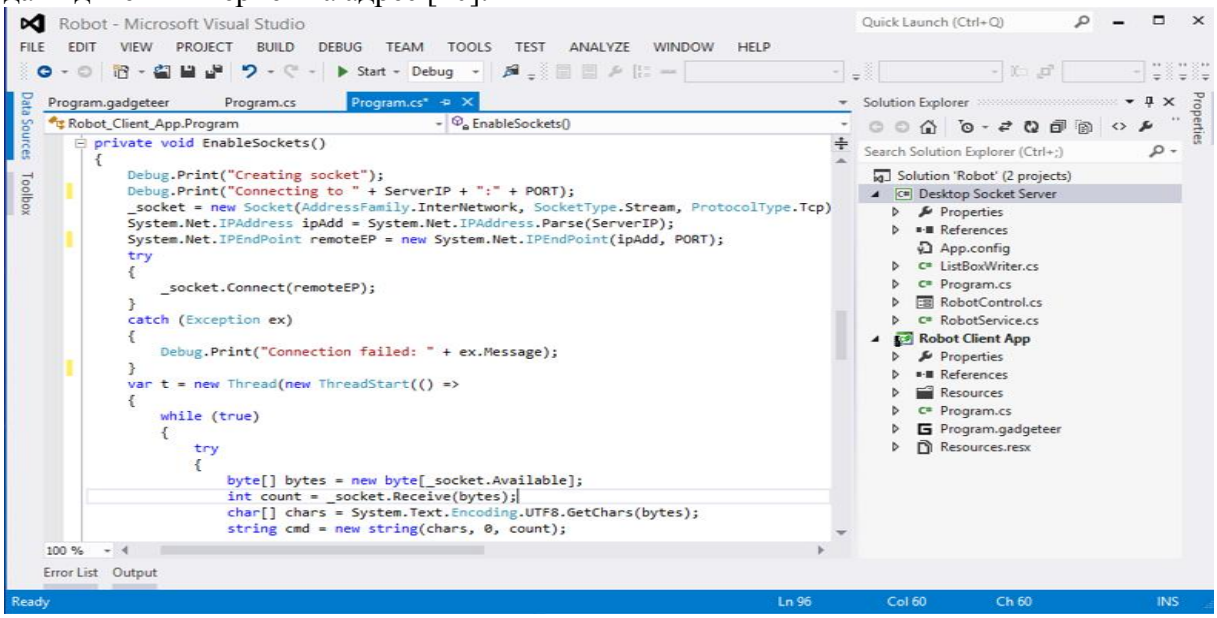
Фиг.4. Блокова схема на комуникация при изпращане на управляващи команди от компютъра към робота

В описания сценарий, както роботът, така и компютърът, имат по 2 роли. Всеки един от тях играе едновременно ролята на клиент и сървър. Това е възможно посредством създаването на 2 сокета, съответно на порт 5150 за предаване на управляващите команди от компютъра (*сървър*) към робота (*клиент*) и порт 8080 за предаване на снимков материал от камерата монтирана на робота (*сървър*) към компютъра (*клиент*).

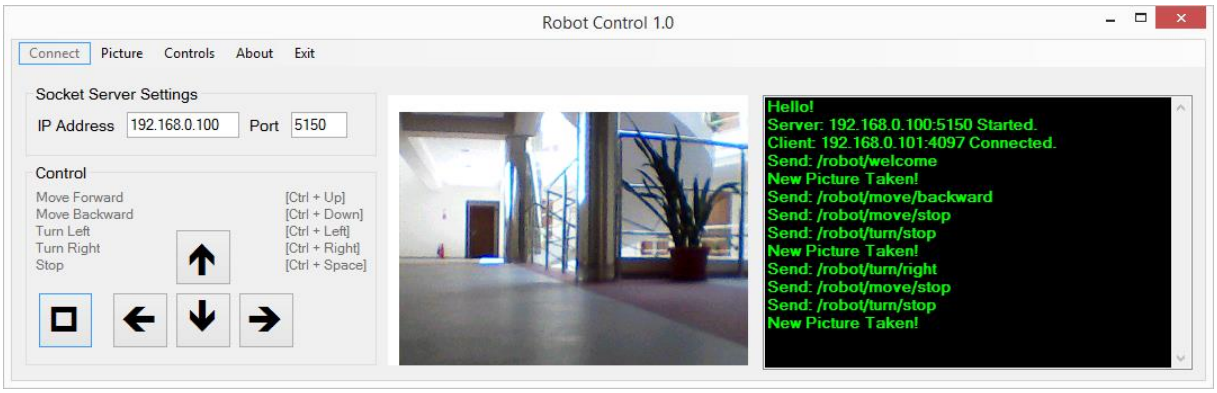
Програмиране на софтуерната част на робота в среда Microsoft Visual Studio 2012 е представено на фиг.5. Завършен вид на софтуерната част на управляващият робота компютър е представено на фиг.6. Приложението от фиг.6. е организирано, както следва: в ляво се намират бутони за управление на робота, в средата се вижда снимка направена от камерата монтирана на него, в дясно се извеждат съобщенията при успешна комуникация с робота.

Ценна теоретична и практическа информация на български език за реализация на синхронна и асинхтонна комуникация посредством сокети, заедно с демонстрирани проекти базирани на .NET технологията се дава в глава четвърта на монографията [18].

Първото представяне на успешно реализираният безжично управляем робот, който е базиран на тенологията Microsoft .NET Gadgeteer, бе осъществено на конференцията BurgasConf 2013 [19], която се проведе в Колежът по туризъм към университет „Проф. д-р Асен Златаров“ на 24 август 2013 г. Видео запис от представянето на проекта от гл. ас. д-р Димитър Минчев на тема "От джаджи до роботи с Microsoft Gadgeteer" можете да видите в Интернет на адрес [20].



Фиг.5. Програмиране на софтуерната част на робота



Фиг.6. Завършен вид на софтуерната част на управляващият робота компютър

ЗАКЛЮЧЕНИЕ

В настоящата публикация бе представен безжично управляем робот, който е базиран на тенологията Microsoft .NET Gadgeteer. Разработено е програмно осигуряване на принципа на сокет клиент-сървър технология за подаване на управляващите команди към робота, както и за визуализиране на изображенията направени от камерата монтирана на него. Настоящият робот има широк диапазон на научно-изследователски експериментални области. Продуктът може да се използва с учебна цел за провеждане на лабораторни упражнения по различни дисциплини в ЦИТН на БСУ.

ЛИТЕРАТУРА

[1] Simon Monk, Getting Started with .NET Gadgeteer, O'Reilly Media, ISBN 978-1-4493-2823-8, May 2012.
[2] Marco Minerva, Wi-Fi Gadgeteer Robot controlled by Windows Phone with image streaming, http://mikedodaro.net/2012/07/10/wi-fi-gadgeteer-robot-controlled-by-windows-phone-with-image-streaming/, Интернет, Август 2013.
[3] GHI Electronics, http://www.ghielectronics.com/ , Интернет, Август 2013.
[4] FEZ Spider Mainboard, http://www.ghielectronics.com/catalog/product/269 , Интернет, Август 2013.
[5] WiFi RS21 Module, http://www.ghielectronics.com/catalog/product/282, Интернет, Август 2013.
[6] Motor Driver L298 Module, http://www.ghielectronics.com/catalog/product/315, Интернет, Август 2013.
[7] Camera Module, http://www.ghielectronics.com/catalog/product/283, Интернет, Август 2013.
[8] OLED Display Module, http://www.ghielectronics.com/catalog/product/343, Интернет, Август 2013.
[9] Multicolor LED Module, http://www.ghielectronics.com/catalog/product/272, Интернет, Август 2013.
[10] USB ClientDP Module, http://www.ghielectronics.com/catalog/product/280, Интернет, Август 2013.
[11] SD Card Module, http://www.ghielectronics.com/catalog/product/271, Интернет, Август 2013.
[12] SocketServer, http://mikedodaro.net/2011/12/05/a-tcp-server-using-net-gadgeteer/, Интернет, Август 2013.
[13] SocketClient, http://msdn.microsoft.com/en-us/library/hh202858(VS.92).aspx, Интернет, Август 2013.
[14] Constructing a Bluetooth controlled Robot, http://mikedodaro.net/2012/05/19/constructing-a-bluetooth-controlled-robot/, Интернет, Август 2013.
[15] Microsoft .NET Micro Framework SDK 4.3 (RTM), http://netmf.codeplex.com/releases/view/81000, Август 2013.
[16] GHI Software Package 4.2, https://www.ghielectronics.com/download/ghi-software-package-v4.2-04302013, Интернет, Август 2013.
[17] Microsoft Visual Studio 2012, http://www.microsoft.com/visualstudio/eng/downloads, Август 2013.
[18] Десислава Петрова-Антонова, Проектиране и интегриране на софтуерни системи с .NET технологии, УИ "Св. Климент Охридски", ISBN 978-9-5407-3301-2, 2012.
[19] BurgasConf 2013, Lightning talks, http://burgasconf.com/, Конференция, Колеж по туризъм към университет „Проф. д-р Асен Златаров“, 24 август 2013.
[20] "От джаджи до роботи с Microsoft Gadgeteer" Димитър Минчев, http://va.ludost.net/files/burgasconf/2013/09.Lightning/01.%D0%9E%D1%82%20%D0%B4%D0%B6%D0%B0%D0%B4%D0%B6%D0%B8%20%D0%B4%D0%BE%20%D1%80%D0%BE%D0%B1%D0%BE%D1%82%D0%B8%20%D1%81%20Microsoft%20Gadgeteer%20-%20%D0%94%D0%B8%D0%BC%D0%B8%D1%82%D1%8A%D1%80%20%D0%9C%D0%B8%D0%BD%D1%87%D0%B5%D0%B2.mp4, Интернет, Септември 2013.

EVALUATION OF THE EFFECTIVENESS OF ANOMALY IDS BASED ON THE CLUSTERING ALGORITHM AND DATA MINING TECHNIQUES

V. Jecheva[*], Burgas Free University, vessi@bfu.bg
E. Nikolova[*], Burgas Free University, enikolova@bfu.bg

Abstract: *The purpose of this paper is to examine the feasibility of clustering-based approach to anomaly-based intrusion detection systems (IDS). The examined methodology includes a 2-means clustering algorithm with and without data mining techniques, i.e. classification trees. With purpose to evaluate the effectiveness of the methodology, Jaccard index was applied. Davies-Bouldin index, Dunn index and C-index were applied in order to compare the performance results of the two models.*

Keywords: *Anomaly based IDS, 2-means clustering, classification tree, Wagner-Fischer distance, Jaccard index, Davies-Bouldin index, Dunn index, C-index*

INTRODUCTION

Intrusion detection systems (IDS) play an important role in the network security systems. Their purpose is to protect the system by raising an alarm when an intrusive activity is found, as well as to start some proactive mechanisms. As new exploits and attacks appear every day and amount of audit data, which has to be processed, constantly increases, IDSs should develop in order to counteract them.

According to the intrusion detection method, IDS could be broadly divided into two categories: anomaly-based and misuse-based. Misuse-based IDS uses specifically known patterns, referred to as signatures, of unauthorized behavior to detect intrusions. However, it has low degree of accuracy in detecting unknown intrusions since it relies on signatures extracted by human experts [13]. The anomaly based detection relies on preliminarily made description of acceptable and inacceptable user behavior. The network behavior is in accordance with the predefined behavior, then it is accepted or else it triggers the event in the anomaly detection. The accepted network behavior is prepared or learned by the specifications of the network administrators [10]. The major advantage of this approach is the ability to detect novel attacks, or deviations of existing attacks without prior knowledge of the attack nature. The major challenges that anomaly IDS have to solve are the improvement of the detection process and the reduction of the number of the false alarms ([1]).

MOTIVATION

The task of IDS could be modelled with various methods and in various levels of description of normal behaviour and current activity monitoring. A typical supervised anomaly recognition model will analyze data, compare to a known profile, run statistical analysis to determine if any deviation is significant, and flag the event(s) as a normal activity or an attack. On contrary, unsupervised approaches do not need any description of normal user activity, since they try to create a real-time model of legal activities in current data traces. All data, which does not conform to the described model, is marked as anomalous.

The present paper describes an adaptive approach for anomaly intrusion detection using 2-means clustering with purpose to examine the effectiveness of both approaches – supervised and unsupervised. The paper compares the results of evaluations of the performance of the

[*] This work was supported under project 6/2013 of Burgas Free University.

following two models – the first one, denoted with A, applies a 2-means clustering anomaly detection technique; and the second, denoted with B, applies 2-means clustering algorithm with combination with some data mining techniques, i.e. classification trees. As cluster distance in both cases was used Wagner-Fischer distance.

THE DESCRIPTION OF THE METHODOLOGY

Distance metric

The intrusion detection itself is performed using different distances, which measure the degree of proximity between normal and real-time data sequences. We stop our attention to the Wagner-Fischer distance.

Wagner-Fischer distance (*WFD*) [15] is a string metric between two strings, which stands for the minimum number of operations (insertion, deletion, substitution of a single character, transposition of two characters) needed to transform one string into the other. Let the weighting for the cost of transforming symbol *a* into symbol *b* be denoted by $w(a,b)$. Then $w(a,b)$ is the cost of a symbol substitution $a \to b$, $w(a,\varepsilon)$ is the cost of deleting *a* and $w(\varepsilon,b)$ is the cost of inserting *b*. The *WFD* are computed using the following recurrence relation:

$$d_{WF}(i,j) = \min \begin{Bmatrix} d(i-1,j) + w(x_i,\varepsilon), d(i,j-1) + w(\varepsilon,y_j), \\ d(i-1,j-1) + w(x_i,y_j) \end{Bmatrix}.$$

It calculates the cost of the optimal string alignment, which does not equal the edit distance. The cost of the optimal string alignment is the number of edit operations needed to make the strings equal under the condition that no substring is edited more than once. This value is referred to as *restricted edit distance*.

Proposed clustering algorithm

K-means clustering [12] is the algorithm of cluster analysis, which groups the objects in K disjoint clusters, based on the distance function. In our case, the goal is to divide them into two classes, one of the normal data, and the other – for the anomalies. The algorithm in this case consists of the following steps:

- Two arbitrary different objects for centres – one of the normal observations, and the other – from the anomalies, are selected.
- When all observations are classified in their closest clusters, the centres of clusters are recalculated. The *j* new centre is determined by

$$\xi_j = \arg\min_{\xi} \sum_{i:\pi_i = j} d(x_i,\xi),$$

  where $\pi_i = \arg\min_j d(x_i,\xi_j)$, $d(,)$ - the measure of the distance between two vectors, in this case – *WFD*.
- The Step 3 is repeated until to find the exact centre of each cluster.

Classification tree

Classification tree is a frequently applied data mining technique in the field of intrusion detection [4], [11]. In our approach the implementation of the classification trees is performed through the process of description of the normal system activity. The normal activity patterns compose a set *Q* with *N* states: $q_1, q_2, \ldots, q_N$ which the system passes through its work in the discrete moments of time $t=1, \ldots, T$. We assume that the probability of occupying a state is

determined solely by the preceding state. Each state transition probability represents the probability of transitioning from a given state to another possible state. Based on the state transition probabilities, we construct classification trees of level *L*, whose roots are all possible states q_k, $k=1,...N$. The inheritors for each vertex are the states for which the corresponding transition probabilities from their predecessor are non-zero.
By traversing the tree from the root to the leaves we can receive all possible state sequences with length *L* along with the corresponding transition probabilities. The obtained lists of system calls consist of all possible sequences with given state in k^{th} position and contain states for which the transition probabilities for each couple of neighbors is non-zero.
Within the created classification trees the *WFD* was applied between the received sequence and normal sequences for the number of errors calculation. The obtained value indicates the degree of similarity between the normal and observed sequences, which is considered as basis for the current activity classification.

EXPERIMENTAL RESULTS

Simulation data

Extensive empirical testing of the proposed methodology was performed on the data, generated and published by the researches in Immune Systems Project from the Computer Science Department, University of New Mexico [14]. The data are obtained from Unix system examination during a period of time and consist of normal user activity patterns of some privileged processes executed on behalf of the root account as well as some anomalous data. The methods for pattern generation are described in [7] and [8]. They substantiate that the short sequences of system calls are reliable discriminator between normal and anomalous activities in the system. Each pattern is a sequence of system calls, which are the results of the examined process. The input data files are sequences of ordered pairs of numbers, where each line consists of one pair. The first number in each pair is the process ID (PID) of the process executed, and the second one is the system call number.
As a first stage of the experiments, a 2-means clustering algorithm, described above, was applied and the current activity sequences were divided into two different clusters. As a second stage of testing model B, based on the normal user activity patterns, the state transition probabilities for the sequences of the normal system activity were evaluated and the normal database, which consists of the classification trees of level *L*, was created. These trees compose the normal program behavior profiles. During the last stage, which is the intrusion detection itself, the anomalous data were divided into portions of length *L* and compared to the lists, extracted by the trees in normal database. The testing data contain both normal and anomalous patterns for the following processes: inetd, login, named and synthetic sendmail.

Performance measures

One of the measures of quality of a cluster algorithm using external criterion is the Jacard index. The *Jaccard index* [9] is used to quantify the similarity between two datasets. An index of 1 means that the two dataset are identical and an index of 0 indicates that the datasets have no common elements. The Jaccard index is defined by the following formula:

$$J = \frac{TP}{TP + FP + FN},$$

where *TP* is the number of true positives, *TN* is the number of true negatives, *FP* is the number of false positives and *FN* is the number of false negatives.

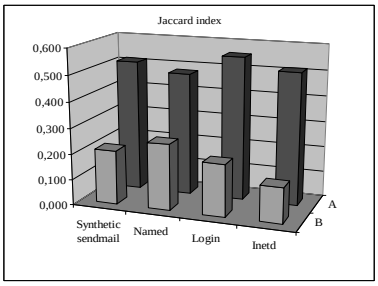


Figure 1. Jaccard index

Figure 1 shows the values of Jaccard index. This index computes the probability that two nodes belonging to a same cluster in a partition also belong to a same cluster in the other partition. The obtained values indicate the good quality of data separation into clusters, i.e. classification of normal or anomalous traffic.

The receiver operating characteristic (*ROC*) [6] curve is a method of graphically demonstrating the relationship between sensitivity and specificity, where sensitivity evaluates intrusion correctly detected and specificity evaluates how well a binary classification test correctly identifies the negative cases. Mathematically, its are expressed as follows

$$Sensitivity = \frac{TPR}{TPR + FNR} \qquad Specificity = \frac{TNR}{TNR + FPR},$$

where the false negative rate (*FNR*) represents undetected attacks on a system, the true positive rate (*TPR*) represents intrusion correctly detected, the false positive rate (*FPR*) represents the frequency with which the IDS reports malicious activity in error and the true negative rate (*TNR*) represents an IDS that is correctly reporting that there are no intrusions.

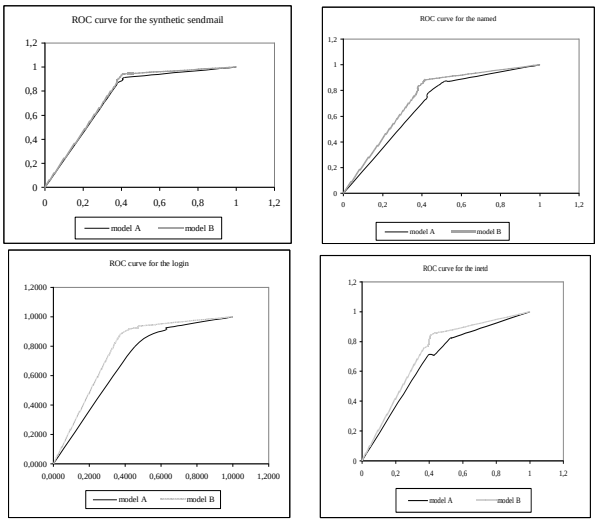


Figure 2. ROC curve

An ROC space is defined by 1-sensitivity and specificity as *x* and *y* respectively, which depicts relative trade-offs between true positive and false positive. Maximal sensitivity is realized when all tests are reported as abnormal. Specificity moves in concert from 0 (no true negatives) to one (no false positives). Maximal specificity is achieved by reporting all tests as normal. The best possible prediction method would yield a point in upper left corner (0,1) of the ROC space, representing 100% sensitivity (all true positives are found) and 100% specificity (no false positives are found). This point is called a perfect classification. The

diagonal line (from the left bottom to the right corner) divides the ROC space in areas of good and bad classification. Points above this line indicate good classification results, while points below the line indicate wrong results.
Each sensitive value can be plotted against its corresponding specificity value to create the diagrams for the processes synthetic sendmail, named, login and inetd in the case of models *A* and *B* in Figure 2. Since the graphs are over the diagonal line this methodology gives good classification results such as the better results are given by the model *B* for the all processes. When we compare the ROC curves for all examined processes we can make the conclusion that it is recommended to use the model *B*.

Cluster validity assessment
When analyzing the cluster it is natural to assume that the cluster with a greater number of vectors is a cluster comprising a vector of the normal operation, and the other contains anomalies. Vectors in the same cluster are similar, which usually means that they are “close” to each other. Although it seems illogical, in the case of large-scale attacks could be seen that more vectors are generated by anomalies of the normal vectors. For this reason for a better classification should be analyzed the structures of the clusters. For this purpose, the size and the distance between the clusters are calculated. The compactness is used to describe similarities between objects in the same class. As measure for cluster compactness intra-cluster distance is applied

$$\Delta(K_i) = \max_{x,y \in K_i} \{d(x,y)\},$$

where $\Delta(K_i)$ - intra-cluster distance, $d(,)$ - the measure of the distance between two vectors in cluster K_i.
It is small when the objects are close to their cluster-centroids. It increases if the number of clusters decreases.
The separability measure provides an evaluation of distances between the classes. Inter-cluster distance are used for measure the separability. It is small when there are a few large clusters. One way to calculate it is to find shortest distance between two observations belonging to two different clusters. Higher the inter cluster distance indicating much better distance between the center of the clusters.
In order to evaluate validation through which assesses the compactness of the clusters and the distances between them, we use *Davies-Bouldin index* and C-index.

- *Dunn index*. The Dunn index [3] is a metric for evaluating clustering algorithms. It is defined as the ratio between the minimal inter-cluster distance to maximal intra-cluster distance. The Dunn index is limited to the interval $[0,\infty)$ and its higher value indicates better clustering.
- *Davies-Bouldin index* [2] takes into accout both the error caused by representing the data vectors with cluster centroids and the distance between clusters. It is defined as:

$$DB(K) = \frac{1}{k}\sum_{i=1}^{k} \max_{i \neq j} \left\{ \frac{\Delta(K_i) + \Delta(K_j)}{\delta(K_i, K_j)} \right\},$$

where *n* is the number of clusters, $\Delta(K_i)$ - intra-cluster distance, $\delta(K_i, K_j)$ - inter-cluster distance. Small values of Davies-Bouldin index correspond to clusters that are compact and whose centers are far away from each other.

- *C-index.* The C-index [5] is a cluster similarity measure expressed as:

$$C = \frac{S - S_{\min}}{S_{\max} - S_{\min}},$$

where S is the sum of all distances between pairs of observations in the same cluster over all clusters. Let n be the number of these pairs. S_{min} and S_{max} are the sums of n lowest/highest distances across all pairs of observations. The C-index is limited to the interval $[0,1]$ and should be minimized.

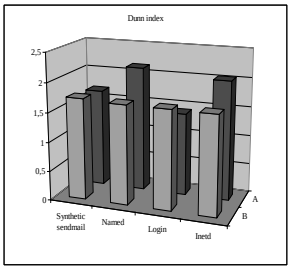


Figure 3. Dunn index

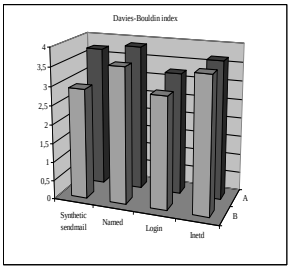


Figure 4. Davies-Bouldin index

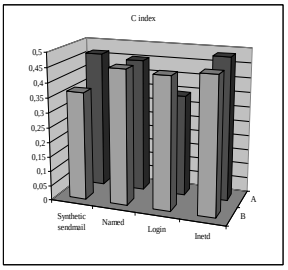


Figure 5. C-index

Figure 3 shows Dunn validity index values. The main goal of the measure is to determine whether the proposed method achieved maximize inter-cluster distance and minimize the intra-cluster distance. From Figure 3 it is clear that we are getting good performance results in terms of the Dunn index.
Figure 4 shows Davies-Bouldin validity index. This index attempts to minimize the average distance between each cluster and the one most similar to it. The presented values reveal that the applied clustering algorithm yields good performance in data separation into clusters.
Figure 5 contains the obtained values of C validity index. C-index values should be minimized. From Figure 5 could be observed, that all obtained values belong to the interval (0.3, 0.5), which means reliable clusterization of the examined data.

CONCLUSIONS

The results, obtained for models *A* and *B* and presented in Figures 1-5, indicate reliable and stable classification results for both models. Some of the values indicate that model B yields better performance results than model A, but it has to be mentioned it takes more time and resources than model A. The purpose of the future work could be comparison with different anomaly detection techniques and obtained results.

REFERENCES

1. Dagorn N., WebIDS: A Cooperative Bayesian Anomaly-Based Intrusion Detection System for Web Applications, Recent Advances in Intrusion Detection, LNCS, Vol. 5230/2008, Springer Berlin / Heidelberg, pp. 392-393.
2. Davies, D.L., Bouldin, D.W., (2000) A cluster separation measure, *IEEE Trans. Pattern Anal. Machine Intell.*, 1(4), 1979, 224-227.
3. Dunn, J. (1974) Well separated clusters and optimal fuzzy partitions, *Journal of Cybernetics* , 4, 1974, 95-104.
4. Gorodetsky, V., Karsaeyv, O., Samoilov, V., Multi-agent technology for distributed data mining and classification, In Proceedings of IEEE/WIC International Conference on Intelligent Agent Technology, 2003. IAT 2003, pp. 438- 441.
5. Hubert L, Schultz J. Quadratic assignment as a general data-analysis strategy . *British Journal of Mathematical and Statistical Psychologie*, 1976; 190-241.

6. Ferri C., N. Lachinche, S. A. Macskassy, A. Rakotomamonjy, *Second Workshop on ROC Analysis in ML*, 2005.
7. Forrest S., S.A. Hofmeyr, A. Somayaji, T.A. Longtaff, A Sense of Self for Unix Processes, In Proceedings of the 1996 IEEE Symposium on Security and Privacy, IEEE Computer Society Press, Los Alamitors, CA, pp.120-128.
8. Forrest S., S.A. Hofmeyr, A. Somayaji, Intrusion detection using sequences of system calls, Journal of Computer Security, Vol. 6, 1998, pp. 151-180.
9. Jaccard P., Distribution de la florine alpine dans la Bassin de Dranses et dans quelques regions voisines, *Bulletin de la Societe Vaudoise des Sciences Naturelles*, 37, 1901, 241-272.
10. Jyothsna V., V. V. Rama Prasad, K. Munivara Prasad, A Review of Anomaly based Intrusion Detection Systems, International Journal of Computer Applications, Volume 28– No.7, August 2011, pp. 26-35.
11. Li X. B., A scalable decision tree system and its application in pattern recognition and intrusion detection, Decision Support Systems, Vol. 41, Issue 1, November 2005, pp. 112-130.
12. MacQueen J., Some methods for classification and analysis of multivariate observations, Proceedings of 5-th Berkeley Symposium on Mathematical Statistics and Probability. University of California Press, 1967, pp. 281-297.
13. Marinova V., A Short Survey of Intrusion Detection Systems. Problems of Engineering Cybernetics and Robotics, 58:23–30, 2007.
14. [UNM] University of New Mexico’s Computer Immune Systems Project, http://www.cs.unm.edu/~immsec/systemcalls.htm.
15. Wagner R. A., M. J. Fischer, The string-to-string correction problem, *Journal of the Association for Computing Machinery* 21, pp. 168-173, 1974.

Списание „Компютърни науки и комуникации”, Том 2, № 3 (2013), БСУ, Бургас

ANALYSIS OF MICRO BENDING LOSSES IN OPTICAL FIBER

Angel Toshkov

***Abstract*:** This paper reviews the influence of some design parameters of optical fibers to the macro bending losses. Below are the graphical results and are some conclusions

Keywords: macrobending, losses, optical, fiber, graphics, relations, influence, input, parameters.

АНАЛИЗ НА ЗАГУБИТЕ ОТ МАКРООГЪВАНИЯ В ОПТИЧНИ ВЛАКНА

Ангел Тошков

Абстракт: В статията се прави преглед на влиянието на някои конструктивни параметри на оптичните влакна върху загубите от макроогъвания. Представени са графични зависимости и са направени изводи от получените резултати.

Ключови думи: макроогъвания, конструктивни, параметри, влияние, графики, оптични влакна.

Въведение

Голяма част от загубите в оптичните влакна са в пряка зависимост от технологията на производство – например загубите от поглъщане от примеси, такива, които се дължат неконцентричност, елиптичност и др. Част от тези загуби имат постоянен характер и не зависят съществено от входните параметри на влакното.

Една част загубите, обаче се влияе много силно от конструктивните параметри. Това са загубите от огъвания, и по точно загубите от микро и макро огъвания. Поради тази причина, на този тип загуби се обръща особено внимание.

Загуби от макро огъвания има тогава, когато влакно е огънато с радиус по-голям в сравнение с диаметъра на влакното. Тези загуби се определят както обикновено по формулата $\boldsymbol{P(z) = P(0)e^{(-\gamma z)}}$ където $\boldsymbol{\gamma}$ са загубите от макроогъвания в децибели, $\boldsymbol{P(0)}$ е вкараната във влакното мощност, а $\boldsymbol{P(z)}$ е изходната мощност, съответно на разстояние $\boldsymbol{z}$.

В статията са дадени някои комплексни трикоординатни графични зависимости на загубите от макроогъвания за да могат да се видят тенденциите при комплексна връзка на два параметъра и тяхното взаимно влияние върху наблюдавания изходен параметър.

За пресмятането на загубите от макроогъвания има нняколко различни модела. Един от тези модели [1], [2], е подходящ за изчисляване на загубите от макро огъвания за всеки режим LP, за основните и тези от по-висок порядък моди, при произволен профил на коефициента на пречупване на оптичните влакна. Използвайки уравнение 1 коефициентът $\boldsymbol{\gamma}$ на загубите от макро огъвания се изразява като функция на мощности, и на радиуса на огъване $\boldsymbol{R_b}$ в следния вид:

$$\gamma = \frac{\sqrt{\pi}\left(\frac{P_{clad}}{P}\right)}{2sr_c\left[K_{v-1}(W)K_{v+1}(W)-K_v^2(W)\right]} \cdot \frac{\exp\left(\frac{-4\Delta W^3}{3r_cV^2}R_b\right)}{W\left(\frac{WR_b}{r_c}+\frac{V^2}{2\Delta W}\right)^{\frac{1}{2}}} \qquad \textbf{1)}$$

Параметрите, включени по-горе са [3]:

$V = k_0 r_c \sqrt{n_{\max}^2 - n_{\min}^2}$ - *нормирана характеристична честота (бездименсионна стойност)*

$W = r_c \sqrt{\beta^2 - (k_0 n_{\min})^2}$ - *лента на пропускане (MHz km)*

$\Delta = \frac{\left(n_{\max}^2 - n_{\min}^2\right)}{2n_{\max}^2}$ - *нормираната разлика в коефициентите на пречупване (бездименсионна стойност)*

- $\boldsymbol{r_c = d/2}$ - радиус на ядрото на влакното;
- $\boldsymbol{n_{max}}$ и $\boldsymbol{n_{min}}$ са максималните и минималните стойности на индекса на пречупване;
- $\boldsymbol{\beta}$ е константа на разпространение при конкретния режим;
- $\boldsymbol{k_0}$ е константа във вакуум;
- $\boldsymbol{v}$ е азимуталния номер на модата;
- $\boldsymbol{S}$ = 2, ако $\boldsymbol{v}$ = 0 или **S** = 1 за $\boldsymbol{v} \neq 0$;
- $\boldsymbol{K_v}$ е промяна на функцията на Бесел ;
- $\boldsymbol{P_{clad}}$ е мощността, разпространяваща се в обвивката;
- $\boldsymbol{P}$ е мощността, разпространяваща се в ядрото на влакното;
- $\boldsymbol{R_b}$ е радиуса на огъване на влакното.

Уравнение 2 е изведено по друг [2] модел за пресмятане на коефициента $\boldsymbol{\gamma}$ за загуби от макро огъвания.

$$\gamma = \left(\frac{\pi V^8}{16 r_c R_b W^3}\right)^{\frac{1}{2}} \exp\left(-\frac{4}{3}\frac{R_b}{r_c}\frac{\Delta W^3}{V^2}\right) \frac{\left[\int_0^\infty (1-f)F_0 R dR\right]^2}{\int_0^\infty F_0^2 R dR} \qquad \textbf{2)}$$

където $\boldsymbol{F_0}$ е диаметърът на радиалното модово поле на основната мода $\boldsymbol{LP_{01}}$ или , $\boldsymbol{F_0} = 2w_0$.

Радиусът на модовото поле $\boldsymbol{w_0}$ при постоянен радиус на сърцевината $\boldsymbol{a}$ е функция само на $\boldsymbol{V}$ параметъра, който от своя страна зависи от числената апертура $\boldsymbol{NA}$ и дължината на вълната $\boldsymbol{\lambda}$ [4]. На базата на този модел е изведена проста апроксимация за едномодово влакно със стъпално-радиален профил, която в обхвата за $\boldsymbol{V}$ параметъра 1,6 < V < 2,6 описва зависимостта на $\boldsymbol{w_0}$ от радиуса на сърцевината $\boldsymbol{a, \lambda, V}$ параметъра и $\boldsymbol{\lambda_c}$ - критичната дължина на вълната и:

$$2w_0 \approx \frac{2.6}{V} a \ \frac{\lambda}{\lambda_c}, \qquad \textbf{3)}$$

$$f = \frac{n(r)^2 - n_{\min}^2}{n_{\max}^2 - n_{\min}^2} . \qquad \textbf{4)}$$

n(r) е коефициентът на пречупване от профила на влакното в точка от ядрото;

r – е разстоянието от центъра на ядрото до точката в която правим изчислението.

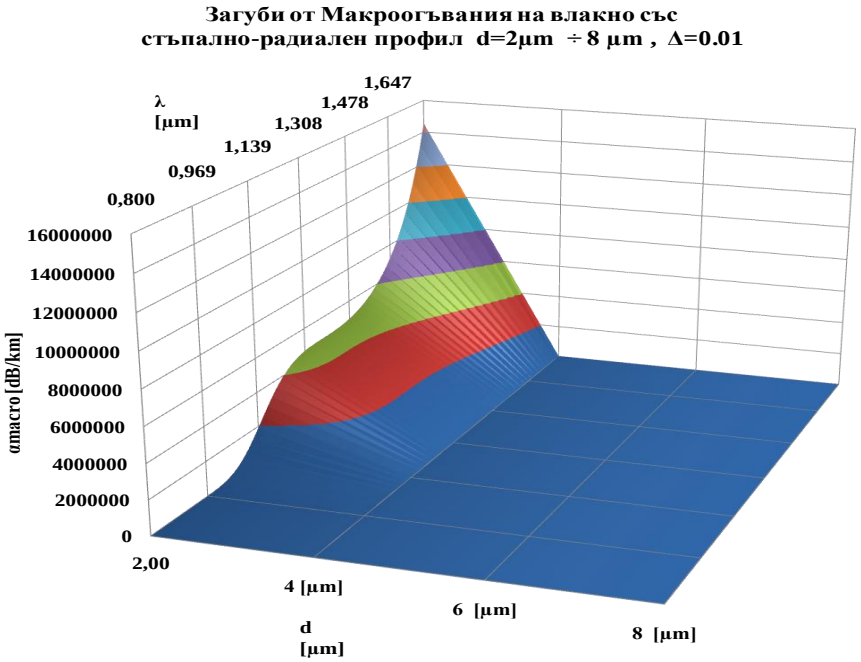
Тези два модела дават подобни резултати [3] за влакна със стъпално-радиален профил на коефициента на пречупване.

Коефициентът на загубите ***γ*** може да бъде конвертиран до загуби в децибели на километър, както следва:

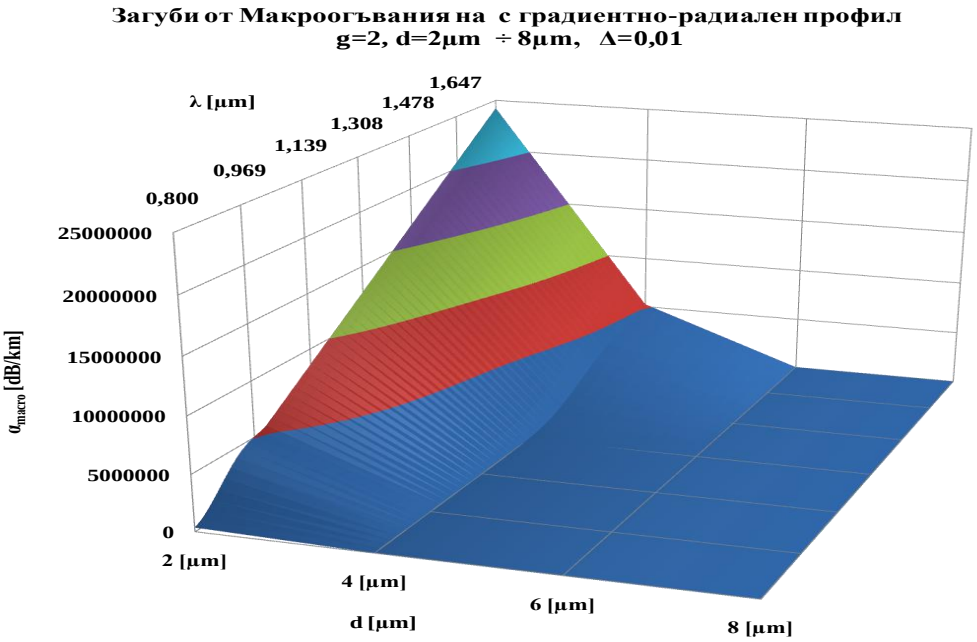
$$\alpha_{macro} = \frac{10}{L} \log \frac{P_{in}}{P_{out}} = \frac{10}{L} \log[\exp(\gamma L)] = \frac{10}{\ln 10} \gamma \text{ dB/km.} \qquad \textbf{5)}$$

Графични зависимости на загубите от макроогъвания [3]

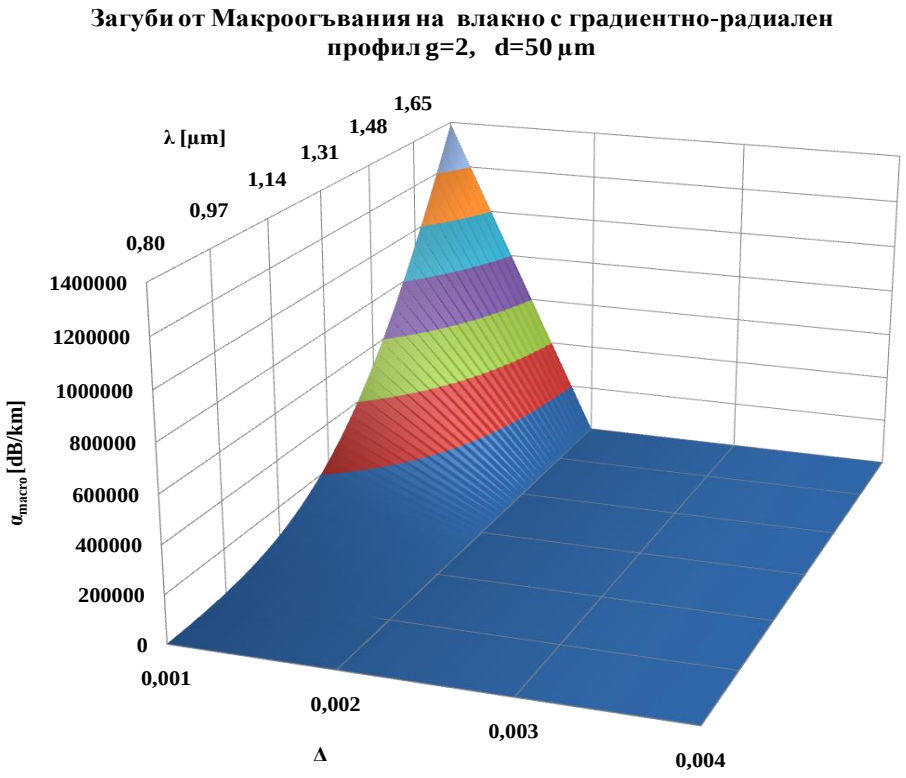
На базата на показаните по-горе модели, по надолу са показани в графичен вид връзките между някой от управляващите параметри - ***n, d, Δ и g*** и загубите от макроогъвания. За извеждането на тези резултати е избрана една от популярните функции за изграждане на профила на коефициента на пречупване g - pic функция - $n(x) = n_{max}\sqrt{1 - 2\Delta\left(\frac{a}{r}\right)^g}$, където ***g*** е стойността на степенния показател. Входните параметри се променят в диапазони подходящи за производството на реални едномодови и многомодови оптични влакна.



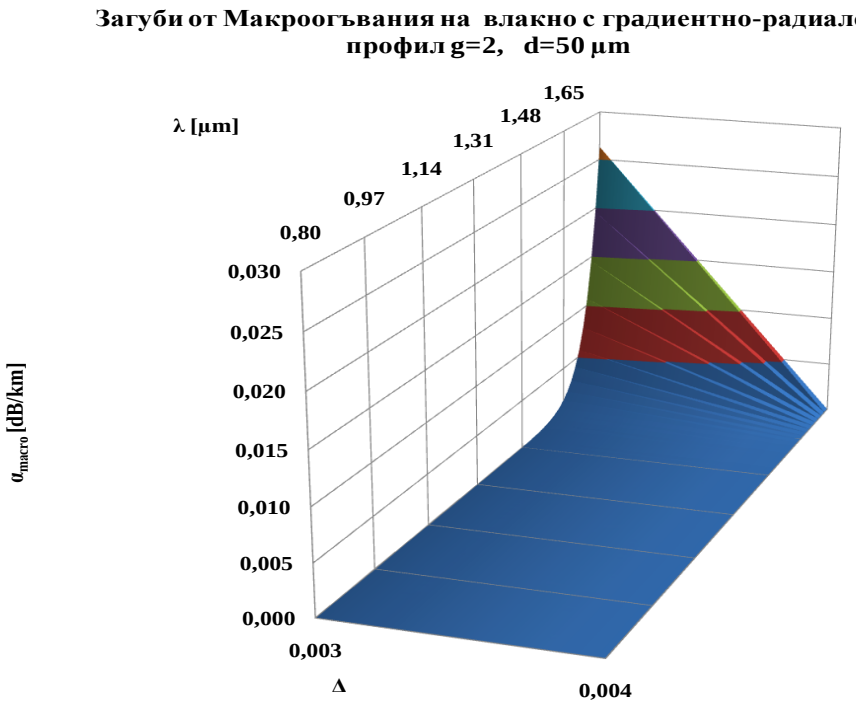
Фигура 1



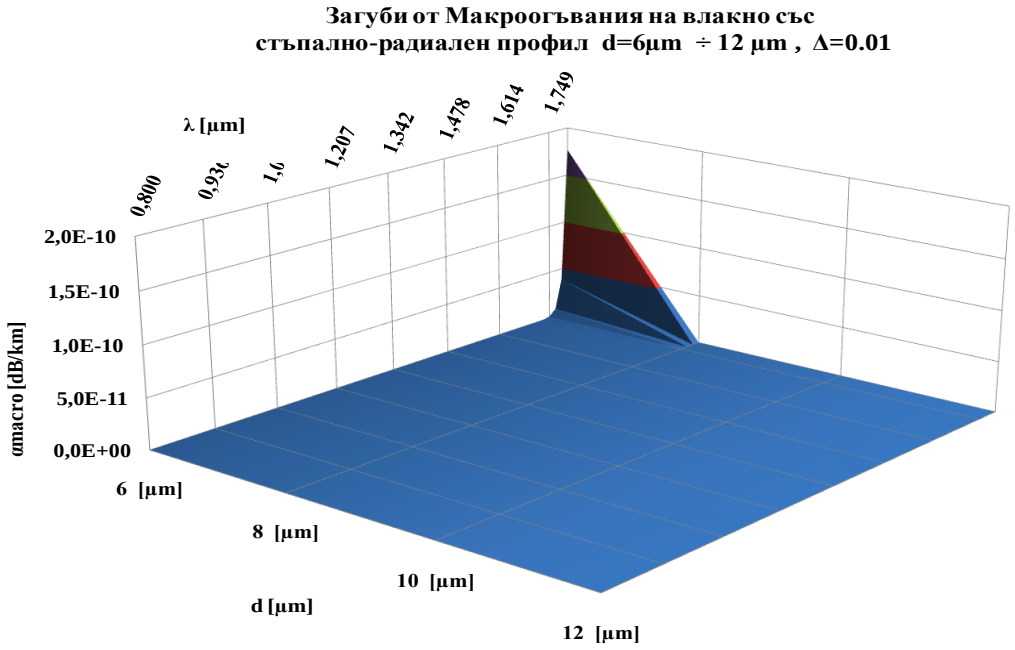
Фигура 2



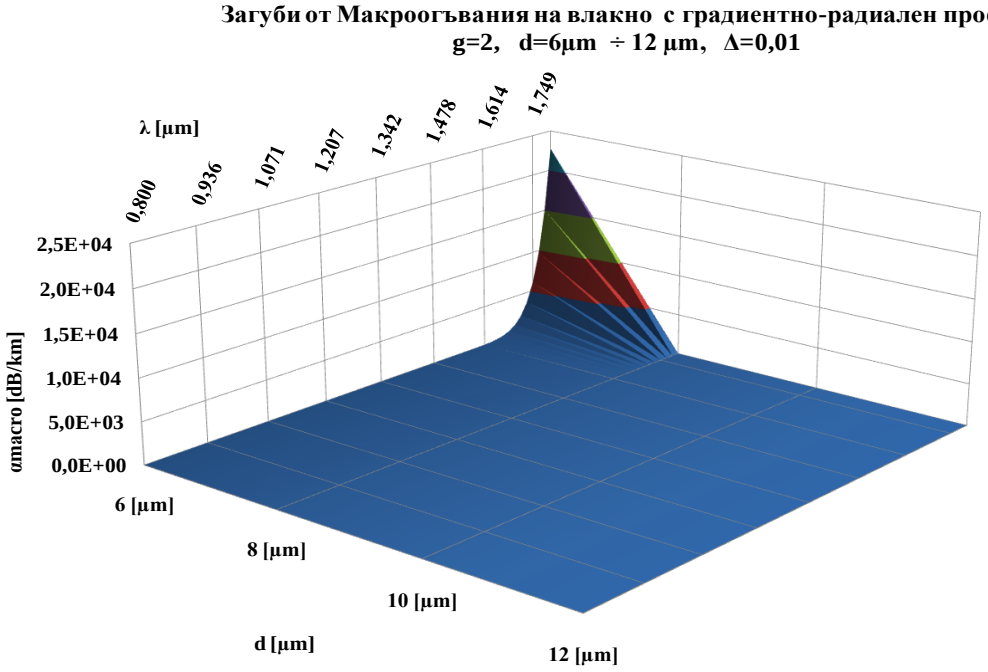
Фигура 3



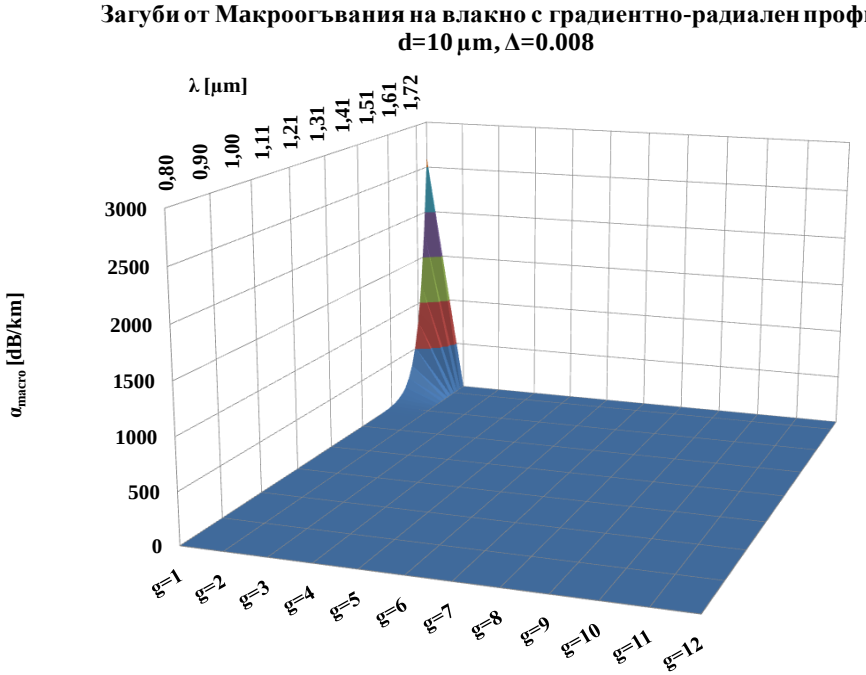
Фигура 4



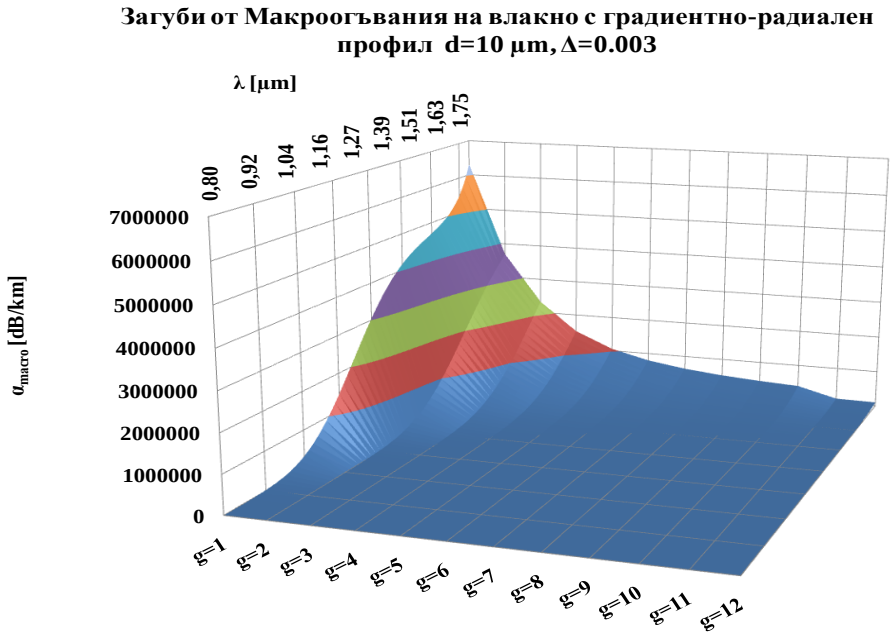
Фигура 5



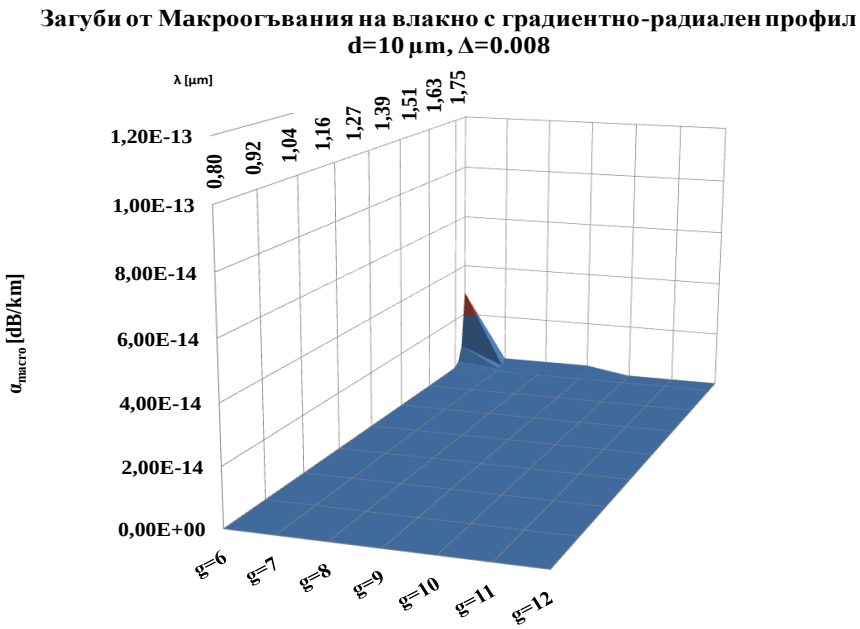
Фигура 6



Фигура 7

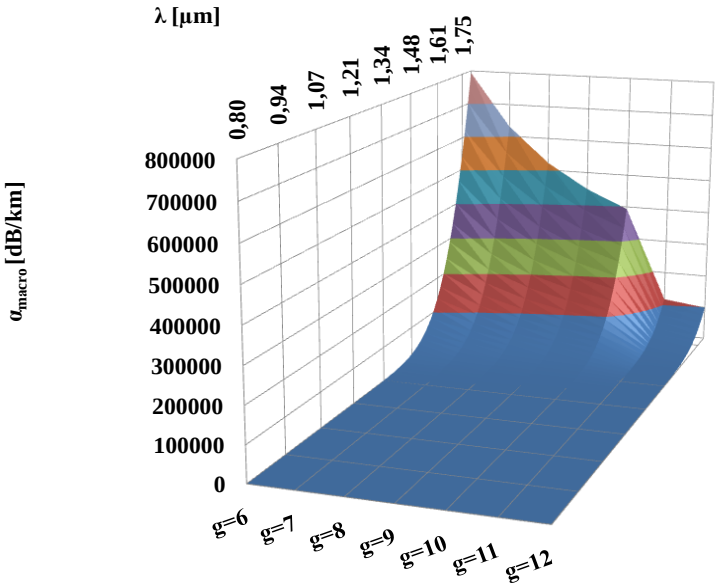


Фигура 8



Фигура 9

Загуби от Макроогъвания на влакно с градиентно-радиален профил d=10 μm, Δ=0.003



Фигура 10

Изводи.

Загубите от макроогъвания проявяват силно изразен характер при нарастване на дължината на вълната. Това се отнася особено до влакната с профил близък до градиентния и за дължини на вълните в оптичен прозорец II и III - Фиг. 1 до 10. [3] [4]

Трябва да отбележим факта, че за влакна с Δ=0,01 и g=2 загубите имат изключително високи стойности достигащи до 10^6 и нагоре dB/km при диаметри на ядрата под 4μm (Фиг. 1,2,3).

Подобен ефект се наблюдава и при намаляването на Δ, като при стойности на g под 2, Δ под 0,005 и диаметър на ядрото d под 4-6μm тези ефекти имат силно изразен характер Фиг. 3,4,9.

При оптични влакна с голям диаметър d > 20μm, тези загуби нямат практическа стойност (освен при много малки стойности на Δ< 0.003 в комбинация с g=1) Фиг.5 и 6.

Този тип загуби нарастват значително при намаляване на диаметъра на ядрото – обикновено под 6-4μm. Фиг. 1,2,

Същото се отнася и за намаляването на Δ под 0,005, както и за намаляването на g под 2. Фиг. 8,9,10.

Необходим е много внимателен подбор на управляващите параметри n, d, Δ и g, особено при конструиране на влакна с дължини на вълната в оптичен прозорец III, за да се постигнат минимални загуби от макроогъвания.

References:

[1]. J. Sakai, and T. Kimura, “Bending loss of propagation modes in arbitrary-index profile optical fibers”, Applied Optics vol. 17, no. 10, pp 1499-1506, May 1978

[2]. Optical Fiber Design Software, Technical Documentation, 2006 Optiwave.

[3]. ТОШКОВ А., „Анализ на връзките между конструктивните и експлоатационните параметри на оптичните влакна”, Монография, Бургас, 2011, Изд. „Онгъл”, Варна.

[4]. ТОШКОВ А., „Оптични комуникации”, БСУ, 2005.

GENETIC ALGORITHMS AS A TOOL FOR SOLVING OPTIMIZATION PROBLEMS

Georgieva Penka, Burgas Free Univercity, e-mail: pgeorg@bfu.bg

***Abstract**: Genetic algorithms as a heuristic method, based on the principles of natural selection, can successfully be applied in the modeling of nonlinear optimization problems with nonlinear objective functions, including cases where these functions do not satisfy the conditions for continuity, and what is more the arguments can be both discrete and continuous variables.*
Constructing a portfolio is a major task in making financial investment decisions. In this task, the an objective function is been optimized to obtain shares of the portfolio assets which maximize return and minimize risk of the investment.
This article is a brief overview of genetic algorithms and a concept for their application for constructing optimal portfolio is proposed.

***Keywords**: artificial intelligence methods, genetic algorithm, portfolio optimization*

ГЕНЕТИЧНИТЕ АЛГОРИТМИ КАТО СРЕДСТВО ЗА РЕШАВАНЕ НА ОПТИМИЗАЦИОННИ ЗАДАЧИ

Пенка Георгиева, Бургаски свободен университет, e-mail: pgeorg@bfu.bg

Абстракт: *Генетичните алгоритми, като евристичен метод, основан на естествените принципи за подбор, могат успешно да бъдат прилагане при моделиране на нелинейни оптимизационни задачи с нелинейни целеви функции, включително и в случаите, в които не са удовлетворени условията за непрекъснатост, а аргументите могат да са както дискретни така и непрекъснати величини.*
Конструирането на портфейл е основна задача в процеса на вземане на финансовите инвестиционни решения. При решаването на тази задача се дефинира целева функция, от която чрез оптимизиране да се получат дялове на включените в портфейла активи, максимизиращи възвращаемостта и минимизиращи риска от инвестицията.
В тази статия е направен кратък обзор на генетичните алгоритми и е предложена концепция за прилагането им при конструиране на оптимален портфейл.

***Ключови думи**: методи на изкуствения интелект, генетични алгоритми, портфейлна оптимизация*

1. Въведение

Развитието на компютърните технологии са свързани с наподобяване и автоматизиране на различни човешки умения и способности, комбинирани с бързината и точността на пресмятане на изчислителните машини, при което се разработват различни методи за решаване на всяка конкретна задача, известни като алгоритми.
Алгоритъм е термин от математиката, информатиката, лингвистиката и други области, с който се означава крайна поредица от инструкции или описание на постъпкова

процедура за решаване на даден проблем, често свързан с изчисление или обработка на данни. Етимологията на думата произлиза от името на персийски математик и астроном Al-Khwārizmī (на персийски : خوارزمی) [1]. С други думи, алгоритъмът е ефективен метод, който при даден списък от коректно дефинирани команди за изпълнение на задача и зададено начално състояние, преминава през коректно дефинирана поредица от последователни състояния и завършва в едно крайно състояние. Преходът между състоянията не е задължително да е детерминиран, тъй като някои алгоритми, известни като вероятностни алгоритми съдържат в себе си случайност. В компютърните технологии алгоритмите се използват за пресмятане, обработка на данни и в много други области.

Оптимизирането на инвестиционен финансов портфейл е важна задача на финансовата практика. През 1959 Хари Марковиц разработва количествен модел [2], който използва очаквани възвращаемости на активите и свързаните с тях вариации и ковариации, като целевата функция е портфейлният риск, изразен като функция на дяловете на активите. Решение на портфейлната задача се получава за тези стойности на неизвестните, които минимизират целевата функция при фиксирана стойност на портфейлната възвращаемост. Поради различни причини (една от които проблемът с кардиналността [3], използването на модела на Марковиц е ограничено. За преодоляване на някои от проблемите се използват различни оптимизационни алгоритми: оптимизация с ограничителни условия, квадратно програмиране, линейно програмиране и др.

Но един важен въпрос все още не е намерил отговорът си: как едновременно да се максимизира възвращаемостта и да се минимизира риска.

Средствата на софт компютинг дават възможност за различен, ориентиран към реалната практика подход. Едно важно направление на софт компютинг, като направление в изкуствения интелект, са еволюционните пресмятания.

Модерната теория на еволюцията се базира върху идеите на Чарлз Дарвин [4] представени в книгата му «Произход на видовете» и многобройни изследвания в областта на генетиката, обединени под общо наименование «Нео-Дарвинизъм». Идеята за прилагането на принципите на Нео-Дарвинизъма за автоматично решаване на задачи възниква през 30-те години на 20 век. През 1948 година, Алън Тюринг предлага използването на «генетично или еволюционно търсене», а през 1962 година, Ханс-Йоаким Бремерман прави първите компютърни експерименти по «оптимизация чрез еволюция и рекомбинация». През 60-те и 70-те различни приложения на основните идеи се разработват едновременно от няколко учени. В САЩ, Фогел, Оуенс и Уолш представят еволюционно програмиране [5], докато Холанд нарича своя метод генетичен алгоритъм [6]. По същото време, в Германия, Рахенберг и Швайфел създават еволюционни стратегии [7]. В следващите две десетилетия тези области се развиват отделно, но от 90-те години на двадесети век те започват да се възприемат като различни представители на една технология, която се нарича еволюционни изчисления (evolutionary computing). През първата половина на 90-те се образува ново направление, наречено генетично програмиране [5]. Днешната терминология използва понятието «evolutionary computing» за цялата област, а всички изброени методи – EP, GA, ES и GP се обобщават като еволюционни алгоритми.

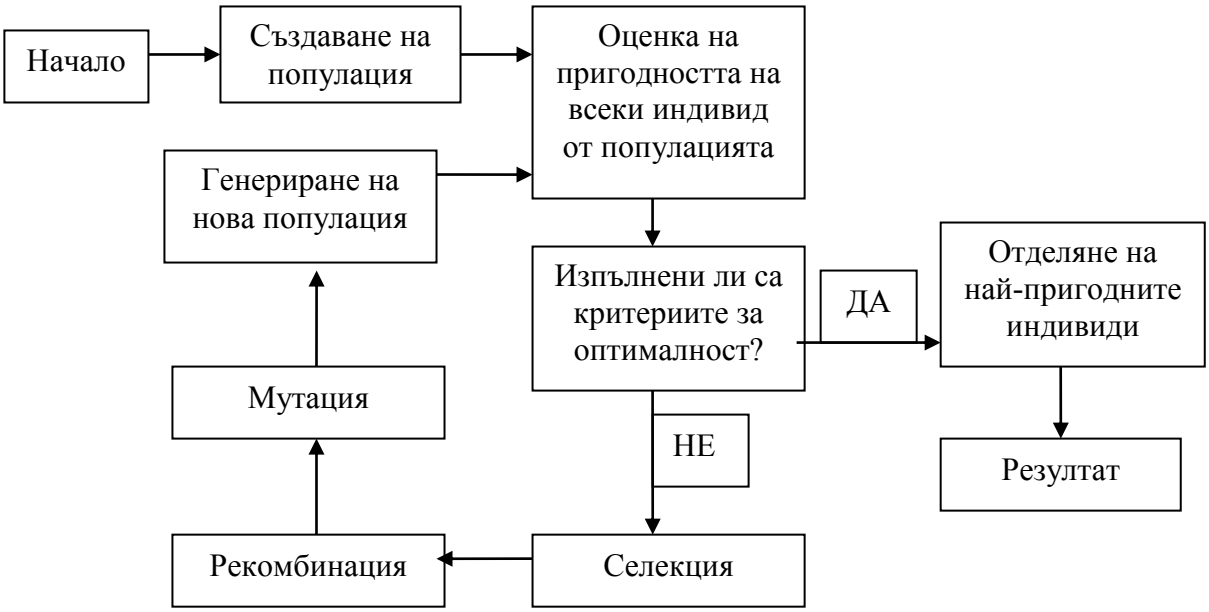
През 1975 година, в книгата си «Adaptation in Natural and Artificial Systems» Джон Холанд формулира базовите понятия на генетичните алгоритми, като модел за машинно обучение, които произхождат от биологичната теория на еволюцията. През 1989 година ученикът му Дейвид Голдбърг в дисертацията си решава сложна задача за моделиране на газопровод с помощта на ГА, което води до тяхното популяризиране.

2. Еволюционни пресмятания

В еволюционните пресмятания се използват оптимизационни алгоритми, които имитират принципи на природата, които са доказали своята пригодност и приложимост. Особено важни са случаите, в които природата е намерила “стабилни острови” в “бурния океан” от възможни решения – като централната нервна система, биологичната еволюция, процеса на закаляване на стоманата, и др.
От използването на принципите на еволюцията са произлезли оптимизационни методи като:

- генетични алгоритми (ГА)- Genetic Algorithms (GA);
- еволюционно програмиране - Evolutionary Programming (EP);
- еволюционни стратегии - Evolution Strategies (ES);
- системи за класифициране - Classifier Systems (CFS);

генетично програмиране - Genetic Programming (GP), които са елементи на еволюционните алгоритми (ЕА).
Всички те симулират еволюцията на индивидуалните структури чрез процесите селекция, мутация и възпроизвеждане и са базирани върху биологични наблюдения, като теорията на Дарвин за естествения подбор и оцеляване на най-пригодните.
ЕА представляват популация от структури, която еволюира според правилата на оператори за селекция, търсене, рекомбиниране на индивиди с висока пригодност и мутация, която променя тези индивиди. Всеки индивид получава мярка за пригодност към средата. Селекцията се концентрира върху избор на най-пригодните индивиди. Рекомбинирането и мутацията променят индивидите (фиг. 1.). [9]
В процеса на селекция се определя кои индивиди да бъдат избрани за рекомбиниране и от колко индивида да се състои поколението, при което има два етапа: първо се определя критерий за пригодност и всеки индивид получава вероятност, с която ще се включи в процеса на репродуциране в зависимост от неговата и на другите индивиди от популацията пригодност. Методите за определяне на тези вероятности са пропорционален, чрез ранкиране, чрез многозначно ранкиране. В следващия етап се избират индивидите за репродукция: чрез рулетка, чрез случайна извадка, чрез локална селекция, чрез състезание.
При рекомбинацията се получават комбинации от информацията на родителите по различни методи: бинарна, линейна, дискретна и др. След рекомбинацията се прилага мутация. Новите индивиди заменят някои от старите и процесът се повтаря до достигане на критериите за оптималност.
Еволюционното програмиране, предложено от Фогел през 1966, се характеризира с това, че при кодирането на индивидите се използват представяния, пряко свързани с изследваната област: вектори с реални координати за оптимизационни задачи върху реална област; списъци за задачата за търговския пътник и др. Всички индивиди стават родители и от тях се получава същият брой деца. От получените два пъти повече на брой индивиди се избират най-пригодните и те мутират. При тези мутации се използват нива на мутация, които ако се изберат със стандартно разпределение водят до ефект, подобен на рекомбинирането.



Фиг. 1. Схематичен модел на еволюционен алгоритъм

Еволюционните стратегии с популация с обем 1 са предложени от Рехенберг през 1973. През 1981 Швефел предлага модел с по-голям обем на популацията. При селектирането на родители се използва равномерно случайно разпределение (т.е. не се отчита ниво на пригодност). Рекомбинацията става след като се комбинират родители по двойки и получените деца са повече от първоначалния брой. В следващия етап се избират индивиди за оцеляване по един от два различни начина: или най-пригодните деца оцеляват и заместват родителите си или оцеляват най-пригодните от всички индивиди. Както и при еволюционното програмиране, при мутациите може да се използват нива на мутации с нормално случайно разпределение.

Генетични алгоритми предлага Холанд през 1975, като представянето на индивидите е с битове и популацията се развива при специфични правила за избор до момент, при който „пригодността” е максимална. През 1989 Голдбърг излага този подход в книга и успешно го прилага в газопровод. В по-късни модели се предлагат и други видове представяния. При генетичните алгоритми родителите се селектират чрез вероятностна функция, базирана на относителната пригодност, т.е. колкото е по-пригоден индивидът, толкова вероятността да бъде избран за родител е по-голяма. От първоначалните родители се получават същия брой деца, които след това мутират и заместват родителите си. При ГА акцентът е върху рекомбинирането, а мутациите са съвсем слаби. ГА могат да бъдат разглеждани като алгоритми за търсене, защото при тях се изследват елементите на дадено пространство, като се използват евристични подходи, вдъхновени от природата. Всеки оптимизационен проблем трябва да бъде представен с използването на хромозоми, които са кодирано представяне на реалните стойности на променливите. За различни приложения са разработени бинарен ГА, непрекъснат ГА, паралелен ГА, симулирано закаляване, „роене”, „колонии от мравки”, генетично програмиране, еволюционни стратегии.

Фундаменталната теорема на генетичните алгоритми на Холанд гласи, че ако дадена хромозома (кодирана информация на даден индивид) с фиксирани гени има пригодност над средната, то се наблюдава тенденция броят на индивидите с този шаблон да се увеличава експоненциално. Доказателство на тази теорема в случай на пропорционална селекция с равномерно разпределение дава Конар [10].

Еволюционното програмиране и еволюционните стратегии имат за цел получаване на едно оптимално решение и следователно са добри за решаване на оптимизационни проблеми, докато генетичните алгоритми дават няколко оптимални решения.
В приложенията, базирани на еволюционния подход се акцентира върху следните основни предимства: оптимизацията е възможна с дискретни и непрекъснати величини, няма нужда от информация относно производните, възможно е едновременно претърсване на голяма област от целв.та повърхнина, възможно е опериране с голям брой променливи, има възможност за паралелизация, целв.та повърхнина може да е изключително сложна, има възможност за излизане от локален минимум, като резултат се получават няколко оптимални стойности, а не само едно решение.
През 1996 Холанд публикува системи за класификация, базирана на генетични алгоритми, през 1989 Дейвис – адаптивни системи, през 1990 Гарис – генетично програмиране.
През 1991 Белю и Букман представят разпределени еволюционни алгоритми. Тъй като при еволюционните алгоритми операторите се извършват едновременно върху всички индивиди от популацията, то те могат да бъдат реализирани върху паралелни компютърни системи.

3. Генетични алгоритми и техните приложения

Генетичните Алгоритми (ГА) са евристични алгоритми за търсене, пригодени за решаване на сложни оптимизационни задачи и моделиране чрез механизми, наподобяващи биологичната еволюция.
Като подклас на еволюционните алгоритми, ГА използват методи от еволюционната биология за симулиране на процеси в природни системи, необходими за оцеляване и еволюция (наследяване, мутация, естествен подбор и кръстосване)[11]. Като такива, те представляват методика за случайно търсене в определено пространство на търсене с цел намиране на решение на дадена задача.
ГА са пригодени за решаване на сложни оптимизационни задачи, с които традиционни техники не са в състояние да се справят за рационално количество време. Те не само предоставят алтернативни решения на проблемите, но също така значително превъзхождат други традиционни методи в повечето оптимизационни задачи. Много от задачите от реалния свят за намиране на оптимални параметри, които са трудни за решаване с традиционните методи, са идеални за ГА. Освен това те са подходящи инструменти и за други цели. В много по-широк аспект генетичните алгоритми могат да се разглеждат като:

- средство за решаване на задачи;
- среда за машинно обучение и симулации;
- предизвикателство за технически задачи и основа за нови иновативни системи;
- изчислителни модели за иновативни идеи и творчество.

Основните предимства на ГА са преди всичко способността им да оптимизират параметри с много сложни стойностни функции (cost function) и да работят с разнообразни типове данни. Освен това те могат да пресмятат няколко решения едновременно, обхващайки голяма област на търсене, след което създават списък с най-добрите решения, а не само едно решение. Генетичните алгоритми се представят добре и при паралелно приложение.
Оптимизационните възможности на ГА намират широко приложение в много области на науката, компютърни дисциплини, инженерство, техника, бизнес и развлечения, а също така и в численна и комбинаторна оптимизация.

Приложенията на ГА в оптимизациоони модели са:

- избор на оптимални математически модели за описание на различни видове биологичните системи;
- разпознаване на молекулярната структура и оптимизация в химия;
- оптимизация на системи за компресиране на данни;
- задачата за пътуващия търговец;
- оптимизация на товар на контейнер
- планиране на график;
- дизайн на електронни схеми;
- разработка на водоразпределителни системи;
- оптимизация на инфраструктура на мобилните комуникации;
- безжични сензори/специализирани мрежи;
- видео и аудио оптимизация.
-

ГА се прилагат за разширение на компютърни програми и разработка на други изчислителни структури, такива като:

- средства за програмиране;
- числена оптимизация на компютърни модели;
- топологии на компютърни мрежи и разполагане на файлове за сложни разпределени системи;
- автоматизация на откриване на недостатъци на системата.
-

ГА се използват в различни системи за машинно обучение, включително приложения за класифициране, прогнозиране и симулиране:

- лингвистичен анализ и анализ на филогенетични дървета;
- изчисляване на различни състояния във физика;
- анализ на генетичният профил;
- биоинформатика;
- симулации на химични процеси;
- разработка на мехатронни системи;
- роботика и др.
-

ГА се използват за моделиране на биологични и екологични явления, като миграция на видовете, коеволюция на «носител-паразит», симбиоза и разпределение на ресурсите в екосистемите, а също така еволюционни явления в социалните модели:

- еволюционни аспекти на социални и социотехнични системи;
- алгоритъми, наподочяващи колония от мравки (или рояк пчели);
- топологии на множествени популации и методологии на взаимодействия;
- еволюционна екология и бактериологична адаптация;
- моделиране на естествената имунна система.

ГА намират широко приложение и в областта на теоия на игрите, развлекателната индустрия и компютърно творчество:

- решаване на задачи за равновесие в теорията на игрите;
- еволюционна компютърна музика и развлечения.

4. Концепция за използване на генетичен алгоритъм в портфейлна оптимизация

ГА се използват за моделиране на иновационни процеси, неопределеността на икономическите пазари и разработка на борсови стратегии:

- представяне на различни фактори при икономически модели;
- анализ на пазара и създаване на интелигентни системи за търгуване на финансови активи;
- разпределение на активи;
- инвестиционни стратегии.

Тъй като всеки генетичен алгоритъм е итеративна процедура за търсене на оптимално решение чрез генериране на популации с един и същ обем, то първата стъпка е да се определят кандидатите на популацията и техния брой.

Полпулацията се състои от кандидати, наречени хромозоми, като алгоритъмът е основан на конкуренция между хромозомите. Всяка хромозома е кодирано възможно решение на задачата и се състои от множество от елементи, наречени гени, които могат да приемат различни стойности. При всяка итерация се създава нова популация със същия обем, но съдържаща по-добри хромозоми, т.е. хромозоми, които са се адаптирали към околната среда и това е целевата функция. За целта се дефинира и функция на прогодност, показваща до каква степен популацията удовлетворява търсеното решение. След прилагане на достатъчен брой итерации, хромозомите се приближават към оптималното решение.

Основните оператори в генетичен алгоритъм са три:

- селекция – пресмята се репродуктивната вероятност p_i за всеки индивид:

$$p_i = \frac{f_i}{\sum_{i=1}^{n} f_i},$$

където f_i е стойността на фитнес функцията за *i*-ия индивид и *n* е обемът на популацията. При всяка итерация се избира единствена нова хромозома, така че за произволно случайно число $r \in [0\ ,\ 1]$, ако $r \leq p_1$ да се избира първата, а ако е изпълнено

$$p_{j-1} < r \leq p_j \ ,$$

се избира *j*-тата;

- кръстосване – популацията, получена след селекция се разделя на две равни части, избират се двойки хромозоми от всяка и дадена двойка се кръстосва с вероятност P_c;
- мутация – след кръстосването случайно избрани битове се променят с вероятност P_m.

За търсене на оптимален инвестиционен портфейл се предлага следната процедура:

- фиксира се *n* - желаният брой финансови активи в инвестиционния портфейл и едновременно обем на полулацията;
- прилага се модела FLQM, откъдето се получава множество от портфейли и това са хромозомите;
- дефинира се целевата функция:

$$G(x_1, x_2, \dots x_n) = R_p - \sigma_p,$$

където x_j са дяловете на активите в портфейла, R_p е възвращаемостта на портфейла и σ_p е инвестиционният риск на портфейла;

- търси се максимум на целевата функция $G(x_1, x_2, \dots x_n)$ с генетичен алгоритъм.

REFERENCES

[1] http://en.wikipedia.org/wiki/computational_creativity)
[2] Markowitz H., Portfolio selection. Journal of Finance, 7, (March, 1952), 77-91.
[3] Георгиева П., Оценка на капитала при конструиране на инвестиционни портфейли. ММ XXI (5), 2013
[4] Fogel L., Owens A., Walsh M., Artificial Intelligence through a Simulation of Evolutio”. Spartan, Washington DC, 1965, pp. 131-156.
[5] Darvin Ch., The Origin of Species. John Murray, 1859.
[6] Holland J., Adaptation in Natural and Artificial System”. University of Michigan Press, 1975.
[7] Rechenberg, Evolutionstrategie: optimierung technisher systeme nach prinzipien des biologischen evolution. Fromman-Hozlboog Verlag, Stuttgart, 1973.
[8] Poli R., etc., “A Field Guide to Genetic Programming”. Published via: http://lulu.com and freely available at http://www.gp-field-guide.org.uk, 2008 (with contributions by J. R. Koza)
[9] Garkaz M., The Selection and Optimization of Stock Portfolio using Genetic Algorithm based on Mean-semi Variance Model, International Conference on Portfolio Selection Using Genetic A Economics and Finance Reaserch, IPEDR, LACSIT Press, Singapore, 4, (2011), 379-381.
[10] Lin D., Xiaoming Li, Mingiang Li, A Genetic Algorithm for Solving Portfolio Optimization Problems with Transactions Costs and Minimum Transactions Lots. Proceedings of the First International Conference on Advances in Natural Computation ICNC'05, Publisher Springer-Verlag Berlin, Heidelberg, 3, (2005), 808-811.
[11] Georgieva P., I. Popchev, "Fuzzy Q-measure Model for Managing Financial Investments," *Compus Rendus Acad. Bulg. Sci.* , vol. 66(5), pp. 651-658, 2013

GENERALIZED NET MODEL OF THE PROCESS OF DATAWAREHOUSE

Daniela Orozova

Abstract: *A theoretical model, describing interconnection between datawarehouse components, is proposed. The sequence of actions, referring to creation and functioning of datawarehouse, as well the information flow, is defined. The model is universal and independent of datawarehouse's problem area.*

Key words: *Knowledge Discovery in Databases, Generalized nets, Datawarehouse, OLAP tools,, Intelligent database system.*

МОДЕЛ НА ПРОЦЕСИТЕ В СКЛАД ОТ ДАННИ ЧРЕЗ СРЕДСТВАТА НА ОБОБЩЕНИТЕ МРЕЖИ*

Даниела Орозова

Абстракт: *Създаден е теоретичен модел на процесите на взаимодействие на отделните компоненти на склад от данни (Datawarehouse). Определени са последователностите от действия, свързани със създаване и функциониране на склад от данни и информационните потоци в него. Даденият подход е универсален и не зависи от конкретната предметната област на склада.*

Ключови думи: *Откриване на знания в бази от данни, Обобщени мрежи, Склад от данни, OLAP средства, Интелигентни бази от данни.*

1. Архитектура на склад от данни

Огромното количество събрани данни значително превишава възможността на човека те да бъдат ефективно използвани без помощта на специализирани мощни средства за анализ на данни. Концепцията за склад от данни се изразява в преобразуването на големи масиви от данни в значима информация, която подпомага взимането на бизнес решения. Дефиниция за понятието „склад от данни” е дадена от Уилям Инмон през 1990 г., според която: „складът от данни е предметно-ориентиран, интегриран, зависещ от времето и неизменящ се набор от данни в подкрепа на процеса на взимане на решения”.

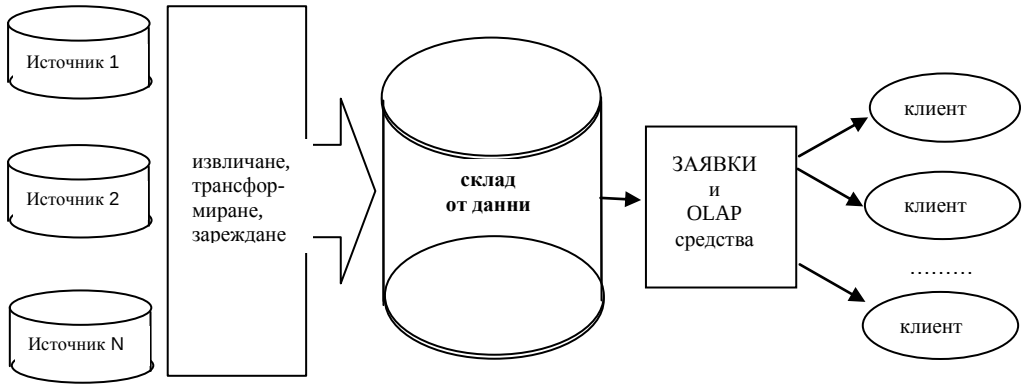
Повечето бази от данни се създават, за да се управлява реалния процес на организациите. Тези данни са **операционални**, а системите, които използват данните са **OLTP** (On-Line Transaction Processing). Системите за обработка на транзакции - складират големи количества данни и дават възможност на потребителите да достигат до данните чрез бърз, интерактивен (онлайн) достъп. От друга страна един склад с данни обикновено не съхранява текуща информация за дадена бизнес дейност, а данни удобни за колективна обработка. Той обединява данни от множество транзакционни системи, организира ги и прави данните достъпни за целите на анализа. Този тип данни

* Изследването е свързано с работата по проект Д12/21.02.2013, финансиран от фонд „Научни изследвания” при Бургаски свободен университет.

могат да се нарекат **информационни**, а системите за обработка на информационни данни - **OLAP** (On-Line Analytical Processing).

Процесът „Knowledge Discovery in Databases” (откриване на знания) включва подготовка на данните, избор на информативни признаци, пречистване на данните, приложение на метод за извличане и откриване на данните, преработка на данните и интерпретация на получените резултати [4]. Основен акцент в процеса е извличане и откриване на данните (Data Mining). **Data mining** също се използва за анализ на данните, но обхваща технологии, позволяващи да се открият неявни шаблони (образци) и зависимости в данните. Той позволява получаването на знания, като правила, описващи връзката между свойствата на данните, модели на данните, резултатите от класификацията и клъстеризацията на данните.

Схематично представяне на процеса на конструиране на склад от данни е дадено на Фигура 1. Един **склад с данни** е хранилище на информация, събрана от множество източници, съхранена под обща (единна) схема на данните и която обикновено се съхранява на едно място. Складовете с данни се конструират чрез процеси на извличане, пречистване на данните, трансформация на данните, интегриране на данните, зареждане на данните и периодично обновяване на данните. Предоставят се средства за извличане на различна информация като *OLAP* средства, софтуер за *data mining* и различни системи, подпомагащи вземането на решения



Фигура 1. Схема на процеса на конструиране на склад от данни

2. Модел на процесите в склад от данни

Обобщено-мрежовият модел [1,2,3] на процесите на взаимодействие на отделните компоненти на склад от данни е представен на Фигура 2. Моделът е изграден от 6 прехода и 20 позиции, като преходите представят описаните по-долу процеси.

Първоначално и по време на цялото функциониране на обобщено мрежовият модел в позиция l_{17} присъства α-ядро с начална и текуща характеристика:

”*склад от данни*”.

В даден момент от време α-ядрото може да се разцепи на две ядра, като оригиналното ядро остава в първоначалната си позиция през цялото време. Всички постъпващи в позиция l_{17} ядра се сливат с оригиналното α-ядро.

Входни позиции за мрежата са: l_1 , l_2 *и* l_3 .

През позиция l_1 , в модела посъпват α-ядра, с начална характеристика:

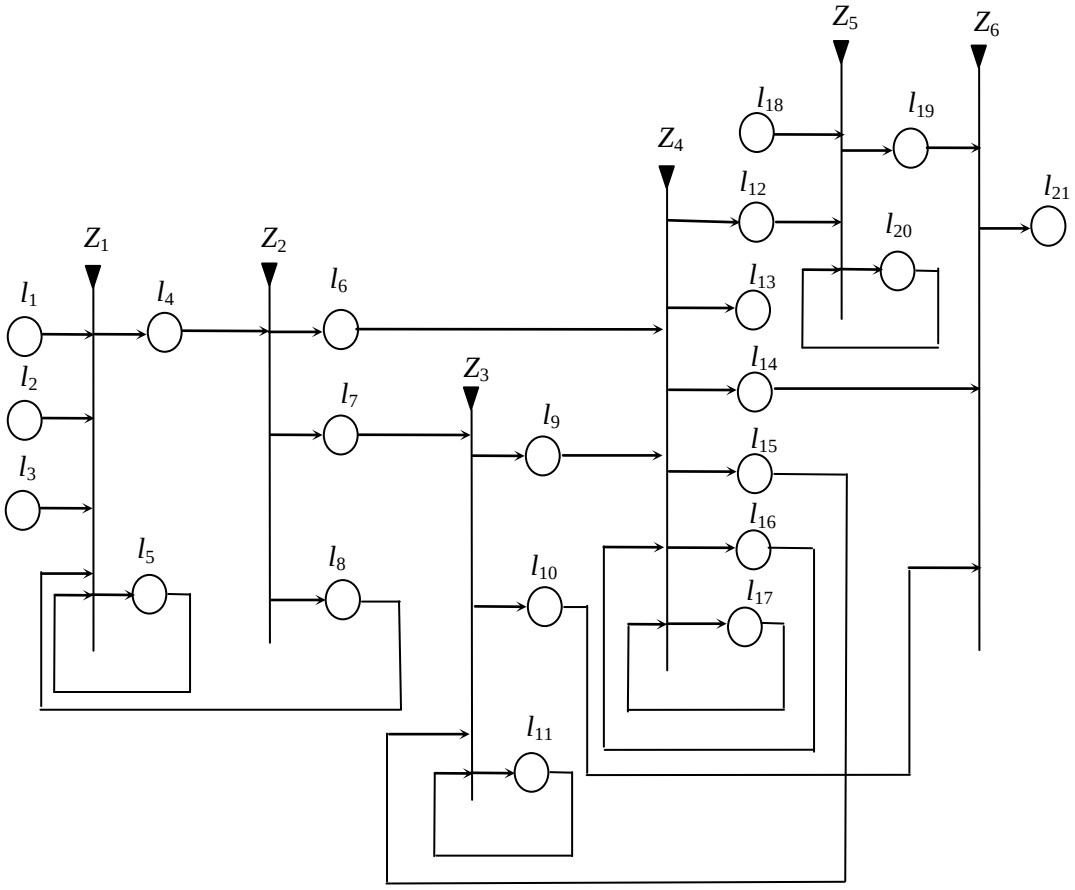
„*данни от релационни бази от данни”,*

през позиция l_2 посъпват α-ядра, с начална характеристика:

„*нерелационни данни от външни системи”,*

през позиция l_3 , посъпват α-ядра, с начална характеристика:

”*външни маркетингови данни*”



Фигура 2. Обобщено-мрежови модел на процесите на изграждане на склад от данни

Преходът Z_1 представя в модела извличането на данни от различни външни източници. От тези източници обикновено постъпват херерогенни данни от бази от данни, файлови системи, web-данни, e-mail и др. Поради това тук са необходими добре развити инструменти за достъп до различни източници от данни.

$$Z_1 = <\{ l_1 , l_2 , l_3 , l_5 , l_8 \}, \{ l_4, l_5 \}, r_1, \vee_1>,$$

$$r_1 = \begin{array}{c|cc} & l_4 & l_5 \\ \hline l_1 & F & T \\ l_2 & F & T \\ l_3 & F & T \\ l_5 & W_{5,4} & T \\ l_8 & F & T \end{array}$$

където предикатът $W_{5,4}$= “Извлечени са данни от външен източник”.

Условието за настъпване на прехода е: $\vee_1 = \vee\ (l_1 , l_2 , l_3 , l_5 , l_8)$.
След активиране на прехода в позиция l_4 постъпва α-ядро, с характеристика:

„*операционални данни”,*

в позиция l_5 , стоиа α-ядра, с характеристика:

„*архив* с *операционални данни”,*

Преходът Z_2 представя преработването на данните, които ще се съхраняват в хранилището: пречистване, филтриране и обобщаване на данните на различни нива на обобщение. Данните могат да съдържат пропуски, шум, аномални стойности, неподходящи или неизползваеми данни, да са недостатъчни и т.н. извършва се проверка и отстраняване на противоречиви или дублирани данни. От операционалните данни трябва да се създадат обощени данни – групирани по различни категории и за всяка категория се съхранят важните характеристии. Понякога се налага данните да се преформатират и декомпозират като се понижава или повишава размерността на изходното пространство, прилагайки специални алгоритми.

$$Z_2 = <\{l_4\}, \{l_6, l_7, l_8\}, r_2, \vee_2>,$$

$$r_2 = \begin{array}{c|ccc} & l_6 & l_7 & l_8 \\ \hline l_4 & W_{4,6} & W_{4,7} & W_{4,8} \end{array}$$

където:
$W_{4,6}$ = “Данните са преработени, готови за постъпване в склада от данни”,
$W_{4,7}$ = “Необходимост от обобщаване и OLAP анализ на данните ”,
$W_{4,8}$ = “Заявка за допълнително извличане на данни от външни източници”,

Условието за настъпване на прехода е: $\vee_2 = \vee (l_4)$.
α-ядрата, постъпващи в позиции l_6, l_7 и l_8 получават следните характеристики:

- в позиция l_6: “*данни за въвеждане в склада от данни”;*
- в позиция l_7 “*заявка за анализ на данните към OLAP сървъра”;*
- в позиция l_8: “*заявка за допълнително извличане на данни*”.

Преходът Z_3 представя процеса на работа на OLAP сървъра. Данните се преобразуват в подходящ формат и се подлагат на анализ. OLAP сървъра извършва сложен статистически анализ на данните в реално време. Средствата позволяват на квалифицирани потребители да анализират данните по няколко параметъра, с цел изготвяне на пазарни прогнози, планиране на технически мощности и др. Те работят с “многомерни БД”, което предполага поддържане в склада от данни на групирани структури във вид на кубове от данни. Създаването и различните операции с тези структури от данни се реализират от OLAP сървъра.

$$Z_3 = <\{l_7, l_{11}, l_{15}\}, \{l_9, l_{10}, l_{11}\}, r_3, \vee_3>,$$

$$r_3 = \begin{array}{c|ccc} & l_9 & l_{10} & l_{11} \\ \hline l_7 & F & F & T \\ l_{11} & W_{11,9} & W_{11,10} & T \\ l_{15} & F & F & T \end{array}$$

където
$W_{11,9}$ = “Изпълнена е заявка за OLAP анализ на данните, преизчислените данни са готови за запис в склада от данни”.

$W_{11,9}$ = “Изпълнена е заявка за OLAP анализ на данни на краен потребител”.

Условието за настъпване на прехода е: $\vee_3 = \wedge(\vee(l_7, l_{15}), l_{11})$.
През позиция l_7 или l_{15} в модела постъпва α- ядро с характеристика:
“*заявка за анализ на данните към OLAP сървъра”.*
α-ядрата, постъпващи в позиции l_9 и l_{10} получават характеристиките съответно:
“ резултат от OLAP анализ на данните за въвеждане в склада от данни”;
“ резултат от OLAP анализ на данните за краен потребител”.
α-ядрото в позиции l_{11} не получава нова характеристика.

Преходът Z_4 представя входно-изходния информационен поток и съхранение на данните в склада от данни. Детайлните данни могат да се обобщават на различни нива и върху тях да се извършват различни OLAP операции; върху данните могат да се прилагат различни потребителски заявки за селекция, проекция, съединение, резюмиране на данните; данните могат да се конвертират в различни формати (електронни таблици, схеми, диаграми, отчети и др.); да се пакетират и остарелите данни да се съхраняват на външни носители.

$$Z_4 = <\{l_6, l_9, l_{17}\}, \{l_{12}, l_{13}, l_{14}, l_{15}, l_{16}, l_{17}\}, r_4, \vee_4>,$$

$r_4 =$

	l_{12}	l_{13}	l_{14}	l_{15}	l_{16}	l_{17}
l_6	F	F	F	F	$W_{6,16}$	$W_{6,17}$
l_9	F	F	F	F	$W_{9,16}$	$W_{9,17}$
l_{17}	$W_{17,12}$	$W_{17,13}$	$W_{17,14}$	$W_{17,15}$	T	T

където:
$W_{6,16}$ = “Постъпили са данни за запис в склада от данни”;
$W_{6,17} = W_{6,16}$;
$W_{9,16}$ = “Постъпил е резултат от OLAP заявка за запис в склада от данни”;
$W_{9,17} = W_{9,16}$;
$W_{17,12}$ = “Заявка за Data Mining анализ върху данни от склада”;
$W_{17,13}$ = “Извършен е запис на данни от склада на външен носител”;
$W_{17,14}$ = “Заявка за извличане на данни от краен котребител”;
$W_{17,15}$ = “Заявка към OLAP сървър за анализ на данни от склада”;

Условието за настъпване на прехода е: $\vee_4 = \wedge(\vee(l_6, l_9), l_{17})$.
След активиране на прехода:
α-ядрата, постъпващи в позиции l_{12} получават характеристика:
”данни от склада. заявка за Data Mining анализ”.
α-ядрата, в позиции l_{13} получават характеристика:
“ данните от склада, записани на външен носител”;
α-ядрата, в позиции l_{14} получават характеристика:
“данни от склада за предоставяне на краен потребител”;
α-ядрата, в позиции l_{15} получават характеристика:
“ данни от склада. заявка за OLAP анализ на данните”;
α-ядрата, постъпващи в позиции l_{16} се сливат с α-ядрото, което цикли в тази позизия по време на цялото функциониране на мрежата с характеристика:
"Хранилище на метаданни”;

α-ядрата, постъпващи в позиции l_{17} се сливат с α-ядрото, което цикли в тази позизия по време на цялото функциониране на мрежата с характеристика:

"”Склад от данни”.

Преходът Z_5 представя Data Mining анализи, прилагани върху данните от склада. Прилагат се различни видове анализ като: класификационен анализ, регресионен анализ, характеристичен анализ, клъстерен анализ, еволюционен анализ и др. За различните цели на анализа могат да се прилагат различни техники и средства за откриване на закономерности върху данните [4], например: невронни мрежи, дървета на решенията, алгоритми за клъстеризация, установяване на асоциации и т.н. Откритите закономерности се оценяват с помощта на различни коефициенти за измерване на приложимостта им и в резултат могат да се определят като знания.

През позиции l_{18} постъпва β-ядра с начална характеристика:

”*Data Mining средства*”.

Условието за настъпване на прехода е: $\vee_5 = \wedge\ (l_{12}, l_{18})$.
След активиране на прехода α-ядрото от позиции l_{12} се слива с β-ядро от позиции l_{18} и постъпва в позиция l_{20} за прилагане на избрано Data Mining средство (средство за извличане на знания от данните).

$$Z_5 = <\{l_{12}, l_{18}, l_{20}\}, \{l_{19}, l_{20}\}, r_5, \vee_5>,$$

$r_5 =$

	l_{19}	l_{20}
l_{12}	F	T
l_{18}	F	T
l_{20}	$W_{20,19}$	T

където:
$W_{20,19}$ = “Получени са нови знания, т.е. извлечени са модели (шаблони) на данните”.

α-ядрото, постъпващо в позиция l_{19} получава характеристика:

“ модели (шаблони) на данните”.

β-ядрото цикли в позиция l_{20} с текуща характеристика:

“ Data Mining (алгоритми) средства ”.

Интерпретацията на прехода Z_6 е представяне на получените резултати и знания на крайни потребители или бизнес приложенията. Тук се включва и визуализация на информацията с цел потребителят да разбере и интерпретира резултатите. Получената информация може да подпомага взимането на неструктурирани, стратегически решения.

$$Z_6 = <\{l_{10}, l_{13}, l_{14}\}, \{ l_{21}\}, r_6, \vee_6>,$$

$r_6 =$

	l_{21}
l_{10}	$W_{10,21}$
l_{13}	$W_{13,21}$
l_{14}	$W_{14,21}$

където
$W_{10,21}$ = “Данни, получени в резултат на OLAP анализ”;
$W_{13,21}$ = “Данни, получени в резултат на заявка към склада от данни”;

$W_{10,21}$ = “Знания, изведени в резултат на заявка към Data Mining средства”.

Условието за активиране на прехода е: $V_6 = \vee$ (l_{10}, l_{13}, l_{14}).

α_{13}-ядрото, постъпващо в позиция l_{21} получава характеристика

“*Резултат от потребителска заявка*”.

Прилагайки йерархичен оператор H_3 от теорията на обобщените мрежи [2], представения ОМ модел от фигура 2 може да бъде детайлизиран и разширен. Следващият модел, представен на фигура 3 може да се разглежда като подмрежа, заменяща позиция l_{20} в модела на функциониране на склад от данни.

Следва кратко представяне на ОМ модел на процеса на прилагане на Data Mining средства, подробно представен в [6]. Подмрежата съдържа 5 прехода и 21 позиции, групирани в две групи и свързани с два типа ядра, които ще постъпват в съответните типове позиции: α-ядра и l-позиции представят процеса на прилагане на Data Mining средства, β-ядра и t-позиции представят критериите за ограничаване на средствата и избор на подходящи Data Mining средства.

За краткост ще се използва означението α- и β-ядра вместо α_i- и β_j-ядра, където i, j са номерата на съответните ядра.

Първоначално едно β_0-ядро стои в позиция t_6 с начална характеристика

„*налични Data Mining средства*“.

На следващия преход от функционирането на мрежата β-ядрото се разделя на две. Оригиналното β-ядро ще продължи да стои в позиция t_6, докато другото β-ядро ще се придвижи към прехода Z_5, преминавайки през прехода Z_2.

Ядрата α_0 и α_1, постъпващи в мрежата през позиции l_0 и l_1, получават характеристики съответно: „*начални хипотези*“ и „*начални данни*“.

Ядрата β_1 и β_2 постъпват в мрежата през позиции t_0 и t_1 съответно. Тези ядра получават съответно начални характеристики:

„*нова Data Mining техника* (*средство*)“;

„*критерии за избор на Data Mining средства*“.

Следва кратко описание на отделните преходи.

$Z_1 = \langle\{l_0, l_1, l_{10}, t_2\}, \{l_2, l_3, l_4\}, r_1, \vee(\wedge(\, l_0, l_1), l_{10}, t_2)\rangle$;

$r_1 =$

	l_2	l_3	l_4
l_0	*False*	*False*	*True*
l_1	*False*	*False*	*True*
l_{10}	$W_{10,2}$	$W_{10,3}$	*False*
t_2	*False*	*True*	*False*

,

където:

$W_{10,2}$ = „Data Mining техниката е приложена“;

$W_{10,3} = \neg\, W_{10,2}$.

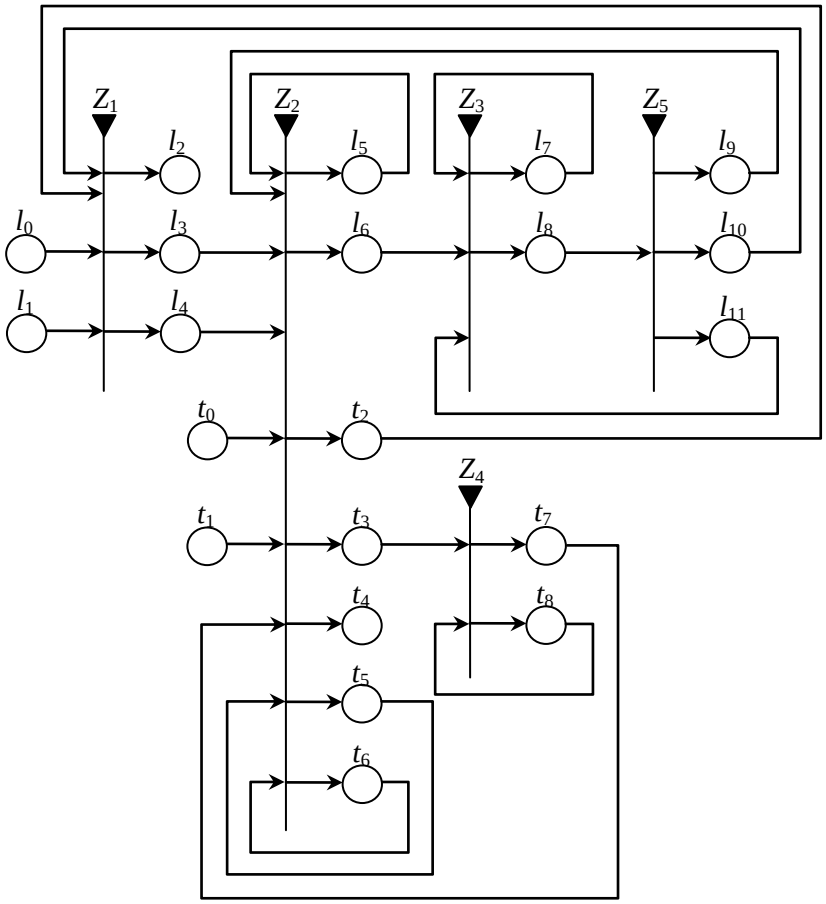
Като първо действие на прехода Z_1 α_0- и α_1-ядрата, които постъпват през позиция l_4 (от позиции l_0 и l_1), се сливат в едно ново α-ядро с характеристика

„*начални хипотези, начални данни*“.

Характеристиката на β-ядрото, което постъпва в позиция l_3 (от позиция t_2), получава характеристиката:

„*цел, Data Mining техника*“.

При следващото активиране на прехода Z_1 β-ядрата могат да постъпят в позиция l_2 или l_3. β-ядрото, което постъпва в позиция l_2, не получава нова характеристика.



Фигура 3. ОМ модел на процеса на прилагане на Data Mining средства

$Z_2 = \langle\{l_3, l_4, l_5, l_9, t_0, t_5, t_6, t_7, t_1\}, \{l_5, l_6, t_2, t_3, t_4, t_5, t_6\}, r_2, \vee(\, l_3, l_4, l_5, l_9, t_0, \wedge(\, t_1, t_5), t_6, t_7)\rangle$;

$r_5 =$

	l_5	l_6	t_2	t_3	t_4	t_5	t_6
l_3	*True*	*False*	*False*	*False*	*False*	*False*	*False*
l_4	*False*	*False*	*False*	*False*	*False*	*False*	*True*
l_5	$W_{5,5}$	$W_{5,6}$	*False*	*False*	*False*	*False*	*False*
l_9	*True*	*False*	*False*	*False*	*False*	*False*	*False*
t_0	*False*	*False*	*False*	*False*	*False*	*False*	*True*
t_5	*False*	*False*	*False*	*True*	*False*	*False*	*False*
t_6	*False*	*False*	*False*	*False*	$W_{6,4}$	$W_{6,5}$	*True*
t_7	*False*	*False*	*True*	*False*	*False*	*False*	*False*
t_1	*False*	*False*	*False*	*True*	*False*	*False*	*False*

където:

$W_{5,5}$ = „има Data Mining техника, която още не е приложена“;

$W_{5,6}$ = „Data Mining техниката е извлечена“;

$W_{6,4}$ = „Data Mining техниката е отхвърлена“;

$W_{6,5}$ = „Data Mining техниката е избрана“.

В позиция l_3 и l_4 α-ядрата получават характеристики съответно

„*избрани подходящи Data Mining техники, критерии за избор на Data Mining техники*“;

„*отхвърлена Data Mining техника*“.

β_1-ядрото, което постъпва в прехода Z_2 (от позиция t_0), ще се слее с оригиналното β_0-ядро, което стои в позиция t_6. β_3-ядрото, което постъпва в позиция t_5, получава характеристика

„*избрани Data Mining техники*“.

β-ядрата, които постъпват в позиции t_2 (от позиция t_7) и t_3 (от позиция t_5), не получават нова характеристика.

$$Z_3 = \langle\{l_6, l_7, l_{11}\}, \{l_7, l_8\}, r_3, \vee(l_6, l_7, l_{11})\rangle;$$

$$r_3 = \begin{array}{c|cc} & l_7 & l_8 \\ \hline l_6 & True & False \\ l_7 & W_{7,7} & W_{7,8} \\ l_{11} & True & False \end{array},$$

където:

$W_{7,7}$ = „има следващи стъпки от работата на текущата Data Mining техника“;

$W_{7,8}$ = „избрана е следващата стъпка от работата на текущата Data Mining техника“.

α-ядрото, което постъпва в позиция l_7, не получава нова характеристика, докато α-ядрото, което постъпва в позиция l_8, получава характеристиката

„*текуща стъпка от прилагането на избраната Data Mining техника*”

$$Z_4 = \langle\{t_3, t_8\}, \{t_7, t_8\}, r_4\rangle;$$

$$r_4 = \begin{array}{c|cc} & t_7 & t_8 \\ \hline t_3 & False & True \\ t_8 & W_{8,7} & W_{8,8} \end{array},$$

където:

$W_{8,7}$ = „избрана е Data Mining техника за поставената цел“;

$W_{8,8} = \neg W_{8,7}$.

β-ядрата, които постъпват в позиция t_8, не получават нова характеристика, докато β-ядрата, които постъпват в позиция t_7, получават характеристиката

„*цел, избрани Data Mining техники*“.

$$Z_5 = \langle\{l_8\}, \{l_9, l_{10}, l_{11}\}, r_5\rangle;$$

$$r_5 = \begin{array}{c|ccc} & l_9 & l_{10} & l_{11} \\ \hline l_8 & W_{8,9} & W_{8,10} & W_{8,11} \end{array},$$

където:

$W_{8,9}$ = „има следваща Data Mining техника, която може да бъде прилагана“;

$W_{8,10}$ = „последната възможна Data Mining техника е приложена“;

$W_{8,11}$ = „има следваща стъпка от текущата Data Mining техника, която ще се изпълнява”.

α-ядрата, които постъпват в позиции l_9 и l_{11}, не получават нови характеристики, докато α-ядрата, постъпващи в позиция l_{10}, получават характеристиката

„*цел, Data Mining средство, оценка от работата на Data Mining средството*“.

3. Приложения

Създаденият обобщено-мрежови модел показва връзките, начина на взаимодействие и процесите между отделните компоненти в склад от данни. Поетапно се проследяват информационните потоци в склада: вливане на данните, получавани от

различни хетерогенни източници, тяхното съхранение, описание и метаданни, архивиране, анализ и предоставяне на данни и знания на крайните потребителите или бизнес приложения. Моделът може да бъде детайлизиран чрез прилагане на йерархичен оператор H_3, който заменя преход или позиция от мрежата с нова мрежа, описваща по-детайлно съответен процес. Такъв модел намира приложение при създаване, изследване и оптимизиране на специализиран софтуер за работа със складове от данни.

Литература:

[1] Atanassov, K., Generalized Nets, World Scientific, Singapore, 1991.
[2] Atanassov, K. On Generalized Nets Theory. Prof. M. Drinov Academic Publ. House, Sofia, 2007.
[3] Atanassov, K., H. Aladjov. Generalized Nets in Artificial Intelligence. Vol.2: Generalized Nets and Machine Learning. “Prof. M. Drinov” Academic Publishing House, Sofia, 2001.
[4] Fayyad, Piatetsky-Shapiro, Smyth, and Uthurusamy, Advances in Knowledge Discovery and Data Mining, AAAI/MIT, 1996.
[5] Shannon, A., E.Sotirova, D.Orozova, Geneneralized Net Model of Using Data Mining Techniques for Process of Undergraduate Matriculation in a Digital University, Eleventh Int. Workshop on GNs and Second Int. Workshop on GNs, IFSs, KELondon, 9-10 July 2010, 1-6.
[6] Sotirova, E., D. Orozova, Generalized Net Model of the Phases of the Data Mining Process, Developments in Fuzzy Sets, Intuitionistic Fuzzy Sets, Generalized Nets and Related Topics, Volume II: Applications, IBS PAN – SRI PAS, Polish Academy of Sciences, Warsaw, 2010, pp.247-260.
[7] Sumathi, S., S.N. Sivanandam, Introduction to Data Mining Principles and its Applications, Studies in Computational Intelligence, Springer, Vol. 29, 2006.

MECHANISMS AND MACHINERY USED IN UTILIZATION OF RENEWABLE ENERGY SOURCES

Dolchinkov Radostin, Burgas Free Univercity, rado@bfu.bg

Abstract: The paper reviews the mechanisms and machinery used in utilization of renewable energy sources. In the overview on wind turbines their applicability as alternative energy sources is shown. The mechanisms constitute the kinematic basis of machinery and mechanical appliances, and so they represent an integral part of them. The functionality of the entire machine or device as a whole substantially depends on the correctness of the operation of its mechanisms.

Keywords*:* knife, gears, screw mechanisms, reducer, multiplier.

МЕХАНИЗМИ И МАШИНИ ВЪВ ВЕИ

Доц. д-р Радостин Долчинков, Бургаски свободен университет, rado@bfu.bg

Резюме: В работата са разгледани механизмите и машините използвани във възобновяеми енергийни източници. С направеният обзор върху ветрогенераторите се показва тяхната приложимост като алтернативни източници на енергия. Механизмите съставят кинематичната основа на машините и механичните уреди и затова те се явяват неотменна тяхна съставна част. Работоспособността на цялата машина или уред като цяло в значителна степен зависи от правилността на работа на техните механизми.

Ключови думи: лостови, зъбни и винтови механизми, редуктор, мултипликатор.

Ветрените технологии използват енергията на въздушните маси над земната повърхност, които са резултат от движението, предизвикано от топлината на Слънцето и движението на Земята. Въздухът задвижва перките на енергийното съоръжение, монтирани на ротор в резултат на силата, която се създава от разликата в наляганията между високото налягане върху плоската повърхност на перките и ниското налягане на обратната им страна. Въртенето им води до директно производство на механична енергия, която може да се превърне в електрическа с помощта на електрогенератор. Най-общо казано, вятърната турбина е уред, който превръща вятърната кинетична енергия в електрическа.

В работата са разгледани механизмите и машините използвани във възобновяеми енергийни източници. С направеният обзор върху ветрогенераторите се показва тяхната приложимост като алтернативни източници на енергия.

В машините използвани във ВЕИ се осъществява работен процес, при който се преобразува механичната енергия и нейното предаване. Предаването на механичната енергия се осъществява чрез съвкупност от подвижно свързани тела, които образуват

механизъм. В уредите движението на подвижните части на чувствителния елемент трябва да се преобразува и предаде на изпълнителното устройство. Това също се осъществява чрез някакъв механизъм, който служи за предаване и мащабно преобразуване на движението и силите.

Механизмите във ВЕИ изпълняват следните задачи:

- предаване на механично преместване от източника на движение към местата и детайлите, където то се реализира;
- силите на източника на движение се предават и се преобразуват в сили и моменти в крайните пунктове за изпълнение на механични операции;
- извършване на изменение на скоростите и преместванията;
- изпълняват функционално преобразуване на механичното движение, т.е. осъществяват преобразуване на движението на телата, извършващо се по един закон, в движение на друго тяло, извършващо се по друг зададен закон.

Цел на преобразуване на движението може да бъде достигането на по-високи или по-малки скорости, увеличаването или намаляването на преместванията, получаването на зададени траектории на движение на отделни елементи на механизма или зададени закони на изменение на преместванията във времето.

Преобразуването на въртеливото движение по механичен начин при предаването му към изпълнителните органи на машината или уреда, се нарича механична предавка.

Механизмите съставят кинематичната основа на машините и механичните уреди и затова те се явяват неотменна тяхна съставна част. Работоспособността на цялата машина или уред като цяло в значителна степен зависи от правилността на работа на техните механизми.

Механизмите, образуващи механичната система, могат да се свързват последователно на принципа „изход – вход” или „изход – стойка” или успоредно.

Входните звена на механизмите могат да се съединят с изходните звена на двигателите или с изходните звена на други механизми. В последния случай се образуват нови механизми, имащи по-сложна структура. Такова последователно свързване на механизмите е на принципа „изход – вход”. Тази схема на свързване е широко разпространена при ветрогенераторите, в които перките на ротора извършват въртеливо движение.

Във ВЕИ се използват разнообразни механизми, които могат да се класифицират по конструктивни признаци, към следните групи:

- лостови;
- зъбни;
- винтови.

Може да бъде направена и функционална класификация, която се основава на функцията, която изпълнява механизмът, а именно:

- механизми на двигатели и преобразуватели;
- предавателни механизми;
- изпълнителни механизми;
- механизми за контрол, управление и регулиране;

Механизмите на двигателите осъществяват преобразуване на вятърната енергия в механична, а механизмите на преобразувателите преобразуват механичната енергия в електроенергия.

Предавателните механизми имат за задача да предават движението от ротора на ветрогенератора към електрогенератора. Тъй като вала на ротора има обикновено по-малък брой обороти, отколкото основния вал на електрогенератора, задачата на предавателните механизми се явява увеличаване на оборотите на вала.

Механизми за управление, контрол и регулиране при ветрогенераторите представляват различни устройства за контрол на ъгловата скорост, устройства следящи посоката и скоростта на вятъра, измерителните механизми за контрол на скоростта на вятъра, налягането и др.

Друг признак по който може да се направи класификация на механизмите използвани във ВЕИ е според движението и разположението на неговите елементи в пространството. Те биват: пространствени, равнинни и сферични.

Според вида на преобразуване на движенията механизмите се класифицират на:

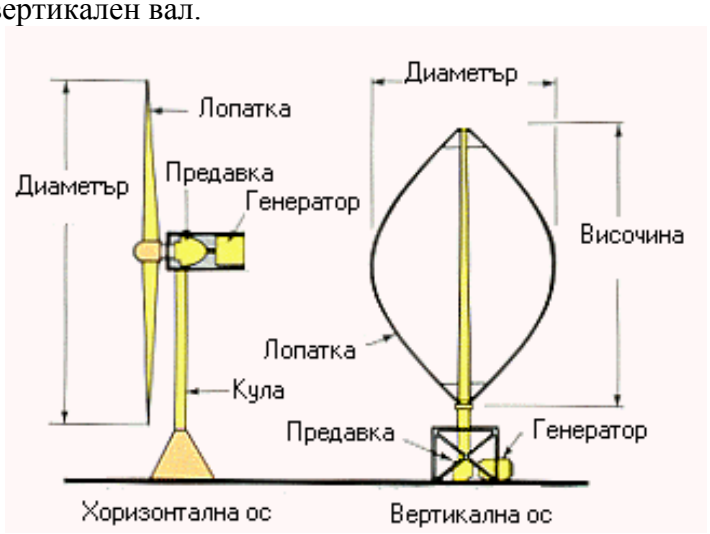
- въртеливо движение във въртеливо движение – редуктори $\omega_{вх} > \omega_{изх}$; мултипликатори $\omega_{вх} < \omega_{изх}$; съединители $\omega_{вх} = \omega_{изх}$;
- въртеливо движение в постъпателно движение и обратно.

Във ВЕИ се използват и понижаващи механизми, което позволява да се използват бързоходни, а оттам, малогабаритни и по-евтини електродвигатели при малки скорости на движение на изпълнителните органи на работните машини и уредите.

Повечето съвременни вятърни турбини работят с променливи обороти на ротора. Променливата скорост предлага редица предимства, сред които подобряване на съвместимостта им с електроенергийната система, редукция на товара, енергоспестяване и др. Работата на вятърната турбина с променливи обороти на въртене на ротора би могла да се реализира чрез широко разнообразие от технически решения. При традиционно използваната концепция за работа на турбината с променлива скорост се използва зъбна предавка, като генераторът се свързва към електрическата мрежа през електрически преобразувател.

В експлоатация се срещат най-различни видове вятърни електрогенератори (фиг.1), които най-често се разделят на два основни вида:

- с хоризонтален вал;
- с вертикален вал.



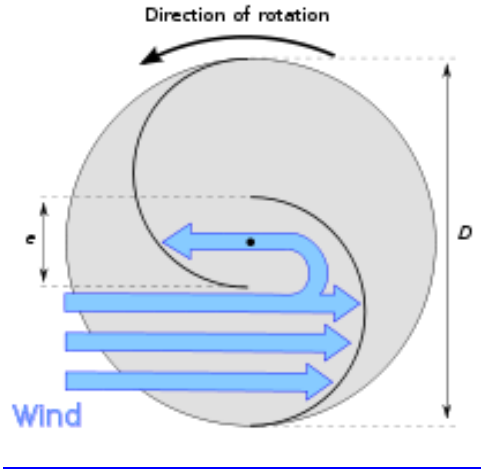
Фиг.1. Видове вятърни турбини

При вятърните централи с вертикален вал роторите са три основни типа: Савониос, Дариус и Н-ротор (фиг.2). Те имат ниска ефективност, но работят при по-ниски ветроскорости, по-лесни са за обслужване и не изискват устройства за насочване.



Фиг.2. Вятърни централи с вертикален вал

Ветроагрегатите тип Савониюс са предназначени за по-малки мощности. Конструкцията се състои от две полу-цилиндрични работни повърхнини, свързани към вертикална ос. Двете повърхнини са радиално разместени така, че да се образуват два отвора. Вятърът влиза в единия от двата отвора и оказва натиск върху вътрешната част на едната работна повърхност, като предизвиква нейното завъртане, а съответно и на вала, към който тя е свързана. След това вятърът се насочва към вътрешната стена на втората повърхност (фиг.3), оказва натиск върху нея, завърта я и излиза през втория отвор.



Фиг.3. Принцип на работа на Савониова вятърна турбина

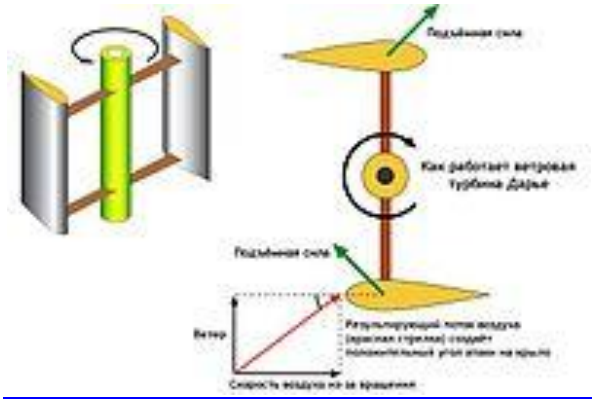
Савониовата турбина е вятърна турбина, работеща на принципа на съпротивлението. Турбината използва различния коефициент на съпротивлението на протичащия флуид, действащо на изпъкнала или вдлъбната площ.

Вътрешните краища на лопатките достигат до средата на ротора и по този начин позволяват протичането на флуида между техните задни стени. Оста на въртене е перпендикулярна на посоката на флуида.

Савониовата турбина има малък коефициент на полезно действие (при водните турбини е само около 15 до 20%). Нейното главно предимство е простата конструкция. Когато Савониовата турбина е поставена с вертикална ос на въртене, работи независимо от посоката на вятъра.

Недостатък на двулопатковите Савониови турбини е присъствието на така наречения „мъртъв ъгъл“. Този недостатък се избягва чрез свързването на няколко ротора с различно насочени лопатки или с винтова форма на лопатките.

Ветроагрегатите тип Дарио могат да се задвижват от вятър, който идва от различни посоки и нямат нужда да бъдат ориентирани (фиг.4). Съществуват конструкции с ротор, който да се адаптира към изменение на височината според скоростта на вятъра. При силни ветрове долната част на ротора се повдига, а горната се спуска, с което височината намалява, но работната площ не се променя. Най-голямата вятърна ЕЦ с вертикален вал е Cap Chat - Quebec и е с височина 110 m.



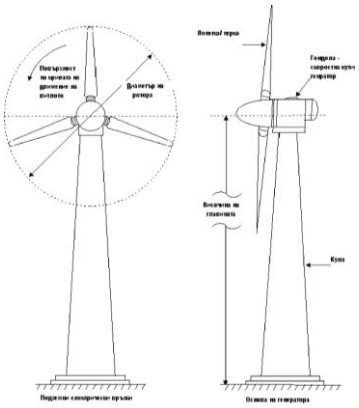
Фиг.4. Принцип на работа на Дариус вятърна турбина

Турбина на Дариус, Дариусов ротор или Мотор на Дариус е вятърна турбина, работеща на принципа на подемната сила. За разлика от другите вятърни турбини оста на тази турбина е перпендикулярна на посоката на вятъра и турбината обикиновено се монтира вертикално. Това прави турбината на Дариус да е независима от посоката на вятъра.
Турбината на Дариус е била изобретена и патентована от френския инженер Жорж Жан Мари Дариус. Турбината била снабдена с устройство за стартиране и регулация. Коефициентът на полезно действие (КПД) на този тип вятърни турбини е около 35 - 38%, т.е. е малко по-висок от Савониовата турбина. Предимствата на този тип турбини са: не е необходимо да бъде насочвана срещу вятъра; генераторът и останалите части могат да се разположат под стълба на турбината; по-малки изисквания към стълба на турбината (тегло, балансиране). Недостатъци на този тип турбини са: трудно регулиране, по-висока скорост на вятъра за стартиране.
Ветроагрегатите Н-тип са много слабо разпространени, въпреки голямото разнообразие на експериментални конструкции.
По данни от пазара, вятърните ЕЦ с хоризонтален вал в употреба са 91%, а с вертикален вал - 9% [4].
При вятърните електрогенератори с хоризонтална ос на въртене генераторът и ветротурбината са монтирани на общ хоризонтален вал. Роторите са с една, две, три или много лопатки (фиг.5). Мощността на вятърната турбина не зависи от броя на лопатките на ротора, а от работната площ, която те описват при своето въртене.
Вятърните електрогенератори с хоризонтална ос на въртене са с по-висока ефективност. При тях генераторът и въртящият се вал са поставени върху висока кула и се насочват по посока на вятъра по различни начини. Витлата на съвременните вятърни генератори имат аеродинамичен профил, подобен на самолетно крило - подемната сила е резултат от разлика в налягането, която пък е разлика между скоростта на вятъра под и над перката. Най-разпространеният тип генератори е с три перки (наричани още „Пропелерен тип“) (фиг.6). Отличават се с висока ефективност, слаба промяна във въртящия момент и висока надеждност. Стартовият въртящият момент е малък. Затова този тип турбини работят само при скорости на вятъра над 5 m/s.

Тенденцията е лопатките да са по-малко, но по-дълги, за да нараства работната повърхност и съответно мощността. Преобладаващата част от съвременните вятърни електрогенератори са с две или три лопатки и съвсем рядко - с една. Ветроагрегатите с много лопатки се използват предимно за изпомпване на вода.

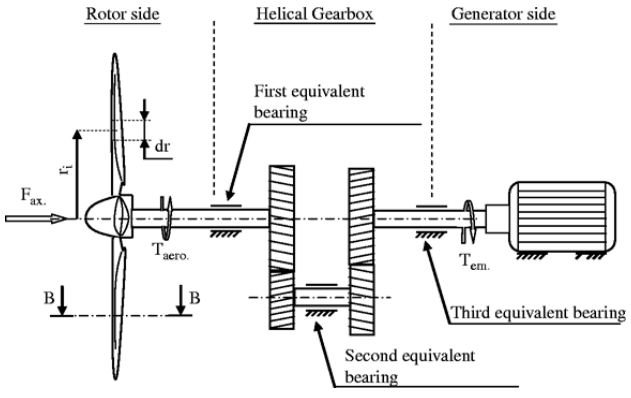


Фиг.5. Вятърните електрогенератори с хоризонтална ос на въртене



Фиг.6. Вятърни електрогенератори „пропелерен тип”

Роторът се състои от хъб, три витла и система за стъпково регулиране на ъгъла на атака на всяко витло спрямо въздушния поток, като всички компоненти са разположени насрещно на вятъра. Те са със специална аеродинамична форма, за да могат да създават и използват подемната сила на въздушното течение. Механичната мощност на роторните витла се подава към генератора посредством трансмисионната система. Тя се състои от кутия с предавателен блок от зъбни колела, блокираща система, както и от спомагателни смазваща и охлаждаща системи. Кинематична схема на конструкцията е показана на фиг.7.

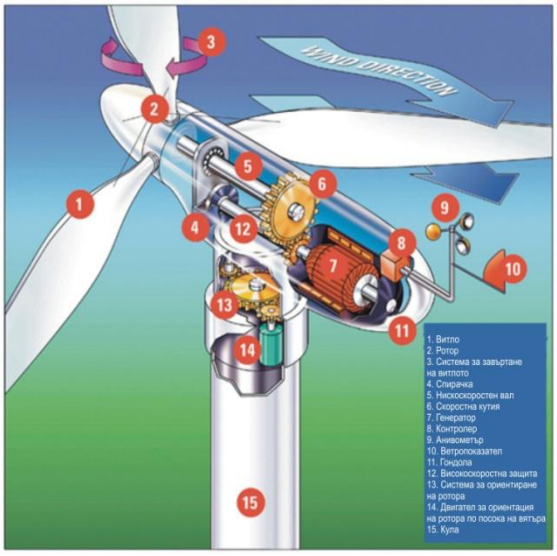


Фиг.7. Кинематична схема на ветрогенератор с хоризонтална ос на въртене

Предавателния блок от зъбни колела осъществява преобразуване на оборотите. Блокиращата система е разработена да “заключва” генератора, когато турбината е спряна. Въртящата система обръща корпуса (кутията) на ротора по направлението на „вятърната атака”, използвайки задвижващ и зъбен механизъм. Микропроцесорна система следи и контролира състоянието на системите на вятърния генератор. Регулиращите системи са разработени за дистанционно обслужване от станция посредством оптични влакна.

Вятърните турбини могат да бъдат използвани самостоятелно или да бъдат свързани към електрическа мрежа, или дори комбинирани с клетки за събиране на слънчева енергия. Те се монтират върху кула. В повечето случаи вятърът е толкова по-силен и постоянен и по-малко завихрен, колкото по-голяма е височината, до която достига кулата. Самостоятелните турбини обикновено се използват за изпомпване на вода или комуникации, но домакинствата и отдалечените селища намиращи се в по-ветровити зони могат да ги използват за да генерират електрическа енергия.

На фиг. 8. са показани механизмите и машините използвани във ветрогенераторите. Системата се състои от:



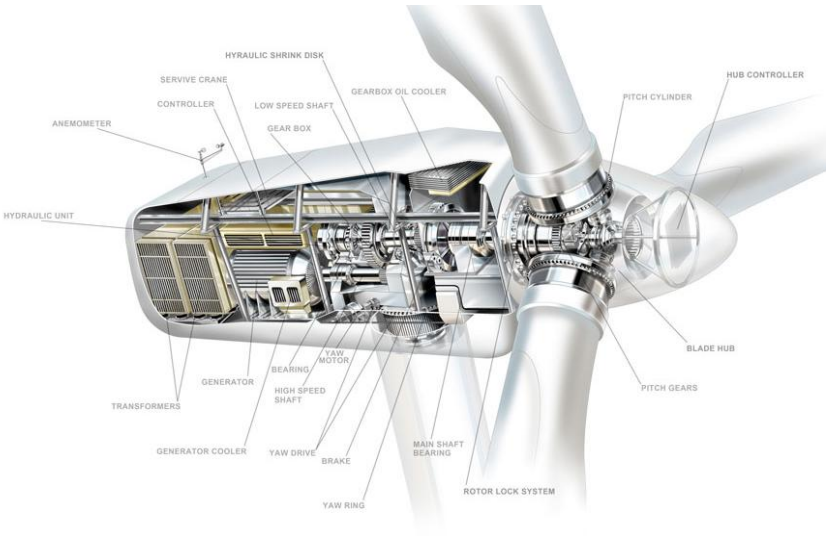
Фиг.8. Елементи на ветрогенератора

- **Гондола** - съдържа най – важните елементи на вятърната турбина (фиг.9.) – скоростна кутия /трансмисия/, спирачки, електрически генератор, система за ориентиране на ротора, хидравлична система, охладителна система. Гондолата се завърта, следвайки посоката на вятъра и по този начин оползотворява максимално количество енергия.

- **Ротор** - улавя вятъра, чрез витлата и предава механичната енергия към главината на ротора.

- **Главина** - присъединява ротора чрез ниско скоростен вал към вятърната турбина.

- **Ниско скоростен вал** (фиг.10.) - свързва ротора към трансмисията. Поместен е във вътрешността на гондолата. Вятърната енергия се превръща в механична енергия на движението.



Фиг.9. Гондола на ветрогенератор



Фиг.10. Ниско скоростен вал

- **Скоростна кутия /трансмисия/** - свързана е към ниско скоростния вал (фиг.11.), предава въртящият момент на високо скоростен вал. Трансмисията представлява мултипликатор, чието предназначение е да увеличава (мултиплицира) оборотите на въртене на високо скоростният вал (вала на генератора), намалявайки въртящия момент без значителни загуби на мощност. Поради променливата скорост на вятъра, ветрогенератора без мултипликатор на практика е неизползваем. При ниски скорости на вятъра максимумът на въртящия му момент е нужен на ниски обороти, а при висока скорост на вятъра - на високи. Именно този недостатък компенсира скоростната кутия, монтирана във ветрогенератора, която изменя предаването на въртящия момент в зависимост текущата необходимост и увеличава оборотите. Механичната част от конструкцията се състои от комплект планетарни предавки, управлявани от лентови спирачки посредством хидравлични клапани. Съединител липсва, но в някои случаи се налага отделяне на кутията от ниско скоростният вал. Със задачата се справя хидротрансформатор. Трансмисията работи бързо и плавно. Основното достойнство на мултипликатора е идеалната приспособимост към условията на работа в зависимост от скоростта на вятъра. Предавателното число се променя плавно и на практика броят на предавките е безкраен, като на изхода от кутията се поддържат необходимите обороти за правилната работа на генератора. Съществен

недостатък на трансмисията е склонността и към прегряване, тъй като функционира за сметка на силите на триене, което налага задължително и изграждането на охладителна система.



Фиг.11. Скоростна кутия (мултипликатор)

По долу на фиг.12 са показани различни типове мултипликатори с предавателно число от 40:1 до 136:1.



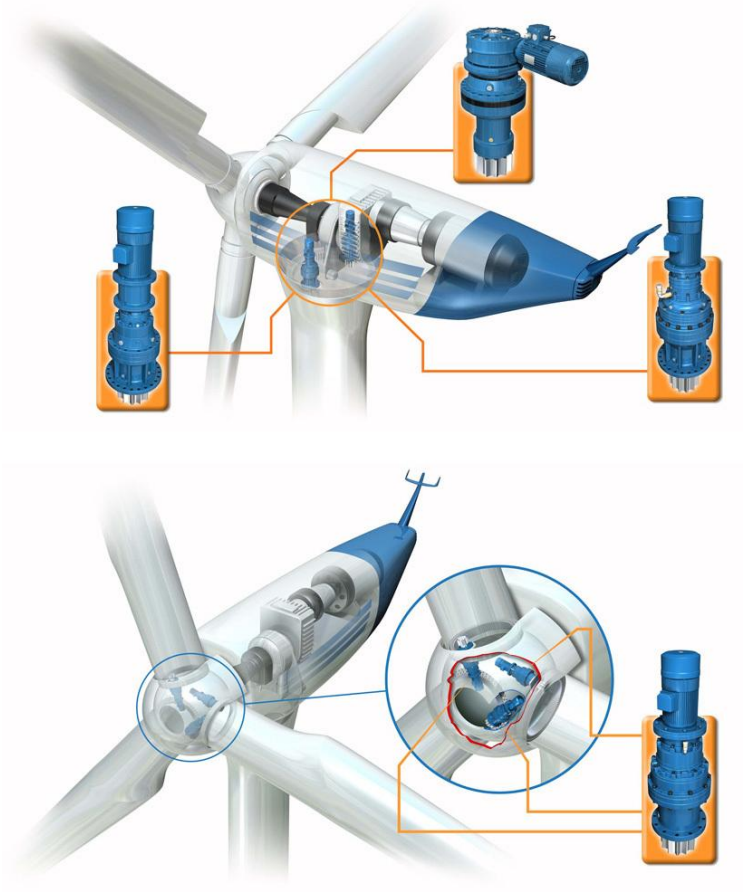
Фиг. 12. Мултипликатори

- **Високо скоростен вал с механична спирачка** - задвижва електрическия генератор с около 1500 обр/мин. Механичните спирачки се използват за предотвратяване на аеродинамичните пробиви, при високи скорости на вятъра или спиране на съоръжението.

- **Електрически генератор** - Обикновено е синхронен или асинхронен с генератор максимална електрическа мощност от 500 kW до 7000 kW.

- **Система за ориентиране на ротора** - насочва гондолата и ротора в посоката на вятъра използвайки електрическо или друг вид задвижване.

Използват се мотор-редукторни групи (фиг.13) при които електродвигателят и редукторът са непосредствено свързани и представляват едно изделие. По този начин се избягва използването на съединител между тях, какъвто е необходим при обикновените редуктори. Отличават се с намалени габаритни размери, висок КПД, опростен монтаж в различни пространствени положения, просто обслужване.
Използваните при ветрогенераторите, според кинематичната схема мотор- редуктори се поразделят на обикновени съосни и планетни. В зависимост от вида на използваните предавки, с които е изпълнен редукторът, се различават съответно цилиндрични, конусни, червячни, и комбинирани редуктори.



Фиг.13. Система за ориентиране на ротора и витлата

Цилиндричните редуктори могат да бъдат изпълнени с цилиндрични зъбни колела с прави, наклонени или шевронни зъби; конусните - с конусни зъбни колела с прави и криволинейни зъби, а червячните - с червячна предавка. Възможно е редукторът да е изпълнен от последователно свързани различни по вид предавки - цилиндрични, конусни или червячни. Обикновено за задвижване на мотор редукторите се използват електродвигатели с фланец или на лапи с или без вградена електромагнитна спирачка и принудително охлаждане, стъпкови мотори, както и такива с вградени честотни регулатори (инвертори), като възможното регулиране на оборотите е 1:5 и 1:10.
От качествата и надеждността на електромеханичните задвижвания в най-голяма степен зависят и технико-икономическите показатели на ветрогенераторите.

Развитието на съвременната ветроенергетика определя няколко основни направления в развитието на мотор редукторите използвани във ветрогенератори:

а/ компановка на електрозадвижването от отделни нормализирани и стандартизирани възли – различни типове редуктори с многоскоростни или

регулируеми електродвигатели и системи за управлението им. Създаването на задвижвания от готови групи или блокове, което спомага за повишаване на надеждността, съкращаване на времето за разработка, монтаж и ремонт;

б/ намаляване на относителното тегло на редукторите, с което се постига икономия на метал и се облекчава обслужването и транспорта им. Това е възможно при използването на материали с високи якостни качества, оптимизирана форма на детайлите, използването на нови технологии и на съвременни методи за термо-химична обработка на елементите (например йонно азотиране, карбонитриране и др.), на прецизни методи на изпитване и други;

в/ използване на усъвършенствани планетни и нови видове механични предавки - вълнови, циклоидни, хипоидни, хиперболоидни, тип Spiroid, тип Helicon, тип Wildhaber, тип Twinspin и други.

При планетните мотор-редуктори се използват различни структурни схеми на епициклични зъбни механизми.

Хиперболоидните предавки са неортогонални високоредуциращи зъбни механизми. Те представляват пространствени зацепвания на различни типове зъбни колела: тип Spiroid е с зацепване на колело с конусен червяк, тип Helicon – на цилиндричен червяк с конусно колело, тип Wildhaber – на цилиндрично колело с глобоиден червяк.

Към основните типове зъбни механизми могат да се отнесат:

Съосен цилиндричен мотор редуктор. При тази схема оста на изходящия вал съвпада с оста на вала на двигателя. Редукторите са едно- дву- или тристъпални с диапазон на предавателното число от 3 до 200. Зъбните предавки са цилиндрични с прави или наклонени зъби. Като правило се монтират "на лапи" или чрез фланец. Компановката на този вид мотор редуктор е сходна на планетните, вълновите и циклоидните редуктори, които също са съосни.

Цилиндричен мотор редуктор с успоредни валове. При него оста на изходящия вал не съвпада с оста на вала на двигателя. Редукторите са едно-, дву- или тристъпални. Диапазонът на предавателното число е от 5 до 200. Използват се цилиндрични зъбни предавки с прави или наклонени зъби. Може да се монтира чрез фланец, "на лапи" или с реактивна опора. Изходящият вал е плътен или кух. Основното конструктивно предимство на кухия вал е, че редукторът може да се монтира към задвижващия вал на машината без използването на еластичен съединител.

Конусен мотор редуктор. При него осите на входящия и изходящия вал се пресичат под 90°. Редукторите са едно-, дву- или тристъпални. Като второ и трето стъпало обикновено се използват цилиндрични зъбни предавки. Диапазонът на предавателното число за тристъпалните редуктори е от 25 до 80. Тези редуктори могат да се монтира чрез фланец, "на лапи" или с реактивна опора. Изходящият вал е плътен или кух.

Червячен мотор редуктор. Обикновено тези редуктори са едностъпални с червячна предавка и предавателно число от 14 до 80 или комбинирани двустъпални с цилиндрично първо стъпало с наклонени зъби с предавателни числа над 90. Сред възможните начини на монтаж са чрез фланец, "на лапи" или с реактивна опора. Изходящият вал е плътен или кух. Червячните мотор редуктори имат плавна и безшумна работа, високи предавателни числа и сравнително малки габаритни размери. Характерна особеност на тези редуктори е свойството „самозадържане”.

Спироиден мотор редуктор - изпълняват се като едно- и двустъпални. Диапазонът на предавателното число за двустъпалните редуктори е от 5 до 200. Може да се монтира чрез фланец, "на лапи" или с реактивна опора. Изходящият вал е плътен

или кух. Предимствата са както на червячните редуктори, като спироидните имат увеличена товароносимост.

Планетни мотор редуктори - те са съосни, имат малки габаритни размери, но са с по-сложна конструкция. Произвеждат се както за големи мощности с предавателно число от 23 до 1000, така и като едно- и двустъпални сервомеханизми с предавателни числа от 3 до 100.

- **Електронни контролери** - за наблюдение състоянието на турбината системата за ориентиране на ротора и лопатките. В случай на неизправност, автоматично спира работата на турбината. Може да бъде проектирана, така че по електронен път да подава сигнал на оператора на турбината.

- **Хидравлична система** - за обезопасяване на вятърната турбина.

- **Охладителна система** - охлажда електрическия генератор, чрез въздушна или водна охладителна инсталация. В допълнение охладителната система може да съдържа и система за охлаждане на маслото в трансмисията.

- **Кула** - носи гондолата и ротора.

- **Анемометър и ветропоказател** - измерва скоростта и посоката на вятъра и чрез електронен сигнал до контролера на турбината включва или спира работата й.

References

[1] Неделчева Ст., ”Нетрадиционни и възстановими енергийни източници в електроенергетиката”, София, 2006.
[2] Тончев Г., “Вятърни турбини”, Ековат технологии, 2006.
[3] Mukund R.,” Wind and solar power systems”, CRC press, 1999.

ЕЛЕКТРОННОТО СПИСАНИЕ НА ЦЕНТЪРА ПО ИНФОРМАТИКА И ТЕХНИЧЕСКИ НАУКИ В БУРГАСКИ СВОБОСЕН УНИВЕРСИТЕТ Е ВКЛЮЧЕНО В БАЗАТА НА ВИНИТИ – РАН

проф. д-р А. Тошков, Главен редактор на списание „„Компютърни науки и комуникации”

Уважаеми колеги, имам удоволствието да ви уведомя, че електронното списание „Компютърни науки и комуникации”, http://ojs.bfu.bg/ издавано от ЦИТН на БСУ, е включено в референтната база на ВИНИТИ – РАН за 2013 г.

ВИНИТИ – РАН (Всерусийски институт за научна и техническа информация на Руската академия на науките) е най- големият информационен и аналитичен център в Русия и света.

От 1952 г. насам ВИНИТИ РАН предоставя научна и техническа информация за постиженията в следните области: математика, механика, науките за живота (биология, медицина), науки за земята (география, геофизика, геология, минно дело, опазване на околната среда), физика, астрономия, химия и инженерна химия, информатика, автоматизация и компютърни науки, електроника, електроинженерство, енергетика, металургия, транспорт, икономика и др.

Списание „Компютърни науки и комуникации” е първото издание на БСУ, реферирано в най-голямата аналитично-информационна база в света. Това е едно признание за качеството на публикациите в списанието, както и за отличната работа на редакционния екип и на екипа за техническа поддръжка на Системата за електронни издания в Бургаски свободен университет.

Благодарим на авторите на статиите, включени в публикуваните за 2012 и 2013 г. броеве, на редакторите и рецензентите за тяхната коректност и компетентност, без които сега нямаше да можем да се поздравим с този успех.

СПИСАНИЕ „КОМПЮТЪРНИ НАУКИ И КОМУНИКАЦИИ” НА ЦИТН В БСУ Е ВКЛЮЧЕНО В БАЗАТА ЗА РЕФЕРИРАНИ ИЗДАНИЯ НА НАЦИД

Електронното списание „Компютърни науки и комуникации” издавано от ЦИТН на БСУ е включено в препоръчителната база за реферирани издания на Национален център за информация и документация.

Националният център за информация и документация - НАЦИД е водеща институция в Националната информационна структура на Република България.

НАЦИД комплектува, съхранява, обработва и разпространява информация и документация за моделите за управление, структурата, контрола и ресурсното осигуряване на системите за народна просвета, висше образование и наука. Центърът поддържа информационни масиви, като изгражда и развива специализирани бази данни в областта на образованието, науката, технологиите и иновациите.

Българските научни публикации трябва да заемат достойно място в световната научна периодика. За целта е необходимо българските научни списания да станат част международните системи за рефериране, индексиране и оценяване.

За да бъдат част от Европейското изследователско пространство и за да участват успешно в световния научен обмен, българските учени и изследователи трябва да направят най-добрия избор къде да публикуват своите научни резултати. Поради тази причина НАЦИД поддържа база от данни на български научни издания, реферирани в бази данни на световноизвестни информационни центрове - http://mail.nacid.bg/newdesign/bg/index.php?id=8

Включването на „електронното списание „Компютърни науки и комуникации” издавано от ЦИТН на БСУ в базата данни на НАЦИД за реферирани издания е официално признание за достойното място, което тези издания заемат в общото семейство на научните издания в страната.