

УНИВЕРЗИТЕТ У БЕОГРАДУ  
МАТЕМАТИЧКИ ФАКУЛТЕТ

## ЗБОРНИК РАДОВА

III СИМПОЗИЈУМ „МАТЕМАТИКА И ПРИМЕНЕ”

25. И 26. МАЈ 2012.



УНИВЕРЗИТЕТ У БЕОГРАДУ  
МАТЕМАТИЧКИ ФАКУЛТЕТ

ЗБОРНИК РАДОВА – III СИМПОЗИЈУМ „МАТЕМАТИКА И ПРИМЕНЕ”

25. И 26. МАЈ 2012.

*Издавач:*

Универзитет у Београду  
Математички факултет

*За издавача:*

проф. др Миодраг Матељевић, декан

*Уредници:*

доц. др Зорица Станимировић  
доц. др Мирослав Марић  
Марек Светлик

*Одговорни уредник:*

проф. др Миодраг Матељевић

*Илустација на корицама:*

ГеоГebra центар Београд, Математички факултет

*Штампа:*

Развојно-истраживачки центар графичког инжењерства  
Технолошко-металуршког факултета у Београду, Карнегијева 4.

*Тираж:*

350 примерака

ISBN 978-86-7589-097-3

СР - Каталогизација у публикацији  
Народна библиотека Србије, Београд

**СР**

51-7(082)(0.034.2)  
371.3::51(082)(0.034.2)

СИМПОЗИЈУМ Математика и примене (3 ; 2012;  
Београд)

Зборник радова [Електронски извор] / III симпозијум Математика и примене, 25. и 26. мај 2012. ; [организатор] Универзитет у Београду, Математички факултет ; [уредници Зорица Станимировић, Мирослав Марић, Марек Светлик]. - Београд : Математички факултет, 2013 (Београд : Развојно-истраживачки центар графичког инжењерства ТМФ). - 1 електронски оптички диск (CD-ROM) ; 12 cm

Системски захтеви: Нису наведени. - Радови на срп. и енгл. језику. - Текст лат. и ћир. - Тираж 350. - Напомене уз текст. - Библиографија уз сваки рад. - Апстакти; Abstracts

ISBN 978-86-7589-097-3

1. Станимировић, Зорица [уредник] 2.

Математички факултет (Београд)

а) Математика - Зборници б) Математика - Настава - Зборници

COBISS.SR-ID 198342924

## ПРЕДГОВОР

Трећи Симпозијум “МАТЕМАТИКА И ПРИМЕНЕ”, национални скуп са међународним учешћем, одржан је 25. и 26. маја 2012. на Математичком факултету Универзитета у Београду. Главни организатор Симпозијума био је Математички факултет, а подршку скупу дали су Министарство просвете, науке и технолошког развоја Републике Србије, Завод за уџбенике и наставна средства и Удружење „Милутин Миланковић“. Активности на Симпозијуму биле су део манифестације “Мај месец математике 2012” у организацији Центра за промоцију науке.

Скуп је отворио академик Никола Хајдин, председник Српске академије наука и уметности. Учеснике и госте поздравио је декан Математичког факултета, проф. др Миодраг Матељевић, дописни члан САНУ, као и мр Александра Дрецун, директорка Центра за промоцију науке. Декан Математичког факултета је овом приликом свечано отворио дигитални легат великог српског научника Милутина Миланковића. Представљена је и књига о најпознатијем српском математичару, Михаилу Петровићу –Аласу, чији је један од аутора био управо Милутин Миланковић. У холу Математичког факултета током трајања Симпозијума била је постављена изложба посвећена животу и делу Милутина Миланковића.

На III Симпозијуму “Математика и примене“ било је око 150 учесника из бројних научно-истраживачких установа из земље и иностранства, од тога око 73 са радом. Скуп је реализован кроз три секције:

- Математика и примене - данас
- Математика и информатика у образовању
- Научноистраживачки и стручни рад студената

Излагања у секцији “Математика и примене - данас” била су посвећена актуелним темама у применама математике у различитим областима, новим правцима у истраживањима и постигнутим резултатима. У оквиру секције “Математика и информатика у образовању”, предавачи су скренули пажњу на актуелне проблеме у настави математике и информатике и предложили неке идеје за решавање тих проблема. Наставници и професори математике и информатике су активно учествовали у дискусији поводом различитих тема које се тичу процеса учења, наставе, мотивације ученика, популаризације математике, итд. Трећа секција била је посвећена научноистраживачком и стручном раду студената са свих нивоа студија. Студенти неколико факултета су у оквиру ове секције представили своје научне и стручне радове, као и резултате пројеката на којима учествују.

Са задовољством можемо констатовати да су испуњени главни циљеви одржавања скупа: сагледавање постојећих и отварање нових могућности примене математике у различитим областима, унапређење наставе математике и рачунарства, промоција и популаризација математике у савременом друштву и активно учешће студената у научним и стручним активностима у области примењене математике. Активности на Симпозијуму представљају подстрек истраживачима, наставницима и студентима за стицање и имплементацију нових знања у области примењене математике и рачунарства.

Програмски одбор симпозијума

**Академик проф. др Градимир Миловановић**, Математички институт САНУ  
**проф. др Миодраг Матељевић**, декан Математичког факултета, дописни члан САНУ  
**проф. др Бошко Јовановић**, редовни професор Математичког факултета, шеф Катедре за  
нумеричку математику и оптимизацију  
**доц. др Зорица Станимировић**, продекан за науку Математичког факултета, Универзитета  
у Београду  
**доц. др Мирослав Марић**, Математички факултет, Универзитета у Београду

Организациони одбор симпозијума

**доц. др Зорица Станимировић**, продекан за науку Математичког факултета Универзитета у  
Београду,  
**мр Миљан Кнежевић**, асистент Математичког факултета Универзитета у Београду,  
**Марек Светлик**, асистент Математичког факултета Универзитета у Београду,  
**Ђорђе Стакић**, Рачунарска лабораторија Математичког факултета Универзитета у Београду,  
**Сања Косановић**, менаџер за односе са јавношћу Математичког факултета Универзитета  
у Београду,  
**Жудана Ђокић**, студент продекан Математичког факултета Универзитета у Београду,  
**Јелена Јовић**, секретар декана Математичког факултета Универзитета у Београду,

Захваљујемо свим учесницима скупа и унапред се радујемо IV Симпозијуму “Математика и примене”, који ће бити одржан 24-25. маја 2013. у организацији Математичког факултета Универзитета у Београду и Српске академије наука и ументности. Активности на IV Симпозијуму су и ове године део манифестације ”Мај месец математике 2013” Центра за промоцију науке.

У Београду, април 2013.



## САДРЖАЈ:

<b>1. DIGITAL LEGACIES AND ARCHIVES</b> Nadežda Pejović, Nenad Mitić, Saša Malkov, Žarko Mijajlović	1
<b>2. INTELLIGENT MARINE SYSTEMS</b> Zoran Đorđević, Aleksandar Perović, Aleksandar Jovanović	5
<b>3. PROBLEMI RASPOREĐIVANJA I PAKOVANJA</b> Tatjana Davidović	15
<b>4. РАЗВОЈ ЕЛЕКТРОНСКИХ УЏБЕНИКА ИЗ МАТЕМАТИКЕ</b> Дарко Дракулић	27
<b>5. ISTRAŽIVANJEM DO MINIMUMA ILI MAKSIMUMA</b> Miodrag Mateljević, Nives Jozić, Marek Svetlik	35
<b>6. УПОЗНАВАЊЕ УЧЕНИКА СА ОСОБИНАМА ФУНКЦИЈА ПОМОЋУ ГЕОГЕБРЕ И ПРИМЕРА ИЗ РЕАЛНОГ КОНТЕКСТА</b> Наталија Будински	55
<b>7. RJEŠAVANJE NEKIH ORGANIZACIONIH PROBLEMA U NASTAVI PRIMJENOM PRONALAZENJA MAKSIMALNO BALANSIRANE POVEZANE PARTICIJE U GRAFU</b> Dragan Matić, Milan Božić	61
<b>8. ОБЛАСТ „МЕРЕЊЕ” У НАСТАВИ МАТЕМАТИКЕ У ОСНОВНОЈ ШКОЛИ</b> Петар Огризовић	69
<b>9. PROBLEM SOPSTVENIH VREDNOSTI SA NELOKALNIM ROBIN-DIRIHLEOVIM GRANIČNIM USLOVIMA</b> Sandra Hodžić	75
<b>10. DVOMREŽNA METODA KONAČNIH ELEMENATA</b> Vanja Nikolić	83
<b>11. RAZVOJ NOVIH MODELA PREDVIĐANJA RETENCIONOG PONAŠANJA U TEČNOJ HROMATOGRAFIJI HIDROFILNIH INTERAKCIJA</b> Aleksandar Đenić, Tijana Rakić, Miroslav Marić, Zorica Stanimirović, Biljana Jančić Stojanović	91
<b>12. FUNKCIJE HROMATOGRAFSKOG ODGOVORA U REŠAVANJU OPTIMIZACIONIH PROBLEMA U TEČNOJ HROMATOGRAFIJI HIDROFILNIH INTERAKCIJA</b> Marina Pekić, Sanja Ribić, Tijana Rakić, Biljana Jančić Stojanović	97
<b>13. HIBRIDNI GENETSKI ALGORITAM ZA JEDNOSTEPENI LOKACIJSKI PROBLEM NEOGRANIČENIH KAPACITETA</b> Stefan Mišković	105
<b>14. „CYCLE-CANCELING ALGORITHM“ ZA REŠAVANJE PROBLEMA PROTOKA SA MINIMALNOM CENOM</b> Olivera Janković	111
<b>15. МЕТАНЕУРИСТИКА ТАБУ ПРЕТРАЖИВАЊА ЗА РЕШАВАЊЕ GENERALIZOVANOG PROBLEMA PRIDRUŽIVANJA</b> Miloš Stanković	117
<b>16. HIBRIDNI GENETSKI ALGORITAM ZA REŠAVANJE НАВ ЛОКАЦИЈСКОГ PROBLEMA NEOGRANIČENIH KAPACITETA SA VIŠESTRUKIM ALOKACIJAMA</b> Maja Đukić	127

<b>17. PRIMENA GREBNEROVIIH BAZA NA PROBLEM INVERZNE KINEMATIKE U ROBOTICI</b>	
B. Banjac, G. Dimitrijević, S. Obradović, N. Ajduković, M. Čampara, B. Malešević	135
<b>18. SKRIVENI MODELI MARKOVA KAO METODA ZA GENERACIJU SCENARIJA</b>	
Milena Kresoja	145
<b>19. ПРИМЕНЕ ПСЕУДО-СЛУЧАЈНОСТИ У ТЕОРИЈИ ВЕРОВАТНОЋА</b>	
Богдан Обадић	154
<b>20. NEKI ELEMENTI MATEMATIČKE ANALIZE SA PRIMENAMA REALIZOVANI U PROGRAMSKOM PAKETU GEOGEBRA</b>	
Branko Malešević, Ivana Jovović, Tamara Koledin, Matija Brašnjević, Vladica Vasiljević, Aleksandar Đajić-Grujić, Marija Nenezić	163

# DIGITAL LEGACIES AND ARCHIVES

Nadežda Pejović<sup>1</sup>, Nenad Mitić<sup>2</sup>, Saša Malkov<sup>3</sup>, Žarko Mijajlović<sup>4</sup>

Faculty of Mathematics, University of Belgrade

[nada@matf.bg.ac.rs](mailto:nada@matf.bg.ac.rs), [nenad@matf.bg.ac.rs](mailto:nenad@matf.bg.ac.rs), [smalkov@matf.bg.ac.rs](mailto:smalkov@matf.bg.ac.rs), [zarkom@matf.bg.ac.rs](mailto:zarkom@matf.bg.ac.rs)

**Abstract:** In this paper we introduce the digital form the legacies of most important Serbian people from the earlier period that had significant works in mathematical sciences. The project has started by developing a Digital legacy “Milutin Milanković, which contains owns digital copies of various items related to Milanković's life and work: photos, letters, diplomas, patents, and also documents that witness about his education, research and recognitions.

**Keywords:** digital legacies, digital archives, Serbian scientists, national heritage

## 1. Introduction

There are important collections of books, photos, documents and other material related to the people from the past which had important contributions to science, or had significant roles in the development of our scientific institutions. We decided to present in digital form the legacies of most important Serbian people from the earlier period that had significant works in mathematical sciences: pure mathematics, mechanics, astronomy, theoretical physics and geosciences. Our plan is to organize these digital data in two levels.

The first level will be the digital archive, where all material will be classified, but kept in the original form. In this part of the project special attention is given to metadata and digitization standards. They have the key role in the organization of the entries in the database and they are crucial for search, retrieval and further use of this material. It is supposed that all accessible and collected material related to a selected person will be held there. Obviously, this work supposes collaboration that provides advice and support of individuals interested in this kind of project, not only of mathematicians. At this moment we are considering what the software platform for digital archive would be appropriate. One possibility is the open source software DSpace which is measured as a turnkey institutional repository application. It has already set up a preliminary version based on DSpace of the digital archive of the Faculty of Mathematics. We have just started uploading into digital archive more than 2000 items (books, papers, photos, etc) related to Milutin Milanković.

The second level will be the digital legacies related to each historical individual for which data are presented in the digital archive. While the digital archive is oriented mostly to specialists, digital legacy should address the general public. Only selected items from the digital archive will be placed there, but they will be commented and appropriately illustrated. We decided to separate digital legacies from the digital archive for several reasons. First, data in digital archive are in the raw form, often not suitable for presentation, while in the legacy they are esthetically processed and adapted for Internet presentation. Further, data in the archive often overlap, for example there are often tens of photos of the same subject. Also, the archive contains copyrighted digital material, and the owner in most cases gives only the limited access to them. There are also technical reasons, such as quality, then the resolution (of the photo for example) and the size of the data that limits the browsing of the presentation.

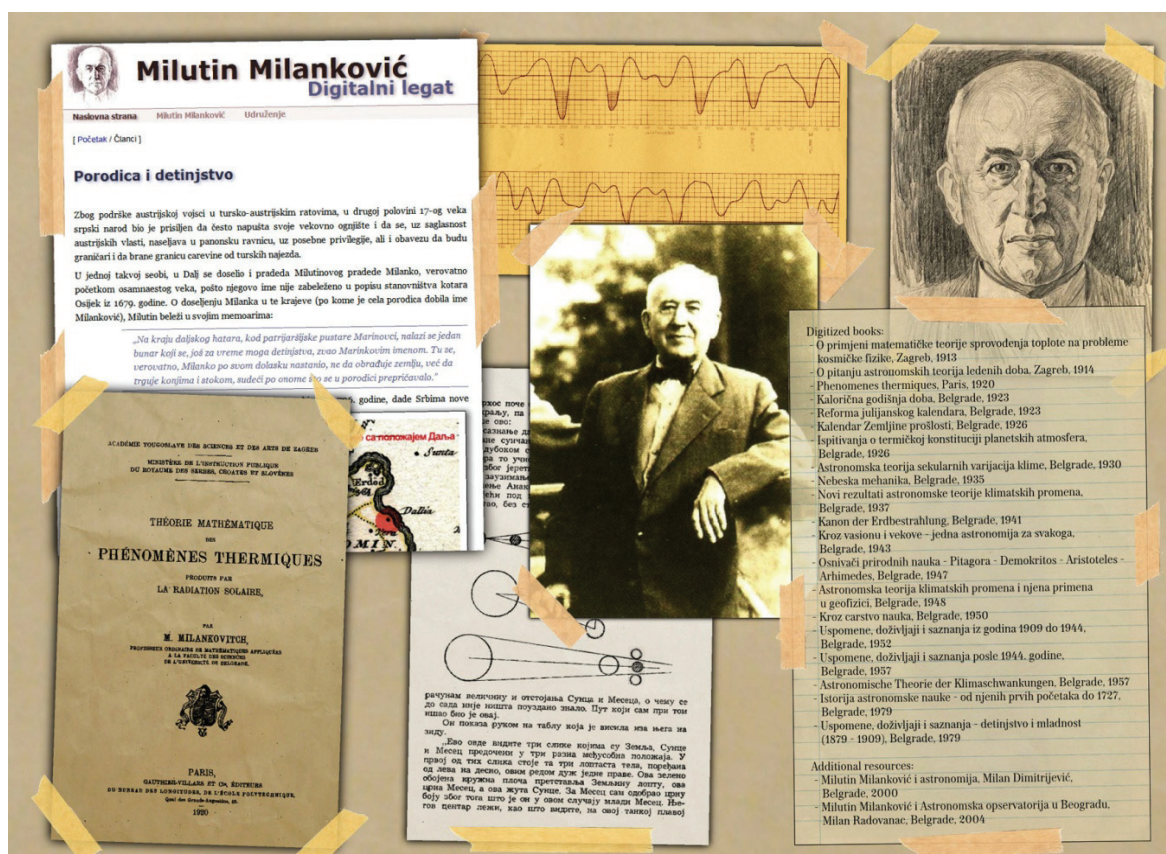
We plan to build digital legacies related to: Bogdan Gavrilović (1864-1948, mathematician, one of the founders of the University of Belgrade), Mihailo Petrović Alas (1868-1943, the founder of Belgrade mathematical school), Milutin Milanković (1879-1958, geophysicist, astronomer,

mechanist and civil engineer, best known for his theory of ice ages), Anton Bilimović (1879-1970, mechanist, the founder of the leading Serbian mathematical journal *Publications de l'Institut Mathématique*, 1932), Jovan Karamata (1902-1967, famous for his theory of regularly varying functions) and Đuro Kurepa (1907-1993, mathematician, known for his important contributions to set theory). At this moment the digital legacy of Milutin Milanković is finished. This project is coordinated by Žarko Mijajlović.

A large part of the printed materials related to these scientists, as well as for some other, see [Nikola Tesla - Clippings Library](#), especially books and scientific papers, has been already collected and partially digitized; see [Virtual Library of the Faculty of Mathematics](#). All these scientists were professors at the Faculty of Mathematics of the University of Belgrade and members of the Serbian Academy of Sciences and Arts.

## 2. Digital Legacy Milutin Milanković

We started the project first developing a [Digital legacy Milutin Milanković](#). We are working on this presentation together with the [Milutin Milanković Society](#), which owns several thousand digital copies of various items related to Milanković's life and work: photos, letters, diplomas, patents, and also documents that witness about his education, research and recognitions. These data were mainly collected by Slavko Maksimović, the president of the *Milutin Milanković Society*.



This electronic archive includes all these data in its original form, while [Digital legacy Milutin Milanković](#) contains a selection from this material with detailed explanation and comments. The accompanying text is taken mainly from [1]. The archive will contain several thousand digital items of books, paper, photos, films, patents, correspondence, diplomas and papers of other authors on Milutin Milanković. In the Digital legacy *Milutin Milanković* almost all Milanković's books are

presented, including the first edition of his famous *Canon of the Earth's Insolation* (*Kanon der Erdbestrahlung*, Serbian Academy of Sciences and Arts, 1941). There are also several books on Milanković's work and life. All these books are deposited in the Virtual Library of the Faculty of Mathematics. There is also extensive but relatively popular explanation of his theory of ice ages. Finally, there is a very good biography of Milutin Milanković written by academician Tatomir Anđelić, [2].

Finally we should mention all the people who worked on digital archive and digital legacies (in alphabetical order): Aleksandar Simonović, Mirjana Maljković, Nadežda Pejović, Nenad Mitić, Nikolina Vukša, Saša Malkov, Slaviša Milisavljević, Slavko Maksimović, Slobodan Stojanović and Žarko Mijajlović. Particular contributions of each member of the team can be found at the appropriate above mentioned Internet sites. We also had very generous supports of the Faculty of Mathematics of the University of Belgrade and the *Milutin Milanković* Society, Belgrade. This work was partially supported by the subproject 44006-4, *Digitization of scientific and cultural heritage* of the Serbian Ministry of Education and Science.

## Bibliography

- [1] Spasova, Danica; Radinović, Djuro; Milićević, Vlado; Maksimović, Slavko *Milutin Milanković putnik kroz vasionu i vekove*, Udruženje *Milutin Milanković*, Beograd, 2008.
- [2] Tatomir Anđelić, *Biografija Milutina Milankovića*, Galerija SANU, 36 (1979), 7-34.
- [3] Nadežda Pejović, *Digitisation of Textbook Небеска Механика by Milutin Milanković*, NCD Review, vol. 19, 2011, pp 63-68.
- [4] Nadežda Pejović, Žarko Mijajlović, *Early Astronomical Heritage in Virtual Library Of Faculty of Mathematics in Belgrade*, NCD Review, vol. 19, 2011, 19-25.
- [5] Saša Malkov: *Customizing a functional programming language for web development*, Computer Languages Systems & Structures, (2010), vol. 36 No. 4, pp. 345-35.
- [6] Saša Malkov, Nenad Mitić and Žarko Mijajlović: *Nikola Tesla online clipping library prototype* Review of the National Center for Digitization, 12(2008) pp. 75 – 81, <http://virlib.matf.bg.ac.rs/tesla>
- [7] Žarko Mijajlović, Zoran Ognjanović, Nada Đorđević and Tijana Zečević, *Virtual library - data base of textual data*, NCD Review, vol. 5, 2004, pp 42-48.
- [8] Ž. Mijajlović, Z. Ognjanović, A. Pejović, *Digitization of Mathematical Editions in Serbia*, Mathematics in Computer Science, Volume 3, Issue 3, 251-263, 2010.



# INTELLIGENT MARINE SYSTEMS

**Zoran Đorđević, Aleksandar Perović, Aleksandar Jovanović**

*Grupa za inteligentne sisteme - GIS, Matematički fakultet,  
Univerzitet u Beogradu, www.gisss.matf.bg.ac.rs, dzoran1@gmail.com*

**Abstract:** Modern maritime problems require the application of methods of Artificial Intelligence, AI. Ease of use of modern technological advances has significantly increased in recent years. There are constant challenges of the threat of illegal drug trafficking, arms smuggling, sea piracy cases are daily, search and rescue SAR operations are expensive and high demanding. Developed countries in an effort to dominate the world's oceans intensively invest in intelligent marine systems. The current concept of maritime system consists of sensor devices, presentational systems to the operators and command structure and big number of resources to be engaged to a potential threat. Such systems do not usually possess intelligence at a high level. In this paper, we systematize the introduction of elements of intelligence into complex marine systems.

**Keywords:** artificial intelligence, marine systems, maritime problems.

## 1. Introduction

Former naval systems could be described as unintelligent. Almost every marine system could be reduced to the sensors, presentation system to operators and command structure to monitor the situation in order to simplify the decision-making, and the actuator system by which the decisions that are made are implemented, with reduced or higher automatisation of components. In the papers [1], [8], [15] and [16] are presented principles applied in the design and implementations of domestic maritime information system with commercial trade name Horizon. Besides versions for the command level of the Navy, there was a series of implementation for ships, submarines, police, air traffic monitoring [17-19]. Generally, the sensors can be single or networked, heterogeneous or homogeneous. Examples of sensors that generate an image in the form of bitmaps or series of a bitmap-in-time, are: Radar sensors for observation of objects on the surface of the sea and in the air, sonar sensors for observation of the sea bottom or objects/living-forms in the sea, video cameras or thermal imaging in the visible part of the spectrum and so on. If data fusion process results from a number of identical sensors, then we have homogeneous sensor systems. If data originate from various sensors then it is a heterogeneous sensor system. Numerous data fusion examples of similar and dissimilar data can be seen in [2-7], [12-13].

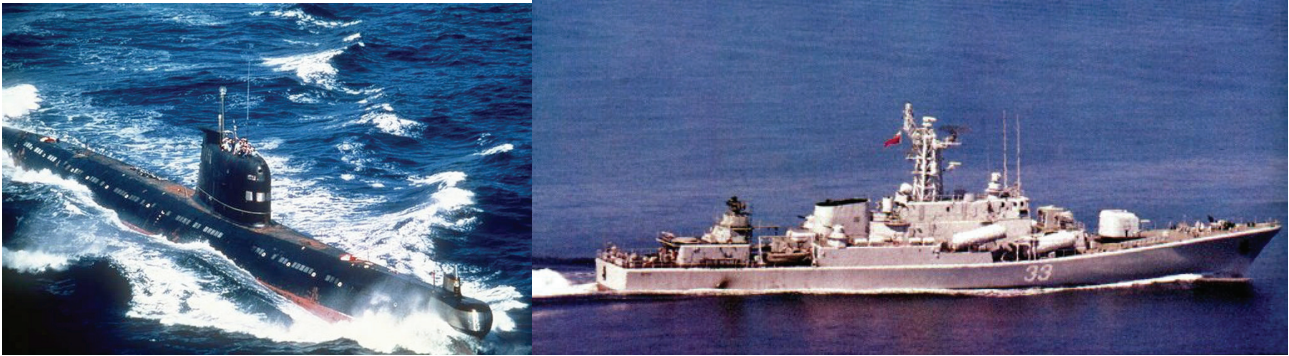
There are numerous types of displays, which present the results from the sensors. Locally displayed local sensor data we have in internal marine information system. When data from more geographically distant sensors are presented to a number of distant displays then we have distributed marine information system. In addition to the functions of monitoring data from the sensors in real-time all data can be recorded for later analysis in row or in preprocessed form. In the case of conventional non-intelligent system operators, or groups of operators, observe the results in real-time and propose specific actions. Deployed automatized actions involving a variety of methods address weapon systems (artillery, missiles, underwater weapons) or navigation control (rudder, hydroplane, propulsion systems) or aim towards requests for the involvement of aviation, helicopters, boats, patrol boats.

Actions that may be taken include: inspection and control, escort with protection, navigational route changes, the engagement of weapon systems to deter and intimidate or to annihilate objects.

This paper discusses different approaches of AI applications in marine systems, briefly presents known methods and results. Results of the Group for Intelligent Systems, GIS at Faculty of Mathematics in extended development of command and information systems with commercial trade



name ORCA and HORIZON which are previously developed in Yugoslav Navy are briefly presented with the discussion of directions for further development. Fig 1 shows the submarine of Foxtrot class and a rocket frigate on which the systems ORCA and HORIZON were integrated.



**Figure 1.** Left: a foxtrot class sub; right: the rocket frigate hosting Horizont system.

## **2. Intelligence on different levels**

Intelligence can be applied at various levels. Intelligence applications near sensors are considered as the low level applications. Usually intelligence is used in the process of forming the detection threshold and sensor data processing. Fusion of data from homogeneous or heterogeneous sensors usually is considered to be at a higher-level. Dominant problems at a higher level are detection, recognition and classification. The top level intelligence involves navigation task-management, weapon systems, command support, identification of threats. In this way intelligence application provides more precise definition of the situation elements and faster and more effective actions.

Problems in the application of intelligent methods in marine systems are numerous and complex. In the late 70's, so called command information systems were introduced. Maritime systems for vessel tracking purposes do not possess standardized method of identification. There have been attempts to roughly classify detected objects and label them. Unfortunately, due to the fluctuation of radar reflection an identification is easily lost, as a consequence, there are multiply noted objects - under the new labels of identification, and then, due to obsolescence the old one is deleted. In systems for Air Traffic Control the secondary radar system, so-called SSR was introduced with rather complex exploitation procedure. Numerous attempts to introduce something similar in marine systems after a short period of exploitation often failed.

### **2.1. Identification problem solution in marine systems - Automated Identification System - AIS**

The problem in radar use is the identification of the observed object. Despite attempts to automate the recognition of contour and shape of the observed objects at sea and in the air, there are no satisfactorily good and reliable mathematical and technological solutions, which are secure and reliable in identification tasks. The observed structures resemble mostly irregular shapes, in size closer to dots, whose shape and appearance fluctuates in time. The identification task using only the radar sensor in the case of maritime navigation on marine routes control, often with hundred(s) of vessels in the observed area, is a hard problem to solve. Then, auxiliary high magnification video equipment is deployed. Such systems are problematic because there are fluctuations in the atmosphere above the sea surface, which leads to distortion of the image causing recognition problems. Automatic Identification System - AIS is implemented to make vessel identification task more reliable and secure. An example is shown in Fig 2. During the past decade system was implemented widely. AIS system was created on the basis of similar system that is used to control the airspace. The system is technically and technologically fully standardized. As the backbone, the AIS system uses a network of digital radio transceivers operating on two frequencies 161.975MHz and 162.025MHz. Digital data



are transmitted at a rate of 9600 bps and used is Gaussian Minimum Shift Keying GMSK modulation over channel bandwidth of 12.5 kHz or 25KHz, with protocol High Level Data Link Control HDLC - packet control. Using STDMA (Self Organized Time Division Multiple Access) multiplexing technology, the participation of more subjects at same time is provided.

The information that comes from the AIS system is in the form of ASCII characters - vectors and it is relatively easy to process them, resulting in memory vessel tables with static, dynamic, and data related to the specific trip for each reported vessels. The corresponding symbol shows the received position on the screen over an electronic sea charts or bitmap obtained via radar. Usually radar images, data produced by processing radar image, GIS system and data obtained from the AIS are shown simultaneously. Among all these data usually there is no any interaction – they are simply overlapped. The man operator decides whether to assign object produced by radar to AIS symbol or with geographical formation on the coast.



**Fig 2.** The portion of the sea around the port of Amsterdam, around 120NM, with 2267 AIS identified objects. (image captured in mid March 2012; Source: [www.aishub.net](http://www.aishub.net).)

The problem of identifying objects on the sea, by using AIS technology is largely resolved. Unfortunately it is impossible to equip all boats and vessels with this system. Under current regulations only vessels over 300 GRT are required to have AIS transceiver. Navy vessels turn on and off their AIS at will. Boats with bad intentions avoid self-identification with the AIS system, they even try to reduce their radar reflection, and are made in the form of mini-submarines or semi submersible boats with only air intake tube above the sea. Small fast boats with pirates which are trying to conduct pirate attacks on merchant ships do not have AIS transceivers and on radar displays usually are represented as small and are often neglected by automated radar processing module. From small and fast boats, electromagnetic energy reflections are not persistent and have large fluctuations. When radar reflection from pirate boats become visible, it is too late for any useful action for the crew on the target ship.

## 2.2. Data Fusion

Designers of a system which displays data from GIS, radar images in the form of bitmaps, and data from AIS have to be careful to avoid possible errors by neglecting projection properties.

Nautical charts are usually drawn in one of the projections - Gauss Kruger, Mercator. The radar sensor measures the distance to the object in a spherical coordinate system. For marine radars when taking into account radiation pattern, results of radar works are presented in polar coordinate system. Images are converted to orthogonal coordinate system.

In the design of marine command information system HORIZON ([1], [8], [15] and [16]) for Yugoslav navy, a compromise was made in the choice of coordinate system's synchronization methods to reduce the error of position presentation. GIS data is stored in a database in geographic coordinates (latitude and longitude) and WGS-84, as reference ellipsoid was chosen. Data received from the AIS system, which are related to the position are also referenced to WGS-84 system. The measurement data come from radar in the form of vectors with 8 or 16 bit value representing the intensity of the reflection at a certain distance. In 2D naval radar each such vector is oriented with respect to the north. A special mathematical block (software or hardware/FPGA) implements conversion from polar to rectangular coordinate system, together with function which integrate values from previous vectors in time (reflection will be considered as a reflection only if it has cell values greater than some threshold at the distance, threshold can be fixed or variable). In papers [9-11] and [14] the use of intelligent methods in the low-level signal processing time series signal, spectrum and spectrum versus time are described.

In addition to traditional sensors like radar, sonar and video cameras, as well as their networks, AIS in marine systems, radar with SSR transponders in aircraft, numerous databases of vessels or aircrafts are used. Identification systems in the marine and air surveillance are characterized as the systems which provide accurate information (identification, position, elements of the movement). In recent years, technology of SSR radar in air traffic control increasingly use the so called S mode, which provides similar information as marine AIS system.

### 2.3. Introducing intelligence in marine systems

Many existing navies' possess marine systems which are technologically outdated and ineffective. There are major problems with maintenance, repair and overhaul (expensive and difficult). Use of such systems is complicated and requires a large number of operators. The systems are so different that often it is not possible to have operators which can work on different types of systems. All operators are specialized for a specific type of system. Navies, which belong to reach countries continually improve existing systems.

Usually when observing data received from the AIS, information about detected objects from radar, sonar or video cameras, if the observed object enters the field of special significance, e.g. possible hazard, an action proposal has to be made. Back in the mid 70's many systems for collision at sea avoidance were developed. If the observed object is approaching at the same angle then there is a risk of collision and it is suggested to perform some maneuver to avoid possible collision. Introducing fuzzy logic provides additional quality in the design and implementation of such systems. If there are several crisp defined hazardous zones then alarm comes only by incursion into the zone. If the danger zone is defined with FUZZY methods, then even approaching to FUZZY marked zone can indicate the potentially dangerous objects, thus reducing the possibility that the entire system of intelligent maritime control gets into saturation. An example for system saturation: If the naval ship is equipped with one cannon and have to perform actions on several targets simultaneously. In the design of intelligent naval systems it is necessary to make special program units, filtering services which monitor all received - observed data, comparing them in real time with defined lines and areas of special interest contained in fuzzy relational databases ([20], [21]). Intelligent surveillance systems propose and implement actions.

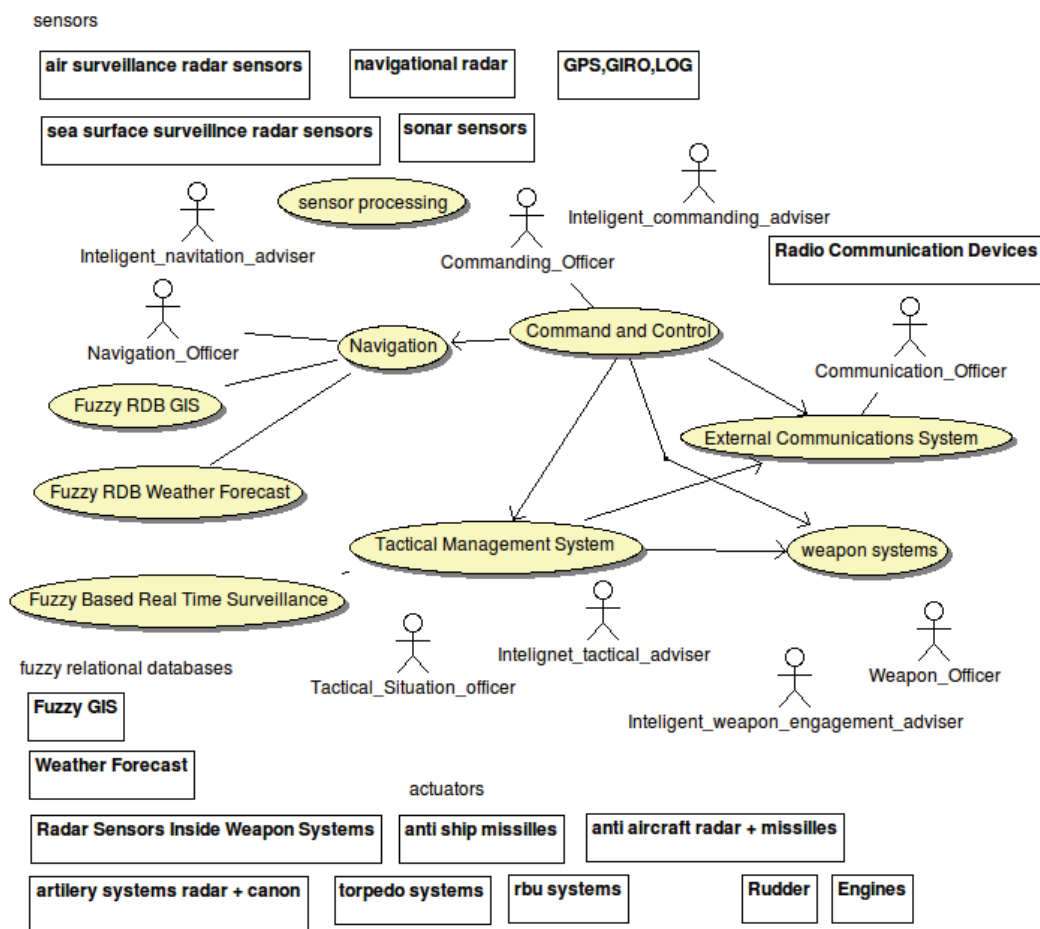
Modern technological advances offer fast computing systems and their integration into more complex functional units. PC / DSP / FPGA technologies are used as a ground technology. UML

design method makes software design and programming very effective. With mentioned technologies it is possible to set up, deploy and check various mathematical models. Fig 3 shows the UML model used in the design of HORIZON system [1-9]. Special attention is given to the definition and implementation of intelligent advisor - services that process or analyze - filter out the situation and make advices to crew members, which are responsible for situation recognition and corresponding action execution.

Modern warships are the platform with crew, which has at its disposal the sensors, actuators and databases. Group of sensors are numerous types of active radar sensors, passive radar sensors, active and passive sonar in towed or hull based version, the video camera in the visible or infrared spectrum, laser sensors, sensors for various parameters of ship motion, meteorological sensors and many others. Examples of databases that are used in naval systems are database of Geographic Information System, databases providing weather situation, the database with layout of the rooms, a database of other civilian and military ships, aircraft (friendly and enemy). In this paper, the systems for propulsion and steering gear are classified as actuators. In this sense, the own-position-control is an actuator subsystem. Similarly, the weapon systems belong to actuators. Weapon systems are composed of various types of missile weapons, torpedoes, depth charges, artillery, small and large caliber weapons and weapons with effects of directed energy (sound generators, microwave generators, lasers).

Despite the most advanced solutions in the field of sensor technology, information processing, in a possible modern real combat situation it is still possible that crew enters saturation.

Here, the UML technology modeling of marine information system is shown.



**Figure 3.** Shows UML use case diagram with proposed model of intelligent navy ship information system

The system modeled in this way will be basis for the integration of intelligent ship command and control system. Computer based intelligent advisers are included. The fact that there is a great combat system power of weapon systems installed on the warships must be respected and taken into account so that the computational intelligent systems must not be allowed to make decisions and

implementation independently without human permission, or such cases must be restricted properly. In this paper an alternative in the form of intelligent advisors is offered. Several different tasks have been identified that could be realized in the form of intelligent advisors.

## 2.4. Proposed Intelligent Tasks

Short list of eventual intelligent tasks is presented in this section. Based on our own experiences, five tasks that form parts of intelligent systems are proposed together with their results.

### *1. Proposing a favorable route for navigation.*

In case that ship is equipped with a Fuzzy Geographic Information System, as mentioned in the works [20] and [21], special intelligent advisor should suggest the best route of navigation, taking into account the current weather situation and weather forecast. Such examples are: a bypassing ship, avoidance of stormy areas with the lowest risk or the best route through the demanding navigation area selection, or iceberg avoidance.

These tasks should be solved on the basis of Fuzzy GIS database and Fuzzy Relational Databases with weather forecasts, taking into account fuzzy rules of ship movements.

### *2. Observed object type identification.*

On the basis of intelligence it is necessary to establish a Fuzzy Relational Database of known object. Such a FR database should have information about objects on the sea, in the air and under the water. These are objects like aircrafts, vessels, submarines, torpedoes ... The database should have information that can be obtained from all sensor types. As an example, let's mention radar reflected shape, the thermal spectrum, sonar spectrum and the visible silhouette. Based on data from its own sensors, radar detectors, radar, video cameras need to identify the types of detected objects.

### *3. Proposed action on an earlier identified target.*

If the observed object is recognized as hostile and could possibly pose a potential threat, it is necessary to form the proposal that one of provided arms should act on it in order to annihilate or neutralize it, with the criteria to fulfill this task with as little material costs as well.

### *4. Assessing the potential terrorist threat*

A special intelligent advisor should be designed to continuously analyze the information with a small and inexpensive sensors such as video cameras and navigation radars in terms of recognition of pirate action or terrorist intent.

### *5. The defense of your own ship*

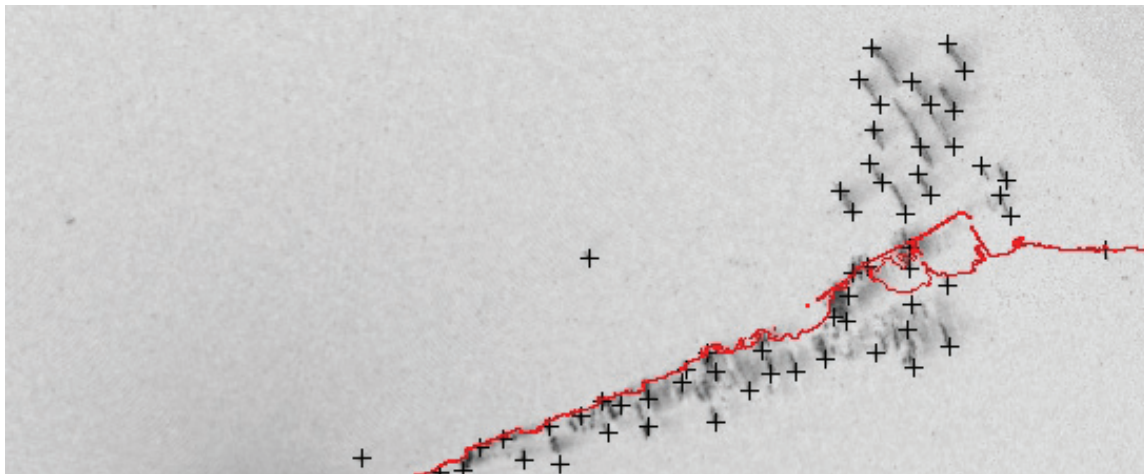
In particular, the high priority task is to propose action if some of the attacks to the ship is identified. For example, if the ship was under the artillery attack, the appropriate action should be suggested (navigation in the zigzag mode, the change of speed, whether to open fire if the missile was spotted flying towards the ship). What priority should be given according to detected missile attack to the ship, especially if the possessed anti missile system is in saturation. It is questionable whether intelligent advisor should do an action independently if the commander does not receive the proposal or does not choose any of the proposed action with some of the available actuators.

## **3. Results**

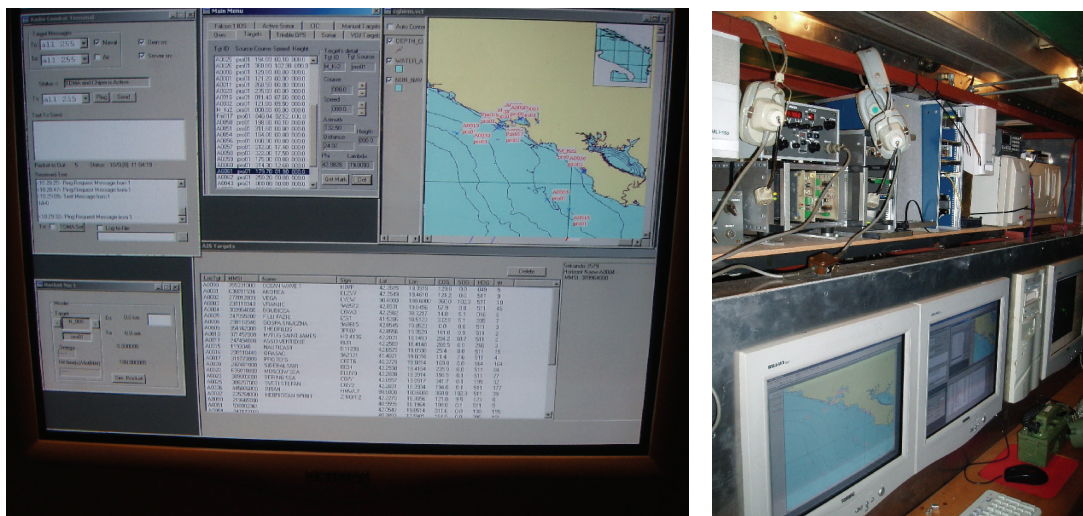
System Horizon was implemented on several different locations for different purposes. Figure 4 shows part of the situational display where data fusion is made: raw radar image, objects detected by automatic intelligent processor that recognize point like structures (modified Shi Tomasi method,



[22]) and a database of GIS. Fig 5 shows the screen with data in tabular form and graphical displays with data from radar, AIS, GIS and remote sensors. The whole system was originally designed, implemented and has been used in practice. System has many elements at lower levels of intelligence, and is the backbone for the implementation of intelligent methods at a higher level.



**Figure 4.** Situational display shows raw radar data fusion images obtained using a DSP processor with TMS3206713, our own software; the results of detection are point like objects (modified Shi-Tomasi method [22] (crosses)) and parts of the coastal line (above the red line is a sea below the land).



**Figure 5.** shown screens with data from AIS, GIS, radars

Our marine system with commercial trade name ORCA is a version of marine navigation and torpedo system for a submarine [17-19], which is implemented on a Soviet submarine type I641K (known as FOXTROT) and submarine made in Yugoslavia type 831 Sava. The system receives data from all sensors on the submarine. Sensors on the submarine are: radar and radar detector if submarines sails on the surface or periscope depth, MG200 active sonar, passive sonar MG10, passive/active sonar Eledone, passive sonar MG13, global positioning satellite receiver, AIS receiver. All sensors for measuring proper motion are connected to the system ORCA: gyrocompass, log, sounder, depth sensor. Torpedo weapon system located in the stern and bow, are connected as actuators (and fully controlled too). ORCA system has supported database in the form of GIS.

Particularly interesting problem to be solved on a submarine is task to determine the elements of movement of objects on the basis of data from passive monitoring - passive sonar, periscope and

radar detectors. These sensors generally provide only the angle (wrt the north) of the observed object, which is changing in time. There are special intelligent signal processor with a database of underwater sounds and features that give the estimated identification of the observed object, the estimated speed and the estimated distance.

Using special training, operators are trained to assess the elements of moving of passively observed objects. As a result of the operator work the distance to the object, its speed and heading (angle of movement with respect to north) are estimated. Azimuth angle of the observed object is obtained from passive sensors.

## Conclusion

The amount of information in the current maritime situation is great. Classical systems have no intelligence at a high level. Therefore, data processing using the classic method becomes inefficient - impossible. The introduction of intelligence applied at various levels: sensors, databases, GIS, the problems of data fusion with intelligent advisers can improve abilities and give better response to modern demands and better survival chances.

## References

- [1] **M. Aleksić, Z. Đorđević, S. Mitrović.** Koncepti modernizacije i integracije mornaričkih komandno informacionih i oružnih sistema, *Zbornik radova, ETRAN 2003, Herceg Novi, 2003.*
- [2] **S. S. Blackman,** Multiple Target Tracking with Radar Applications, *Artech House, 1986.*
- [3] **Z. Đorđević.** Modeling Intelligent Radar echo classificatory system, *ETRAN 2011, Jun 2011*
- [4] **Z. Đorđević.** Intelligent real time GIS based classificatory method for maritime surveillance systems, *SISY 2011, 2011 IEEE 9th International Symposium on Intelligent Systems and Informatics, September 8-10, 2011, Subotica, Serbia doi: 10.1109/SISY.2011.*
- [5] **Z. Đorđević,** Intelligent system for automatic maritime traffic control, *SISY 2011, 2011 IEEE 9th International Symposium on Intelligent Systems and Informatics, September 10-11, 2011, Subotica, Serbia doi: 10.1109/SISY.2010.5647305*
- [6] **Z. Đorđević, A. Jovanović, A. Perović,** Models of intelligent marine surveillance systems, *ETRAN 2010, Jun 2010*
- [7] **Z. Đorđević, N. Gaćeša, A. B. Dorohov.** Modeling of Intelktual radar system for maritime surveillance, *Системи озброенная и вийськова техника, 2011, No 3(27) ISSN 1997-9568, pp28-32*
- [8] **Z. Đorđević, M. Aleksić, S. Mitrović.** Koncepti modernizacije brodskih upravljačkih sistema, *20-ti Godišnjak Fakulteta za pomorstvo, Kotor, 2003.*
- [9] **Z. Đorđević, M. Aleksić, S. Mitrović, D. Radić.** Implementacija distribuiranog procesora signala radara za osmatranje objekata na moru, *Zbornik radova, ETRAN 2003, Herceg Novi, 2003.*
- [10] **M. Skolnik.** Introduction to Radar Systems, *McGrawHill Book Companies, 2008.*
- [11] **S. Spasić, A. Perović, W. Klonowski, Z. Đorđević Z., W. Duch, A. Jovanović.** Forensics of features in the spectra of biological signals, *International Journal for Bio Electromagnetism, Vol. 12, No. 2, pp. 62 - 75, 2010. [www.ijbem.net](http://www.ijbem.net)*
- [12] **A. Jovanović, Z. Đorđević, F. Marić, M. Marić, D. Perisić,** A tool for all astro sensor recordings fusion into color composite images, *Dec. 2003. Serbian Astronomy Journal.*
- [13] **A. Jovanović, Z. Đorđević, F. Marić, M. Marić, D. Perisić,** CCD microscopy – image analysis by Group for Intelligent Systems – GIS, *Archive of Oncology, 11|2, 2003.*
- [14] **A. Jovanović, A. Perović, W. Klonowski, W. Duch, Z. Đorđević, S. Spasić.** Detection of structural features in biological signals, *Journal of Signal Processing Systems, Vol 60, No. 1, pp. 115-129, 2010, Springer, DOI:10.1007/s11265-009-0407-7*

- [15] **A. Jovičić, S. Mitrović, Z. Đorđević, M. Aleksić.** Jedan pristup projektovanju komandno informacionog sistema za nadzor obale, suzbijanje kriminala i podršku spašavanju na moru, *Info M. – Časopis za informacionu tehnologiju i multimedijalne sisteme, SAVPO/JUMM, Beograd, 2003, god. 2, br. 8, str. 9-12.* [COBISS.CG-ID 512082914]
- [16] **A. Jovičić, Z. Đorđević, S. Mitrović, M. Aleksić.** Functional model of C4ISR system for coastal surveillance, crime prevention and rescue support. *Zb. Fak. pomor. Kotoru, 2005, god. 21, br. 21, str. 359-366.* [COBISS.CG-ID 512082146]
- [17] **S. Mitrović, Z. Đorđević, M. Aleksić.** Modernizacija torpednog sistema podmorice, *Zbornik radova, ETRAN 2003, Herceg Novi, 2003.*
- [18] **S. Mitrović, Z. Đorđević, M. Aleksić.** Modifikacija računarskog dela broskog protivpodmorničkog sistema, *Vojnotehnički glasnik, Beograd, 2007, god. 55, br. 3, str. 308-319.* [COBISS.CG-ID 512168930]
- [19] **S. Mitrović, Z. Đorđević, M. Aleksić.** Modernization of submarine torpedo system, *ETF Journal of Electrical Engineering, Vol. 12, No 1, 2004, str. 132-137.* [COBISS.CG-ID 512084962]
- [20] **Đ. Obradovic, Z. Konjovic, E. Pap.** Extending PostGIS by imprecise point objects, *Intelligent Systems and Informatics (SISY), 2010 8th International Symposium on , vol., no., pp.23-28, 10-11 Sept. 2010*
- [21] **A. Takači, S. Škrbic.** Priority, Weight and Threshold in Fuzzy SQL Systems. *Acta Polytechnica Hungarica, Vol 5. issue 1, 2008. 49-68,*
- [22] **J. Shi, C. Tomasi.** Good features to track Computer Vision and Pattern Recognition, *Proceedings CVPR '94., 1994 IEEE Computer Society Conference on, 1994, 593 -600*





# Problemi raspoređivanja i pakovanja

Tatjana Davidović

Matematički institut SANU, Kneza Mihaila 36/III, 11001 Beograd

e-mail: tanjad@mi.sanu.ac.rs

**Rezime.** U svakodnevnom životu često nailazimo na probleme raspoređivanja (poslovi na mašine, radnici u smene, vozila na rute, i sl.). Problemi pakovanja su veoma slični i takođe sveprisutni. Obe grupe problema spadaju u kategoriju problema kombinatorne optimizacije. Svi ovi problemi mogu se precizno matematički formulisati i rešavati poznatim matematičkim alatima. Najčešće korišćeni alati najgrublje se mogu podeliti na tačne (egzaktne) i heurističke metode. Na primeru raspoređivanja zadataka na računarske mreže biće ilustrovano kako se problemi kombinatorne optimizacije mogu matematički opisati, kao i nekoliko metoda za rešavanje ovog problema. Biće razmatrano i nekoliko varijanti problema dvodimenzionog pakovanja.

**Ključne reči:** Kombinatorna optimizacija; mešovito celobrojno linearno programiranje; egzaktne metode; heuristike.

## 1. Uvod

U ovom radu razmatraju se dva značajna problema u oblasti kombinatorne optimizacije: problem raspoređivanja i problem pakovanja. Problem raspoređivanja ilustrovan je primerima u računarstvu [1]: raspoređivanjem zadataka na procesore koji ulaze u sastav nekog zadatog višeprocorskog sistema. Za svakog korisnika višeprocorskog sistema važno je da se njegovi programi izvrše korektno i u što kraćem vremenu. Svaki takav program sastoji se od modula (zadataka) od kojih će se svaki izvršiti na nekom od procesora. Dodeljivanje zadataka procesorima i određivanje redosleda njihovog izvršavanja naziva se raspoređivanje zadataka. Problem raspoređivanja od značaja je ne samo u računarstvu, već i u drugim oblastima (industriji, robotici, medicini, saobraćaju) [2–4].

Problem raspoređivanja zadataka na višeprocorske sisteme može se, na najvišem nivou, podeliti na statički i dinamički. Kod statičkog raspoređivanja su svi podaci potrebni za raspodelu unapred poznati. Osnovna odlika dinamičkog raspoređivanja je da se tokom same raspodele otkrivaju neki novi podaci koji bitno utiču na raspodelu. Dakle, dinamičko raspoređivanje imamo u slučaju da skup zadataka nije zadat unapred, nije fiksna već se menja u toku same raspodele (petlje koje generišu onoliko zadataka kolika je vrednost brojača koja se recimo učitava na početku izvršavanja, zatim grananja koja aktiviraju samo neke zadatke, dok se oni sa "lažne" grane uopšte ne izvršavaju). To su takozvana raspoređivanja u realnom vremenu, u trenutku kada je izvršavanje polaznog programa već počelo i karakteristična su za dinamičke industrijske procese (fleksibilne robotizovane proizvodne ćelije, upravljanje letelicama i navođenim vozilima, nuklearne elektrane, meteorološke aplikacije, i sl.). Za dinamičko raspoređivanje karakteristično je i vremensko ograničenje za samu raspodelu, jer ako se zadatak ne rasporedi i ne završi u predviđenom roku, onemogućeno je izvršavanje celog programa, što može imati katastrofalne posledice u pojedinim primenama (rušenje letelice, nuklearnu katastrofu).

Specifičnost dinamičkog raspoređivanja je da u opštem slučaju rešenje problema raspoređivanja ne mora da postoji, a u nekim slučajevima, čak i kada postoji, nije sigurno da ćemo ga primenjenim postupkom dobiti, baš zbog toga što nam nisu poznate sve informacije. Drugim rečima to samo znači da na početku raspoređivanja nije izvesno da li će raspodela uspeti ili ne. Dinamičko raspoređivanje je stoga izuzetno zanimljiv i izazovan problem kojim se bave mnogi istraživači, a ovde su izdvojeni neki od radova u kojima se ta problematika razmatra [5–12]. Predmet ovoga rada ipak ostaje u domenu statičkog raspoređivanja koje je već dovoljno složeno za rešavanje, a postoji sve veća potreba u svim granama ljudske delatnosti za nalaženjem što efikasnijih rešenja.

Problem pakovanja (kao i njemu vrlo srodan problem sečenja) pojavljuje se u različitim primenama [13, 14], a veoma često se svi ovi problemi mogu svesti jedan na drugi. Glavni akcenat u ovom radu je na problemu raspoređivanja, pri čemu je za neke slučajeve uspostavljena analogija sa problemima pakovanja i/ili sečenja. Varijante problema pakovanja koje se ne mogu svoditi na raspoređivanje, takođe su pomenute.

Rad je organizovan na sledeći način: opis problema raspoređivanja dat je u sledećem odeljku. Odeljak 3 sadrži kratka razmatranja problema pakovanja. Detaljna analiza elementarnog problema raspoređivanja data je u odeljku 4, dok je odeljak 5 posvećen elementarnom problemu pakovanja i analogiji između ta dva problema.

Matematička formulacija jedne od najsloženijih varijanti problema raspoređivanja data je u odeljku 6, a zaključna razmatranja u odeljku 7.

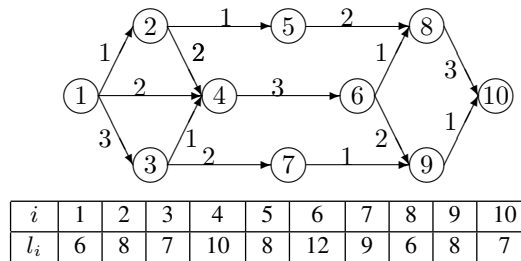
## 2. Raspoređivanje zadataka na višeprocorske sisteme

Kao što je već napomenuto, problem raspoređivanja zadataka na višeprocorske sisteme sastoji se u sledećem: dat je program koji treba paralelizovati, izdijeljen na module (zadatke); treba odrediti koji će se zadatak na kom procesoru izvršavati i kada će početi njegovo izvršavanje, a cilj je da se minimizira ukupno vreme izvršavanja celog programa. Pri tome treba voditi računa i o raznim drugim zadatim ograničenjima, kao što su zavisnost zadataka, vreme potrebno za transfer podataka (međurezultata) sa jednog procesora na drugi, rastojanje među procesorima i slično.

### 2.1. Kombinatorna formulacija problema raspoređivanja

Ovaj način opisivanja problema raspoređivanja može se naći u više radova, na primer, [15–17]. Program koji treba paralelizovati zadaje se u obliku usmerenog acikličnog grafa (Directed Acyclic Graph, DAG) definisanog kao uređena četvorka  $\mathbb{G} = (V, E, L, C)$ , gde je  $V = \{1, \dots, n\}$ , skup čvorova grafa, koji definiše skup zadataka za raspoređivanje;  $E = \{e_{ij} \mid i, j \in V\}$  označava skup lukova, kojima se opisuje zavisnost među zadacima;  $L = \{l_1, \dots, l_n\}$  predstavlja težine čvorova, kojima se zadaju dužine izvršavanja zadataka; a  $C = \{c_{ij} \mid e_{ij} \in E\}$ , skup težina na lukovima, je količina komunikacije među zavisnim zadacima, komunikaciono kašnjenje, tj. količina podataka, rezultata zadatka  $i$  (prethodnika) koje zadatak  $j$  (sledbenik) koristi u svojim izračunavanjima. Komunikaciono kašnjenje  $c_{ij} \in C$  predstavlja količinu podataka koja se fizički prenosi kroz višeprocorski sistem ukoliko su zadaci  $i$  i  $j$  dodeljeni na izvršavanje različitim procesorima. Ako se oba zadatka izvršavaju na istom procesoru, komunikaciono kašnjenje je 0. Skupom  $E$  opisana je tzv. *relacija prethodjenja* (*zavisnosti*) među zadacima. Zadatak  $i$  ne može početi svoje izvršavanje ukoliko svi njegovi prethodnici nisu završeni i svi potrebni podaci raspoloživi na procesoru  $k$ , kome je taj zadatak dodeljen na izvršavanje. Skupom  $C$  opisuje se cena zavisnosti zadataka tj. vreme koje se mora potrošiti na prenos podataka (međurezultata) između procesora. Prekid izvršavanja i redundantna izvršavanja nisu dozvoljena.

Primer grafa zadataka dat je na slici 1. Težine na lukovima označavaju količinu podataka koji se razmenjuju među odgovarajućim zadacima. Brojevi u čvorovima označavaju indekse (redne brojeve) zadataka u grafu, dok su dužine izvršavanja date u tabeli.

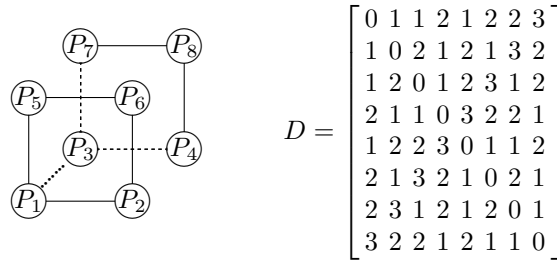


Slika 1. Primer grafa zadataka

Pretpostavlja se da se višeprocorski sistem  $\mathbb{M}$  sastoji od  $p$  identičnih procesora od kojih svaki ima privatnu, lokalnu memoriju, a komuniciraju razmenjivanjem poruka preko dvosmernih kanala od kojih su svi istog kapaciteta. Višeprocorski sistemi se mogu modelirati *neusmerenim grafovima* [17] ili *matricama rastojanja* (*distance matrix*) [18, 16]. Kod grafovskog predstavljanja čvorovi označavaju procesore, a grane komunikacione kanale između procesora. Element na mestu  $(h, k)$  matrice rastojanja  $D = [d_{hk}]_{p \times p}$  predstavlja rastojanje između procesora  $h$  i  $k$ . Dobijena iz grafovskog modela, matrica rastojanja sastoji se od rastojanja između čvorova kojima su predstavljena ta dva procesora. Lako se vidi da je matrica rastojanja simetrična matrica sa nulama po dijagonali. Na slici 2 prikazana je 3-dimenzionalna hiperkočka identičnih procesora i odgovarajuća matrica rastojanja ( $p = 8$ ).

Zavisno od arhitekture višeprocorskog sistema (strukture veza među procesorima definisane matricom rastojanja) na koju se vrši raspoređivanje, vreme komunikacije (communication delay) između zadataka određuje se na sledeći način

$$\gamma_{ij}^{hk} = c_{ij} \cdot d_{hk} \cdot ccr, \quad (1)$$



$$D = \begin{bmatrix} 0 & 1 & 1 & 2 & 1 & 2 & 2 & 3 \\ 1 & 0 & 2 & 1 & 2 & 1 & 3 & 2 \\ 1 & 2 & 0 & 1 & 2 & 3 & 1 & 2 \\ 2 & 1 & 1 & 0 & 3 & 2 & 2 & 1 \\ 1 & 2 & 2 & 3 & 0 & 1 & 1 & 2 \\ 2 & 1 & 3 & 2 & 1 & 0 & 2 & 1 \\ 2 & 3 & 1 & 2 & 1 & 2 & 0 & 1 \\ 3 & 2 & 2 & 1 & 2 & 1 & 1 & 0 \end{bmatrix}$$

Slika 2. 3-dimenziona hiperkocka i odgovarajuća matrica rastojanja

gde je  $c_{ij}$  količina podataka koju je potrebno preneti od zadatka  $i$  do zadatka  $j$  definisana skupom  $C$ ,  $d_{hk}$  predstavlja rastojanje između procesora na kojima se ovi zadaci izvršavaju zadato odgovarajućim elementom matrice  $D$ , a  $ccr$  je parametar koji određuje brzinu komunikacije u okviru datog višeprocorsorskog sistema i predstavlja količnik između vremena potrebnog da se izvrši jedna “elementarna operacija” i vremena potrebnog da se prenese jedinična količina podataka između dva susedna procesora. Ukoliko se zadaci izvršavaju na istom procesoru, tj.  $h = k$ ,  $\gamma_{ij}^{hk}$  će biti jednako nuli, jer je  $d_{kk} = 0$ , za svako  $k = 1, \dots, p$ . Još jedna važna pretpostavka je da svaki procesor poseduje zasebne ulazno/izlazne procesne jedinice. To omogućava istovremeno izvršavanje aritmetičko-logičkih i ulazno-izlaznih operacija, tj. izračunavanja i komunikacije.

Problem raspoređivanja zadataka definisanih grafom  $\mathbb{G}$  na višeprocorsorsku arhitekturu  $\mathbb{M}$ , sastoji se u tome da se za svaki zadatak odredi indeks procesora na kome će se on izvršavati i vremenski trenutak u kome će početi njegovo izvršavanje tako da se minimizira neka zadata funkcija cilja. Uobičajeno je (a to je i u ovom radu slučaj) da se kao funkcija cilja koristi vreme izvršavanja celog paralelnog programa  $T_{max} = \max_i T_i$ ,  $1 \leq i \leq n$ , pri čemu  $T_i$  označava vreme završetka zadatka  $i$ .  $T_{max}$  se naziva *dužina raspodele* (schedule length, makespan) i često se označava sa  $SL$ . Poznate su i druge funkcije cilja, na primer, vreme komunikacije (communication delay) [19], ravnomerno opterećenje procesora (load balance) [20], (težinska) suma vremena završetka svakog zadatka i sl. Ovako definisan problem raspoređivanja naziva se *problem raspoređivanja sa komunikacionim kašnjenjem* (Multiprocessor Scheduling Problem with Communication Delays, MSPCD).

## 2.2. Klasifikacija problema raspoređivanja

MSPCD nije najopštiji mogući problem statičkog raspoređivanja. U ovom radu, razmatranja su ograničena na višeprocorsorske sisteme koji sadrže identične procesore, tzv. homogene višeprocorsorske sisteme. Uopštavanje je moguće dozvoljavanjem da ne budu svi procesori u višeprocorsorskom sistemu istih karakteristika, tj. razmatranjem heterogenih višeprocorsorskih sistema (kao što je to na primer slučaj u računarskim mrežama). Heterogene sisteme takođe je moguće opisati neusmerenim grafovima [21, 22, 17], u ovom slučaju težinskim, gde težine čvorova označavaju brzinu procesora u smislu izračunavanja, tj. izvršavanja računskih operacija, dok težine lukova govore o kapacitetu veza između procesora (komunikacionih kanala), tj. brzini prenosa podataka kroz višeprocorsorski sistem.

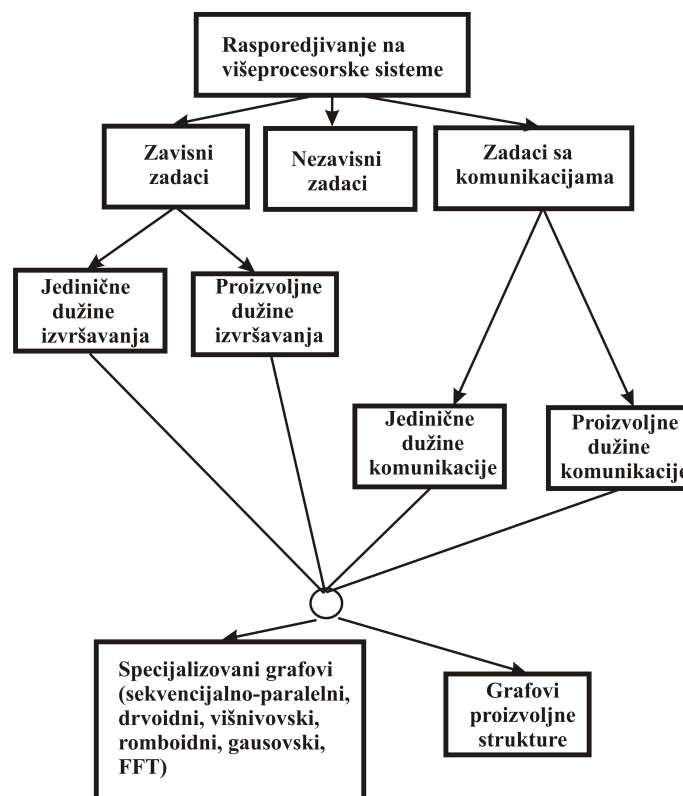
Generalizacija sa homogenih na heterogene sisteme ne predstavlja veliki problem [17]. Da bi se opisala različita dužina izvršavanja nekog zadatka na različitim procesorima dovoljno je svakom zadatku (čvoru grafa zadataka) umesto težine pridružiti niz od  $p$  elemenata takav da je  $l_{ik}$  dužina izvršavanja zadatka  $i$  na procesoru  $k$ . Pri tome, dozvoljeno je da neki element  $l_{ik}$  bude  $\infty$ , sa značenjem da procesor  $k$  ne raspolaže svim resursima (na primer, nema štampač ili CD) potrebnim za izvršavanje zadatka  $i$  (tada se radi o tzv. namenskim heterogenim sistemima). Slično tome, umesto vremena starta razmatra se vreme završetka zadatka na svakom od procesora (jer pravilo da će se zadatak tamo gde najranije počinje, najranije i završiti, više ne važi) [22].

Drugo uopštenje koje se može naći u literaturi je kada zadaci (svi ili samo pojedini) zahtevaju više od jednog procesora za svoje izvršavanje [3, 23]. Za svaki zadatak  $i$  zadaje se  $\pi(i)$ , broj koji označava koliko procesora taj zadatak zahteva za svoje izvršavanje. Prilikom raspoređivanja tog zadatka mora se voditi računa o tome da li postoji dovoljan broj slobodnih procesora kojima će se dodeliti taj zadatak na izvršavanje. Ovaj slučaj se u literaturi naziva *raspoređivanje višeprocorsorskih zadataka* (Multiprocessor Task Scheduling).

Dalja uopštavanja problema raspoređivanja moguća su u dva pravca: zavisno od toga “ŠTA” se raspoređuje i/ili “GDE” se raspoređuje. U prvom slučaju nameću se uslovi koji se odnose na zadatke, na primer vremenski okvir u kome je moguće otpočeti i/ili završiti izvršavanje svakog zadatka. U odnosu na višeprocorsorske sisteme

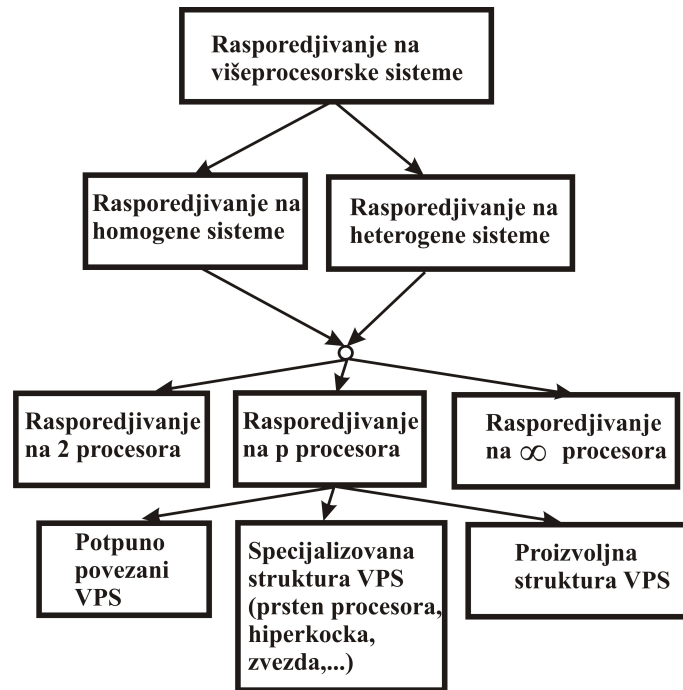
uopštavanje bi se odnosilo na to da se dozvoli neograničen broj procesora, pa da algoritam raspoređivanja utvrdi najpogodniju višeprocorsku arhitekturu.

Mnogo češće od ovih, najopštijih slučajeva, u literaturi se razmatraju neki specijalni slučajevi problema statičkog raspoređivanja zadataka. Specijalni slučajevi problema raspoređivanja ogledaju se u uvođenju dodatnih pretpostavki na strukturu grafa zadataka i/ili višeprocorskog sistema. Jedna od mogućih klasifikacija problema statičkog raspoređivanja prikazana je na slici 3 i odnosi se na klasifikaciju na osnovu karakteristika grafa zadataka koji se raspoređuje. Problem raspoređivanja može se klasifikovati i na osnovu osobina višeprocorskog sistema na koji se vrši raspoređivanje i to je prikazano na slici 4. Svaka od ovih klasifikacija već je poprilično složena, ali se stvarna slika o mogućim varijantama problema raspoređivanja dobija kombinovanjem: "svaki" od slučajeva grafa zadataka raspoređuje se na "svaku" od mogućih arhitektura višeprocorskog sistema (neke kombinacije su, naravno, besmislene, kao što je raspoređivanje nezavisnih zadataka na neke specijalizovane arhitekture). Detaljan opis mnogih varijanti može se naći u [24, 3]. Pri tome je važno napomenuti da su na slikama 3 i 4 zanemareni slučajevi raspoređivanja na jednoprocorske sisteme (Single Machine Scheduling) jer je to posebna oblast istraživanja. Međutim, i taj slučaj je od praktičnog značaja i veoma je zastupljen u literaturi kao što se može zaključiti iz [24, 3].



**Slika 3.** Klasifikacija problema raspoređivanja na osnovu karakteristika zadataka

Najjednostavniji slučaj je kada ne postoje zavisnosti među zadacima koji se raspoređuju. Ovaj slučaj biće detaljno razmatran u odeljku 4. Zatim dolazi raspoređivanje fino granuliranih zadataka, tj. zadataka jedinične dužine izvršavanja (Unit Execution Time, UET). Literatura u vezi sa ovim slučajem brojna je, na primer [25–29]. Pregledom ovih radova vidi se da se i tu mogu izdvojiti podslučajevi: bez komunikacija/sa komunikacijama, način povezivanja (zavisnost) među zadacima (drveta, višenivoovski grafovi, specijalizovani grafovi kao na primer, sekvencijalno-paralelni, romboidni, gausovski, FFT i drugi), struktura višeprocorskog sistema (2 procesora, potpuno povezani, specijalne arhitekture kao nizovi procesora, prstenovi, hiperkocke, zvezde ili proizvoljne strukture). Raspoređivanje fino granuliranih zadataka, pri čemu je u obzir uzimano i vreme komunikacije (Unit Execution Time/Unit Communication Time, UEC/UET), na višeprocorski sistem baziran na transpjuterima, razmatrano je u [30]. Iako jedan od jednostavnijih slučajeva problema paralelizacije, raspoređivanje fino granuliranih



Slika 4. Klasifikacija problema raspoređivanja na osnovu karakteristika procesora

zadataka ima veliki značaj i primenu na primer, u projektovanju i implementaciji prevodilaca za programske jezike. Čak i u savremenim jednoprocorskim računarima, postoje paralelne aritmetičko-logičke jedinice, pa je važno da se one što bolje iskoriste generisanjem efikasnih, paralelizovanih mašinskih kodova pri prevođenju programa sa nekog višeg programskog jezika.

Kao treća grupa problema može se izdvojiti raspoređivanje zadataka proizvoljne dužine izvršavanja, pri čemu se opet razlikuju podslučajevi definisani komunikacijama, vezama među zadacima i tipom višeprocorske arhitekture na koju se vrši raspoređivanje. Osim toga problem zavisi i od kriterijuma koji se koriste za utvrđivanje kvaliteta rešenja: uobičajeno je da se koristi minimizacija ukupnog vremena izvršavanja, ali se mogu pojaviti i drugi (maksimalni protok zadataka, minimizacija vremena koje se troši na komunikaciju, ravnomerno opterećenje procesora i sl.). Dakle, u definiciju problema raspoređivanja ulaze tri "parametra": struktura grafa zavisnosti među zadacima, struktura višeprocorskog sistema i kriterijum optimalnosti. Shodno tome, predložena je klasifikacija [31, 32], koja je korišćena i detaljno opisana u [3]. Svaka varijanta problema označena je trojkom  $\alpha|\beta|\gamma$  gde  $\alpha$  predstavlja karakterizaciju višeprocorskog okruženja,  $\beta$  se odnosi na opis zadataka koje treba raspoređivati, dok  $\gamma$  definiše kriterijum optimalnosti.

Neke specijalne slučajeve problema raspoređivanja moguće je rešiti optimalno u polinomnom vremenu. Na primer, u radu [33] predložen je algoritam polinomijalne složenosti ( $O(n^2)$ ) za optimalno raspoređivanje proizvoljno povezanih zadataka podjednake dužine na dva procesora. Autori rada [28] razmatrali su problem raspoređivanja drveća, degenerisanih grafova, na neograničeni broj procesora. U radu [34] pokazuje se da za višenivoovske grafove (u kojima veze postoje samo među zadacima na susednim nivoima) postoji algoritam za dobijanje optimalne raspodele u polinomnom vremenu, ukoliko je zadovoljen uslov da je maksimalna zahtevana komunikacija u celom grafu manja od minimalnog vremena izvršavanja zadatka, tj. od dužine izvršavanja najkraćeg zadatka u grafu. Ovo su samo neki od mnogobrojnih radova koji razmatraju specijalne slučajeve problema raspoređivanja za koje se može naći optimalno rešenje u polinomnom vremenu. Detaljnije o toj problematici, kao i o raspoređivanju uopšte, može se naći u [3], ili u preglednom članku [24].

Najviše radova koji se mogu naći u literaturi bavi se uvođenjem raznih heuristika za dobijanje zadovoljavajućeg suboptimalnog rešenja. Dakle, polazi se od pretpostavke da je graf zadataka unapred zadat i vrši se pridruživanje zadataka procesorima primenom nekog heurističkog rezonovanja. Na primer, prvo se raspoređuju zadaci koji imaju više sledbenika, jer će se njihovim završavanjem stvoriti uslovi za izvršavanje većeg broja zadataka što ranije. Sledeći primer bio bi forsiranje zadataka koji se duže izvršavaju jer oni zahtevaju najviše

vremena. Suprotno bi bilo forsiranje što kraćih zadataka, jer se tada na početku izvršavanja odradi najviše. Zatim, metoda najranijeg starta (Earliest Start, ES) forsira startovanje izvršavanja zadatka što pre, jer to znači i njegovo ranije završavanje i stvaranje uslova za raniji početak izvršavanja njegovih sledbenika. Heurističke metode raspoređivanja najčešće se sastoje iz dva koraka: 1) određivanje totalnog poretka među zadacima, tj. utvrđivanje redosleda kojim će se zadaci raspoređivati i 2) dodeljivanje zadataka tim redom procesorima na izvršavanje. To nije opšte pravilo, na primer, PPS metoda [35] u prvom koraku određuje koji će se zadatak izvršavati na kom procesoru, a u drugom kada će početi njegovo izvršavanje.

Do sada je razvijeno mnogo heurističkih algoritama, međutim, svaki od tih algoritama efikasan je samo za određenu klasu problema. Na efikasnost algoritma utiče, pre svega, struktura grafa zadataka, zatim struktura višeprocorskog sistema. Pokušaj klasifikacije konstruktivnih heuristika urađen je u [36, 37], a ovde su samo, pregleda radi, navedene neke od najpoznatijih heurističkih metoda predloženih u literaturi.

Sledeći korak u rešavanju problema raspoređivanja je razvoj metaheurističkih metoda. Metaheuristike se sastoje od uopštenih skupova pravila koja se mogu primeniti za rešavanje raznovrsnih problema optimizacije. Primena metaheuristika (genetski algoritmi, neuralne mreže, tabu pretraživanje, metoda promenljivih okolina, simulirano kaljenje, mravlje kolonije, optimizacija kolonijom pčela, itd.), prilikom rešavanja optimizacionih problema treba da obezbedi dobijanje kvalitetnijeg suboptimalnog rešenja od onog koje daju klasične, konstruktivne heuristike za razumno (dostižno) vreme izvršavanja. Primenom metaheurističkih metoda najčešće se omogućava značajna popravka nekog početnog rešenja dobijenog klasičnim heurističkim metodama, uz razuman utrošak vremena za sopstveno izvršavanje.

### 3. Problemi pakovanja

Problemi pakovanja sastoje se u tome da se grupa manjih objekata smesti u jedan ili više većih objekata iste ili različite vrste. Jedan od najjednostavnijih primera je pakovanje manjih pravougaonika u veći, tzv. dvodimenziono pakovanje. Trodimenziono pakovanje je veoma slično, kvadri se pakuju u veliku kutiju. Složeniji primer bio bi pakovanje nepravilnih objekata u pravougaonike ili kvadre što se uz dozvoljavanje promene orijentacije objekata koristi u implementaciji poznate igrice TETRIS. Dalje uopštavanje odnosi se na pakovanje raznih drugih objekata (lopte, poliedri, ...) u kutije koje takođe mogu biti raznih oblika.

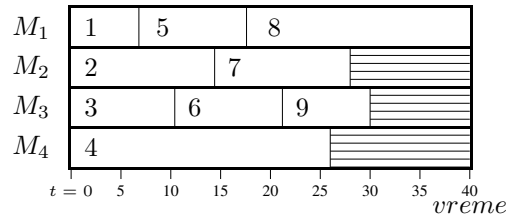
Jedan od najpoznatijih problema pakovanja je problem ranca (Knapsack problem). On se sastoji u tome da se u ranac zadate zapremine smesti što veći broj objekata za koje su takođe poznate zapremine. Uopštenje ovog problema dobija se kada se objektima dodaju i težine (vrednosti), pa je potrebno maksimizirati ukupnu vrednost svih objekata u rancu.

Problemu pakovanja veoma je sličan problem sečenja. Na primer ako od velike pravougaone table lima treba da isečemo što više manjih pravougaonika zadatih dimenzija, dobijamo klasičan problem 2D pakovanja malih pravougaonika u veće. Ovde se već nazire analogija između problema raspoređivanja i 2D pakovanja/sečenja: zadaci su pravougaonici dimenzija  $1 \times l_i$  koji se pakuju u veliki pravougaonik dimenzija  $p \times \sum_{i=1}^n l_i$ . Ako razmatramo problem raspoređivanja zavisnih zadataka i uzimamo u obzir komunikaciona kašnjenja možemo ga poistovetiti sa pakovanjem u kome postoje zabranjene zone (delovi u koje se ne može smestiti ništa), ili sa sečenjem tabli lima kod kojih nisu svi delovi istog kvaliteta.

### 4. Elementarni problem raspoređivanja

Najjednostavniji među problemima raspoređivanja je svakako raspoređivanje nezavisnih zadataka na identične procesore, tzv. job-shop raspoređivanje, kod kojega se formiraju redovi čekanja na procesore za koje tada nije važno kako, pa čak ni da li su povezani. U tom slučaju redosled izvršavanja zadataka i povezanost procesora nisu bitni, a ne postoji ni kašnjenje koje bi prouzrokovao transfer podataka. Potrebno je samo da se odredi koji zadatak se pridružuje kom procesoru, a zatim izračuna ukupno vreme potrebno za izvršavanje svih zadataka.

Detaljniji opis problema: Dat je skup koji se sastoji od  $n$  nezavisnih poslova (zadataka)  $T = \{1, 2, \dots, n\}$ . Za svaki zadatak  $i$  poznata je njegova dužina izvršavanja  $l_i$ ,  $i = 1, 2, \dots, n$ . Dat je skup koji se sastoji od  $p$  identičnih mašina (procesora)  $\mathbb{M} = \{1, 2, \dots, p\}$ . Potrebno je pridružiti poslove mašinama tako da se svaki posao dodeli tačno jednoj mašini, da svaka mašina u datom trenutku obrađuje tačno jedan posao i da se tom prilikom minimizira ukupno vreme izvršavanja svih poslova (dužina raspodele, makespan). Prekidanje kao ni dupliranje izvršavanja poslova nije dozvoljeno. Primer rezultata raspoređivanja 9 zadataka na 4 procesora koje odgovara dužini raspodele od 40 vremenskih jedinica prikazan je na slici 5.



Slika 5. Primer problema raspoređivanja nezavisnih zadataka na identične procesore

#### 4.1. Matematička formulacija problema

Problem raspoređivanja nezavisnih zadataka na identične procesore može se formulirati kao zadatak celobrojnog linearnog programiranja (Integer Linear Programming, ILP) na sledeći način [38].

Definišimo binarne promenljive

$$x_{ij} = \begin{cases} 1, & \text{ako je zadatak } i \text{ dodeljen procesoru } j, \\ 0, & \text{inače.} \end{cases}$$

Problem se sada svodi na minimizaciju funkcije cilja pod zadatim ograničenjima, tj.

$$\min y \tag{2}$$

$$s.t. \sum_{j=1}^p x_{ij} = 1, \quad 1 \leq i \leq n, \tag{3}$$

$$y - \sum_{i=1}^n l_i x_{ij} \geq 0, \quad 1 \leq j \leq p, \tag{4}$$

$$x_{ij} \in \{0, 1\}, \quad 1 \leq i \leq n, \quad 1 \leq j \leq p \tag{5}$$

Funkcija cilja: ukupno vreme izvršavanja svih zadataka - makespan  $y$ , može se shvatiti i kao  $\max y_j$  gde je  $y_j = \sum_{i=1}^n l_i x_{ij}$  za svako  $j$ . To upravo govori ograničenje (4). Prvo ograničenje treba da obezbedi da se svaki zadatak  $i$  dodeljuje tačno jednom procesoru  $j$ . Poslednje ograničenje iskazuje binarnu prirodu promenljivih  $x_{ij}$ . Promenljiva  $y$  može da bude proizvoljan broj (realan, ceo, pozitivan).

Na osnovu prikazane formulacije možemo zaključiti da se ovaj problem opisuje sa  $n \times p$  binarnih promenljivih i  $n + p$  ograničenja. Iako deluje prilično jednostavno, već za  $p = 2$ , problem je NP-kompletan [39]. Za slučaj da svi zadaci imaju istu dužinu izvršavanja, tj.  $l_i = L, \forall i$ , problem je polinomijalne složenosti (najviše  $\lceil n/m \rceil$  zadataka ide na svaki procesor, a dužina raspodele je  $\lceil n/m \rceil \cdot L$ ).

#### 4.2. Karakteristike problema i pristup rešavanju

Prilikom razmatranja problema raspoređivanja nezavisnih zadataka razlikuje se nekoliko podslučajeva: raspoređivanje na dva, više, pa čak i neograničeni broj procesora. Ova varijanta problema lako se prevodi na problem pakovanja, odsecanja ili grupisanja (Packing, Cutting, Clustering). Među radovima koji se bave raspoređivanjem nezavisnih zadataka mogu se izdvojiti [40–42, 38, 43–46]. Već se na osnovu ovog spiska radova vidi koliko raznih varijanti ovog specijalnog slučaja problema (i na koliko različitih načina) se u literaturi razmatra.

Na osnovu dostupne literature, pristup rešavanju ovog problema može se klasifikovati na sledeći način.

- Egzaktne metode (MIP-bazirane, kombinatorne) [41, 42, 38];
- Heurističke:
  - konstruktivne, LPT+ES najpoznatija [40], ima koeficijent aproksimacije  $4/3 - 1/(3m)$ ;
  - iterativne, npr. lokalno pretraživanje (LS) [44];
- Metaheurističke:
  - Tabu pretraživanje [46];
  - Metoda promenljivih okolina [45];
  - Optimizacija kolonijom pčela [43].

**Tabela 1.** Poređenje algoritama za raspoređivanje nezavisnih zadataka na identične procesore

example	OPT	Opt time	MLS	min time	VNS	min time	BCO	min time
Iogra100_4	800	0.140	801	0.491	800	0.001	800	0.027
Iogra100_6	800	8.402	801	4.211	800	0.025	801	0.044
Iogra100_8	800	98.859	805	0.284	800	0.146	803	0.065
Iogra100_9	800	862.788	807	0.799	800	1.572	807	0.224
Iogra100_12	800	MEM_ERROR	817	3.992	800	3.873	811	0.364
Iogra100_16	800	—	816	0.397	800	5.010	809	0.418

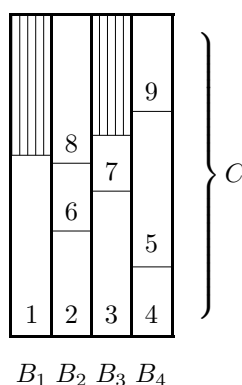
Rezultati primene metaheurističkih metoda na teške test primere raspoređivanja 100 zadataka na različit broj procesora prikazani su u tabeli 1. Opt time predstavlja vreme potrebno CPLEX softveru da pronađe poznato optimalno rešenje.

## 5. Elementarni problem pakovanja

Kao jedan od najjednostavnijih izdvajamo specijalizovani problem 2D pakovanja, pakovanje manjih pravougaonika čija je jedna od dimenzija obavezno jedan u veće pravougaonike kojima je manja dimezija takođe jednaka jedinici. Ovaj problem se u literaturi naziva Bin packing problem (BPP).

BPP se sastoji u sledećem: Dat je skup koji se sastoji od  $n$  objekata  $I = \{1, 2, \dots, n\}$ . Za svaki objekat  $i$  poznata je njegova veličina  $l_i$ ,  $i = 1, 2, \dots, n$ . Data je procenjena gornja granica na broj potrebnih paketa (bin-ova)  $K$  i veličina (kapacitet) svakog paketa  $C$ . Potrebno je sve objekte smestiti u što manji broj paketa tako da se ne premaši kapacitet ni jednog paketa.

Primer BPP-a analogan primeru raspoređivanja sa slike 5 dat je na slici 6.



**Slika 6.** Primer pakovanja u 4 paketa (bin-a)

### 5.1. Matematička formulacija problema

Definišimo najpre binarne promenljive  $x_{ik}$  i  $y_k$  na sledeći način:

$$x_{ik} = \begin{cases} 1, & \text{ako je objekat } i \text{ smešten u paket } k, \\ 0, & \text{inače.} \end{cases}$$

$$y_k = \begin{cases} 1, & \text{ako je paket } k \text{ iskorišćen,} \\ 0, & \text{inače.} \end{cases}$$

ILP za BPP ima sledeći oblik [47]:



$$\min \sum_{k=1}^K y_k \quad (6)$$

$$s.t. \sum_{k=1}^k x_{ik} = 1, \quad 1 \leq i \leq n, \quad (7)$$

$$\sum_{i=1}^n l_i x_{ik} \leq C y_k, \quad 1 \leq k \leq K, \quad (8)$$

$$x_{ik}, y_k \in \{0, 1\}, \quad 1 \leq i \leq n, \quad 1 \leq k \leq K \quad (9)$$

Dakle, cilj je minimizirati broj iskorišćenih paketa, tj. broj promenljivih  $y_k$  koje imaju vrednost 1. Ograničenja (7) govori da svaki objekat mora da bude u tačno jednom paketu. Uslov da se ne sme premašiti kapacitet svakog od paketa iskazan je ograničenjima (8). Ukupan broj promenljivih u ovom slučaju je  $(n + 1) \times K$ , dok je broj ograničenja  $n + K$ .

## 5.2. Analogija sa problemom raspoređivanja

Kao što je već napomenuto, problem raspoređivanja nezavisnih zadataka na identične procesore i BPP su vrlo slični. U oba slučaja imamo velike pravougaonike u koje treba na najpogodniji način smestiti zadate male pravougaonike. Međutim, pri raspoređivanju se minimizira dužina, dok je kod pakovanja dužina svakog paketa data preko njegovog kapaciteta i ima istu vrednost za sve pakete. Naprotiv, kod raspoređivanja je dat broj procesora, a prilikom pakovanja treba minimizirati broj paketa. Najbolja heuristička strategija kod raspoređivanja je ES, dok je kod pakovanja BF (Best Fit). Naime, pri raspoređivanju najbolje je novi zadatak staviti na najmanje opterećen procesor jer je tada najveća verovatnoća da će ukupna dužina izvršavanja ostati što je moguće manja. Obrnuto, kod pakovanja je bolje maksimalno popuniti već upotrebljene pakete pre nego što se posegne za novim. Zato se dati objekat stavlja u onaj paket u kome nakon toga ostaje najmanje slobodnog mesta. Time se u preostalim upotrebljenim paketima čuva više mesta za objekte koji tek treba da budu upakovani.

## 6. Matematička formulacija za MSPCD

Za problem raspoređivanja opisan u odeljku 2, ovde je data mešovito celobrojno linearna (Mixed-Integer Linear Programming, MILP) formulacija, bazirana na 2D pakovanju pravougaonika sa zabranjenim zonama. Izvorna formulacija je sa kvadratnim ograničenjima (MIQP), koja su naknadno linearizovana uvođenjem novih promenljivih. Formulacija je prvi put predložena u radu [48].

Osnovna ideja bila je da se svaki zadatak zamisli kao pravougaonik dužine  $l_i$  i visine 1. Sve takve pravougaonike treba spakovati u veći pravougaonik čija dužina je jednaka dužini raspodele  $W$  koja se minimizira, a visina je jednaka broju raspoloživih procesora  $p$ . Osnovna prednost ovakve formulacije nad klasičnom formulacijom predloženom u [49] je manji broj binarnih promenljivih: dok klasična formulacija zahteva  $n^2 p$  binarnih promenljivih za opisivanje da li je zadatak  $i$  raspoređen kao  $s$ -ti na procesoru  $k$ , formulacija bazirana na pakovanju koristi samo  $n^2 + np$  binarnih promenljivih.

Za svaki zadatak  $i$ , označimo sa  $t_i \in \mathbb{R}$  početak njegovog izvršavanja (tj. apscisu donjeg levog temena pravougaonika kojim se prikazuje zadatak  $i$ ), a sa  $p_i \in \mathbb{N}$  indeks procesora na kome će se izvršavati zadatak  $i$  (tj. ordinatu donjeg levog temena pravougaonika kojim se prikazuje zadatak  $i$ ). Neka  $x_{ik}$  uzima vrednost 1 ako je zadatak  $i$  dodeljen procesoru  $k$ , a nula inače. Definišimo još dva skupa binarnih promenljivih koje treba da spreče preklapanje pravougaonika (zadataka) po obe koordinate.

$$\forall i, j \quad \sigma_{ij} = \begin{cases} 1 & \text{zadatak } i \text{ se završava pre nego (ili istovremeno kad) zadatak } j \text{ počinje} \\ 0 & \text{inače} \end{cases}$$

$$\forall i, j \quad \epsilon_{ij} = \begin{cases} 1 & \text{indeks procesora kome je dodeljen zadatak } i \text{ je strogo manji od onoga na kome je zadatak } j \\ 0 & \text{inače.} \end{cases}$$

Promenljive  $\sigma$  opisuju redosled izvršavanja zadataka, dok promenljive  $\epsilon$  i  $x$  kontrolišu dodeljivanje zadataka procesorima.

Matematička formulacija za MSPCD može se opisati na sledeći način

$$\min_{t,p,\sigma,\epsilon} W \quad (10)$$

$$\forall i \in V \quad t_i + l_i \leq W \quad (11)$$

$$\forall i \neq j \in V \quad t_j - t_i - l_i - (\sigma_{ij} - 1)W_{\max} \geq 0 \quad (12)$$

$$\forall i \neq j \in V \quad p_j - p_i - 1 - (\epsilon_{ij} - 1)|\mathbb{M}| \geq 0 \quad (13)$$

$$\forall i \neq j \in V \quad \sigma_{ij} + \sigma_{ji} + \epsilon_{ij} + \epsilon_{ji} \geq 1 \quad (14)$$

$$\forall i \neq j \in V \quad \sigma_{ij} + \sigma_{ji} \leq 1 \quad (15)$$

$$\forall i \neq j \in V \quad \epsilon_{ij} + \epsilon_{ji} \leq 1 \quad (16)$$

$$\forall j \in V : i \in \delta^-(j) \quad \sigma_{ij} = 1 \quad (17)$$

$$\forall j \in V : i \in \delta^-(j) \quad t_i + l_i + \sum_{h,k \in \mathbb{M}} \gamma_{ij}^{hk} x_{ih} x_{jk} \leq t_j \quad (18)$$

$$\forall i \in V \quad \sum_{k \in \mathbb{M}} k x_{ik} = p_i \quad (19)$$

$$\forall i \in V \quad \sum_{k \in \mathbb{M}} x_{ik} = 1 \quad (20)$$

$$W \geq 0 \quad (21)$$

$$\forall i \in V \quad t_i \geq 0 \quad (22)$$

$$\forall i \in V \quad p_i \in \{1, \dots, |\mathbb{M}|\} \quad (23)$$

$$\forall i \in V, k \in \mathbb{M} \quad x_{ik} \in \{0, 1\} \quad (24)$$

$$\forall i, j \in V \quad \sigma_{ij}, \epsilon_{ij} \in \{0, 1\}, \quad (25)$$

gde  $W_{\max}$  predstavlja gornju granicu za dužinu raspodele  $W$ , tj.

$$W_{\max} = \sum_{i \in V} l_i + \sum_{i,j \in V} c_{ij} \max\{d_{hk} \mid h, k \in \mathbb{M}\}.$$

Funkcija cilja (10) i ograničenja (11) forsiraju minimizaciju dužine raspodele, vremenski okviri izvršavanja zadataka u odnosu na promenljive  $\sigma$  definisani su ograničenjima (12), a ograničenja (13) vode računa o redosledu izvršavanja zadataka u odnosu na indekse procesora definisane promenljivima  $\epsilon$ . Ograničenja (14), (15) i (16) obezbeđuju da svaki zadatak zauzima tačno jednu poziciju i po vremenu i po procesoru. Poštovanje zavisnosti zadataka obezbeđeno je ograničenjima (17), a ograničenja (18) modeliraju komunikaciono kašnjenje. Veza između promenljivih  $x_{ik}$  i  $p_i$  iskazana je ograničenjima (19), a ograničenja (20) se odnose na dodelu svakog zadatka tačno jednom procesoru.

### 6.1. Linearizacija kvadratnih izraza

Kao što se vidi iz ograničenja (18), formulacija za MSPCD nije linearna. Uobičajeni način linearizacije ovakvih izraza je zamena proizvoda  $x_{ih}x_{jk}$  novim promenljivima  $z_{ij}^{hk} \in [0, 1]$ . To zahteva i uvođenje novih ograničenja [50, 51]:

$$\forall j \in V, i \in \delta^-(j), h, k \in \mathbb{M} \quad (x_{ih} \geq z_{ij}^{hk} \wedge x_{jk} \geq z_{ij}^{hk} \wedge x_{ih} + x_{jk} - 1 \leq z_{ij}^{hk}), \quad (26)$$

što dalje implicira:

$$\forall j \in V, i \in \delta^-(j), h, k \in \mathbb{M} \quad (z_{ij}^{hk} = x_{ih}x_{jk}), \quad (27)$$

i obezbeđuje reformulaciju polaznog MIQP u ekvivalentni MILP.

Na osnovu [52], uobičajena linearizacija može se zameniti ekvivalentnom kompaktnom linearizacijom na sledeći način:

$$\forall i \neq j \in V, k \in \mathbb{M} \left( \sum_{h \in \mathbb{M}} z_{ij}^{hk} = x_{jk} \right). \quad (28)$$

$$\forall i \neq j \in V, h, k \in \mathbb{M} \left( z_{ij}^{hk} = z_{ji}^{kh} \right). \quad (29)$$

## 7. Zaključak

Pregled i značaj dva veoma važna problema kombinatorne optimizacije, problema raspoređivanja i problema pakovanja, dat je u ovom radu. Ukazano je na analogiju koja postoji između pojedinih varijanti ova dva problema i navedeno kako je problem pakovanja manjih pravougaonika u jedan veći pravougaonik zadatih dimenzija poslužio kao inspiracija za razvoj efikasne matematičke formulacije problema raspoređivanja zavisnih zadataka na proizvoljno povezane identične procesore uz prisustvo komunikacionih kašnjenja.

## Literatura

- [1] T. Davidović. *Raspoređivanje zadataka na višeprocorske sisteme primenom metaheuristika*. PhD thesis, Matematički fakultet Univerziteta u Beogradu, Feb. 2006.
- [2] J. Blazewicz, K. H. Ecker, E. Pesch, G. Schmidt, and J. Weglarz. *Handbook on Scheduling from theory to Application*. Springer, 2007.
- [3] P. Brucker. *Scheduling Algorithms*. (fifth edition) Springer Verlag, 2007.
- [4] M. Pinedo. *Scheduling theory and its applications*. John Wiley & Sons, 1995.
- [5] I. Ahmad, A. Ghafoor, and G. C. Fox. Hierarchical scheduling of dynamic parallel computations on hypercube multi-computers. *J. Parallel and Distributed Computing*, 20:317–329, 1994.
- [6] W. W. Chu and L. M-T. Lan. Task allocation and precedence relations for distributed real-time systems. *IEEE Trans. Computers*, C-36(6):667–679, June 1987.
- [7] M. L. Dertouzos and A. K. Mok. Multiprocessor on-line scheduling of hard-real-time tasks. *IEEE Trans. Software Eng.*, 15(12):1497–1506, December 1989.
- [8] S. Mitrović-Minić. *The dynamic pickup and delivery problem with time windows*. PhD thesis, School of Computing Science, Simon Fraser University, Vancouver, Canada, 2002.
- [9] S. Mitrović-Minić, G. Laporte, and R. Krishnamurti. The double-horizon heuristic for the dynamic pickup and delivery problem. In *Optimization Days*, page 77, Montreal, 2002.
- [10] D. Peng and K. G. Shin. Modeling of concurrent task execution in a distributed system for real-time control. *IEEE Trans. Computers*, C-36(4):500–516, Apr. 1987.
- [11] K. Ramamritham, J. A. Stanković, and P-F. Shiah. Efficient scheduling algorithms for real-time multiprocessor systems. *IEEE Trans. Parallel and Distributed Systems*, 1(2):184–194, Apr. 1990.
- [12] K. Ramamritham, J. A. Stanković, and W. Zhao. Distributed scheduling of tasks with deadlines and resource requirements. *IEEE Trans. Computers*, 38(8):1110–1122, Aug. 1989.
- [13] D. Cvetković, M. Čangalović, Đ. Dugošija, V. Kovačević-Vujčić, S. Simić, and J. Vuleta. *Kombinatorna optimizacija (Matematička teorija i algoritmi)*. DOPIS, Beograd, 1996.
- [14] J. Levine and F. Ducatelle. Ant colony optimization and local search for bin packing and cutting stock problems. *Journal of the Operational Research Society*, (55):705–716, 2004.
- [15] G. Djordjević and M. Tošić. A compile-time scheduling heuristic for multiprocessor architectures. *The Computer Journal*, 39(8):663–674, 1996.
- [16] Y.-K. Kwok and I. Ahmad. Efficient scheduling of arbitrary task graphs to multiprocessors using a parallel genetic algorithm. *J. Parallel and Distributed Computing*, 47:58–77, 1997.
- [17] G. C. Sih and E. A. Lee. A compile-time scheduling heuristic for interconnection-constrained heterogeneous processor architectures. *IEEE Trans. Parallel and Distributed Systems*, 4(2):175–187, February 1993.
- [18] T. Davidović. Exhaustive list-scheduling heuristic for dense task graphs. *YUJOR*, 10(1):123–136, 2000.
- [19] A. K. Sarje and G. Sagar. Heuristic model for task allocation in distributed computer systems. *IEE Proceedings-E*, 138(5):313–318, Sept. 1991.
- [20] I. Ahmad and A. Ghafoor. Semi-distributed load balancing for massively parallel multicomputer systems. *IEEE Trans. Software Eng.*, 17:138–153, 1990.
- [21] D. A. Menascé, S.C.S. Porto, and S. K. Tripathi. Static heuristic processor assignment in heterogeneous multiprocessors. *Int. J. High Speed Computing*, 6(1):115–137, 1994.
- [22] S.C. Porto and C.C. Ribeiro. A tabu search approach to task scheduling on heterogeneous processors under precedence constraints. *Int. J. High-Speed Computing*, 7:45–71, 1995.

- [23] K. Li. Analysis of the list scheduling algorithm for precedence constrained parallel tasks. *Journal of Combinatorial Optimization*, 3:73–88, 1999.
- [24] J. Blazewicz, M. Drozdowski, and K. Ecker. Management of resources in parallel systems. In J. Blazewicz, K. Ecker, B. Plateau, and D. Trystram, editors, *Handbook on Parallel and Distributed Processing*, pages 263–341. Springer, 2000.
- [25] P. Brucker, S. Knust, D. Roper, and Y. Zinder. Scheduling uet task systems with concurrency on two parallel identical processors. *Mathematical Methods of Operations Research*, 52:369–387, 2001.
- [26] N. Kirčanski, T. Davidović, and M. Vukobratović. A contribution to parallelization of symbolic robot models. *Robotica*, 13:411–421, 1995.
- [27] K. Politopoulos, G. F. Georgakopoulos, and P. Tsanakas. Precedence constrained scheduling: A case in *p*. *The Computer Journal*, 44(3):163–173, 2001.
- [28] T. A. Varvarigou, V. P. Roychowdhury, T. Kailath, and E. Lawler. Scheduling in and out forests in the presence of communication delays. *IEEE Trans. Parallel and Distributed Systems*, 7(10):1065–1074, Oct. 1996.
- [29] H. Wu, J. Jaffar, and R. Yap. A fast algorithm for scheduling instructions with deadline constraints on risc processors. In *Proc. Int. Conf. on Parallel Architectures and Compilation Techniques (PACT'00)*, page 281, Philadelphia, Pennsylvania, USA, Oct. 15–19. 2000.
- [30] T. Petrović. Jedan prilaz paralelizaciji izračunavanja matematičkih modela robota. Master's thesis, Matematički fakultet Univerziteta u Beogradu, Nov. 1992.
- [31] R. E. Graham, E. L. Lawler, J. K. Lenstra, and A. H. G. Rinnooy Kan. Optimization and approximation in deterministic sequencing and scheduling: a survey. *Ann. Discrete Math.*, 4:287–326, 1979.
- [32] B. Veltman, B. J. Lageweg, and J. K. Lenstra. Multiprocessor scheduling with communication delays. *Parallel Computing*, 16:173–182, 1990.
- [33] E. G. Coffman, Jr. and R. L. Graham. Optimal scheduling for two processor systems. *Acta Informatica*, 1:200–213, 1972.
- [34] V. Krishnamoorthy and K. Efe. Task scheduling with and without communication delays: A unified approach. *European Journal of Operational Research*, 89:366–379, 1996.
- [35] B. A. Malloy, E. L. Lloyd, and M. L. Soffa. Scheduling DAG's for asynchronous multiprocessor execution. *IEEE Trans. Parallel and Distributed Systems*, 5(5):498–508, May 1994.
- [36] T. Davidović. Inteligentni paralelizator. In *Zbornik Jug. Simp. YU Info*, Brezovica, 1995.
- [37] N. G. Hall and M. E. Posner. Sensitivity analysis for scheduling problems. (*revised for Operations Research*), 2002.
- [38] E. Mokotoff. An exact algorithm for the identical parallel machine scheduling problem. *Eur. J. Oper. Res.*, 152:758–769, 2004.
- [39] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman and Company, 1979.
- [40] R. L. Graham. Bounds on multiprocessor timing anomalies. *SIAM J. Applied Math.*, 17:416–429, 1969.
- [41] M. Dell'Amico and S. Martello. Optimal scheduling of tasks on identical parallel processors. *ORSA Journal on Computing*, 7:191–200, 1995.
- [42] M. Haouari, A. Gharbi, and M. Jemali. Tight bounds for the identical parallel machine scheduling problem. *Int. Trans. Oper. Res.*, 13:529–548, 2006.
- [43] T. Davidović, M. Šelmić, D. Teodorović, and D. Ramljak. Bee colony optimization for scheduling independent tasks to identical processors. *J. Heur.*, 18(4):549–569, 2012. DOI:10.1007/s10732-012-9197-3.
- [44] T. Davidović and S. Janićijević. Heuristic approach to scheduling independent tasks on identical processor. In *Proc. Symp. on information technology, YUINFO 2009, (on CD 115.pdf)*, Kopaonik, 2009.
- [45] T. Davidović and S. Janićijević. VNS for scheduling independent tasks on identical processor. In *Proc. 36th Symp. on Operational Research, SYM-OP-IS 2009.*, pages 301–304, Ivanjica, 2009.
- [46] A. Thesen. Design and evaluation of a tabu search algorithm for multiprocessor scheduling. *J. Heuristics*, 4(2):141–160, 1998.
- [47] S. Martello and P. Toth. Lower bounds and reduction procedures for the bin packing problem. *Discrete Applied Mathematics*, 28(1):59–70, 1990.
- [48] T. Davidović, L. Liberti, N. Maculan, and N. Mladenović. Towards the optimal solution of the multiprocessor scheduling problem with communication delays. In *Proc. 3rd Multidisciplinary Int. Conf. on Scheduling: Theory and Application*, Paris, France.
- [49] T. Davidović, L. Liberti, N. Maculan, and N. Mladenović. Mathematical programming-based approach to scheduling communicating tasks. *GERAD Tech. Report, G-2004-99, (submitted for publication)*.
- [50] R. Fortet. Applications de l'algèbre de boole en recherche opérationnelle. *Revue Française de Recherche Opérationnelle*, 4:17–26, 1960.
- [51] P. L. Hammer and S. Rudeanu. *Boolean Methods in Operations Research and Related Areas*. Springer, Berlin, 1968.
- [52] L. Liberti. Compact linearization for binary quadratic problems. *4OR*, 5(3):231–245, 2007.

Received

# РАЗВОЈ ЕЛЕКТРОНСКИХ УЏБЕНИКА ИЗ МАТЕМАТИКЕ

Дарко Дракулић

Филозофски факултет Универзитета у Источном Сарајеву  
ddrakulic@gmail.com

## АПСТРАКТ

Информационо-комуникационе технологије (ИКТ) постале су генератор промјена и начина на који људи живе, раде, играју се и уче. Живот у информационом друштву захтјева нешто више од основне писмености и класичних знања. Образовне институције морају одговорити овим потребама и омогућити стицање тих нових знања и вјештина. Такође, оне морају прихватити нове технологије као фундаментално наставно средство у процесу учења. Све више наставника почиње да примјењује информационо-комуникационе технологије у настави, али у већини случајева начин њиховог коришћења није одговарајући. Најчешћи примјер неадекватне примјене јесте коришћење неодговарајућих електронских материјала (презентација и слајдова) које наставници приказују ученицима и које ученици незаинтересовано (не)прате. Често се у настави користе скениране копије традиционалних уџбеника које се називају електронским уџбеницима, а које не представљају никакав технолошко-дидактички напредак у односу на класичне уџбенике.

У овом раду су описани концепти развоја електронских уџбеника који су резултат билатералног пројекта Универзитета у Марибору (Словенија) и Универзитета у Источном Сарајеву (Босна и Херцеговина) под називом "Интерактивно електронско учење математике у културном контексту народа Словеније и Босне и Херцеговине".

**Кључне ријечи:** електронско учење, електронски уџбеници, математика

Овај рад садржи резултате научно-истраживачког пројекта "Интерактивно електронско учење математике у Републици Српској" финансираног од стране Министарства науке и технологије Владе Републике Српске и пројекта "Интерактивно учење математике у културном контексту народа Словеније и Босне и Херцеговине" финансираног од стране Министарства цивилних послова Босне и Херцеговине.

## УВОД

Почетком 2010. године компанија *Apple* је представила уређај *iPad*, који је означио почетак велике експанзије електронских уређаја званих таблети (енглески: *tablet*, на српском се назива још и таблица). Таблете карактеришу велики екрани осјетљиви на додир, мала тежина, релативно ниска цијена, компактност и лакоћа коришћења. Од појаве *iPad*-а, таблети доживљавају велику експанзију у свим сферама електронске комуникације<sup>1</sup>, а заслужено заузимају и водећу улогу у процесу образовања.

Поред различитих платформи за учење на даљину, електронски материјали (е-матријали) и електронски уџбеници (е-уџбеници) представљају основ електронског образовања. Јужна Кореја је лидер пољу примјене ИКТ-а у образовању, гдје се тренутно у више од 60 школа активно користе електронски уџбеници. План корејске владе је да се до 2015. године заврши пројекат, вриједан 2.1 милијарду долара, чији је циљ стварање рачунарске мреже на којој ће се налазити различити образовни е-материјали којима ће ученици приступати помоћу таблета. Крајњи циљ овог пројекта је потпуна замјена класичних уџбеника са електронским уџбеницима [3].

<sup>1</sup> Студија Online Publishers Association (OPA) из јуна ове године је показала да 31% корисника Интернета у САД, старости од 8 до 64 година користи таблете [4]

У свијету се тренутно развија неколико платформи и стандарда за развој електронских уџбеника, као што су *ePUB* формат развијен од стране организације *International Digital Publishing Forum* (<http://idpf.org/epub>), затим језик за означавање *eLML* (*eLesson Markup Language*) којег развијају чланови швајцарског *eLearning* пројекта *GITTA* (<http://www.elml.org/>), језик за означавање  $\langle ML \rangle^3$  који се развија на Универзитету у Ростоку (<http://www.ml-3.org/>) те њемачки пројекат *Digitale schulbücher* (<http://www.digitale-schulbuecher.de>). Компанија *Apple* је почетком 2012. године такође представила другу генерацију алата *iBooks* који омогућава креирање и читање интерактивних електронских књига.

Истраживачи са Универзитета у Марибору се од 2001. године раде на развоју електронских материјала за учење математике, кроз пројекте *Mobid*, *E-UM* [2] (<http://www.e-um.si/>) и *e-уџбеники* (<http://eucbeniki.sio.si/>) а почетком 2012. године се у ово истраживање сеукључује и Универзитет у Источном Сарајеву, кроз билатерални пројекат "Интерактивно учење математике у електронском контексту народа Босне и Херцеговине и Словеније" (<http://eudzbenici.rs.ba/>). Циљ овог пројекта јесте развој електронских уџбеника који ће бити прилагођени наставним плановима школа у Словенији и Републици Српској. Овај рад садржи резултате досадашњих активности на поменутом пројекту.

## ЕЛЕКТРОНСКИ УЏБЕНИЦИ

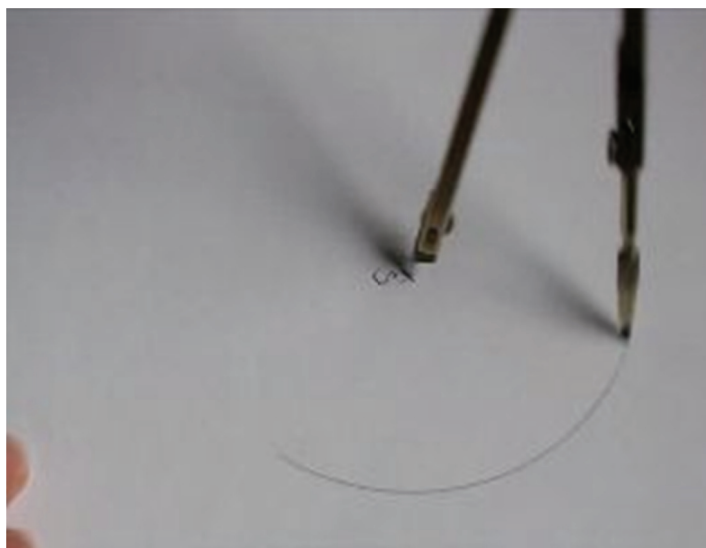
Као и у сваком процесу примјене нових метода и технологија, поставља се питање оправданости увођења електронских уџбеника у наставни процес. Образовни систем на овим просторима је врло конзервативан, инертан и неспреман на сваку промјену, а нарочито ако те промјене значајно одступају од постојећег стања. У овом случају поставља се питање: да ли образовном систему требају електронски уџбеници?

*Janna Quitney Anderson* са *Elon* универзитета је 2012. године у традиционалну подјелу генерација западног свијета увела нову генерацију - генерацију *АО* (*Always-On Generation*) [1]. Овој генерацији припадају дјеца рођена након 2000. године и њихове животе карактерише снажна повезаност са интернетом, тј. они интернет доживљавају и користе као врсту сопствене "екстерне меморије". Припаднике ове генерације карактеришу недостак стрпљена, неспособност апстракције и неспособност дубоког размишљања. Припадници ове генерације ће се задовољавати инстант рјешењима својих проблема која ће пронаћи на Интернету. Они неће листати књиге и енциклопедије, за њих ће интернет представљати примарни извор знања. Њима неће бити потребни уџбеници, њима ће бити потребни е-уџбеници.

С друге стране, примјена електронских уџбеника у настави математике је оправданија него у настави осталих предмета. Градиво из математике (као и из већине природних наука) често је апстрактно и због тога код ученика ствара одбојност према математици. Исправно коришћење адекватних анимација, симулација и дидактичких игара може помоћи ученицима у развијању апстрактног мишљења, као и у развијању интересовања за учење математике.

Електронски уџбеник није електронски (скенирани) облик традиционалног уџбеника. Иако не постоји општеприхваћена дефиниција електронског уџбеника, под електронским уџбеником се подразумијева дигитални документ који у себи садржи елементе класичног уџбеника (текст и слике) и интерактивне елементе - аудио/видео записе, различите тестове, симулације, аплете и дидактичке игре. Интерактивни елементи, према стопи интерактивности, могу се подијелити у три групе:

- Елементи са ниском стопом интерактивности - аудио и видео записи, просте анимације и симулације. Ови елементи имају само двије контроле - покретање и заустављање репродукције.



Слика 1. Елемент ниске стопе интерактивности - видео запис који приказује цртање кружнице помоћу шестара

- Елементи са средњом стопом интерактивности - основни тестови: да/не тестови, тестови вишеструког избора и тестови допуњавања. Корисници ових елемената имају могућност уноса одговора, провјере уноса и могућност приказивања тачних одговора.

**Допуни текст са ријечима:**

помјерали, конопац, нацртали, центар, скратили, затегнут

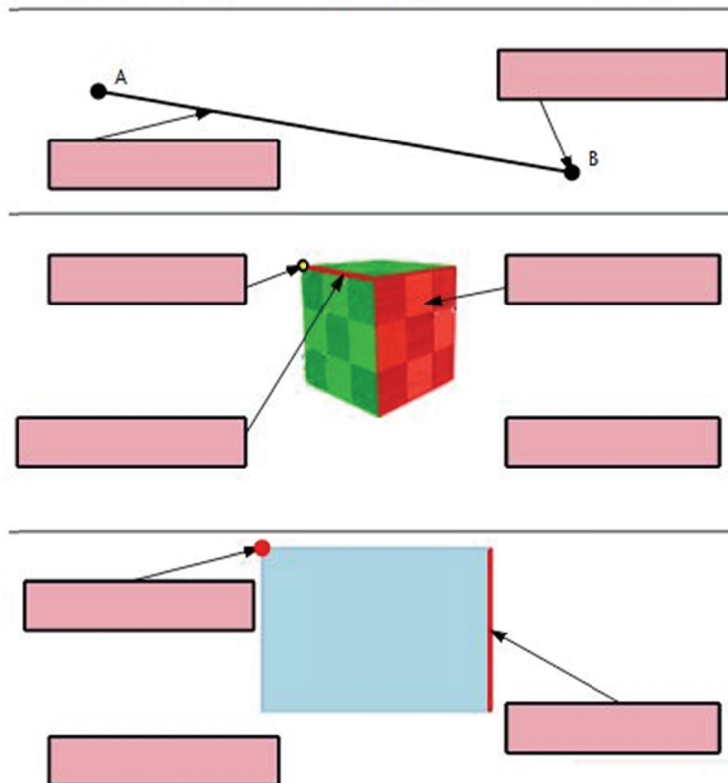
На оба штапа су завезали **центар**. Један штап су заболи у **конопац** круга који представља школско двориште. Са другим су **нацртали** границе прве гредице. Затим су конопацу **скратили** дужину. Штап у средини нису **нацртали**. При "цртању" су пазили да све вријеме конопац буде **затегнут**.

Слика 2. Елемент средње стопе интерактивности - тест допуњавања са унесеним одговорима, провјером унесених одговора и могућношћу приказа талних одговора

- Елементи са високом стопом интерактивности - напредне анимације и симулације (у којима корисник може активно да учествује), напредни тестови (тестови повезивања, тестови исправљања и сређивања), аплети и разне дидактичке игре. У овим елементима корисник може да динамички мијења изглед, садржаје, улазне и излазне податке као и да добија повратне информације у зависности од активности.

### Постави појмове у одговарајућа поља

- Геометријско тијело
- Геометријски лик
- Ивица коцке
- Тјеме правоугаоника
- Страница правоугаоника
- Дуж
- Крајња тачка дужи
- Страница коцке
- Тјеме коцке



Слика 3. Елемент високе стопе интерактивности - дидактичка игра

Електронски уџбеници, који се развијају у оквиру пројекта "Интерактивно електронско учење математике у културном контексту народа Словеније и Босне и Херцеговине" састоје се од електронских јединица, а свака јединица се састоји од пет дијелова: наслова, увода, главног дијела, дијела за понављање и задатака. Сви дијелови (осим наслова) се састоје од једне или више страница на којима се налази градиво везану за ту јединицу.

### ТЕХНОЛОГИЈА ИЗРАДЕ ЕЛЕКТРОНСКИХ УЏБЕНИКА

Основни технички захтјев израде електронских уџбеника јесте независност садржаја од његове репродукције, тј. независност електронских материјала од платформе. Разлог томе су различити стандарди које прописују различити оперативни системи (*Windows, iOS, Android*) као и немогућност рада појединих технологија на неким уређајима (нпр. немогућност рада *Java* и *Flash* апликација на таблетима).




Кружница
Кружница и круг
Центар
Полупречник
Узорци
Понављање
Задачи
Задачи
Задачи

Кружница и круг

### Кружница

Ученици су у ужи избор изабрали два приједлога за уређење школског дворишта.

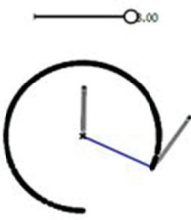


rudički  
 šiba  
 rdeče mini vrtnice  
 trava  
 bele vrtnice

Након гласања, изабран је други приједлог, са лавандом и бијелим ружама. Међутим, они су били забринути како да направе границе између редова цвијећа.

За рјешавање проблема учитељица им је дала два штапа и конопца.

Како им је успјело?



**Допуни текст са ријечима:**

памјерали, конопца, нацртали, центар, скратили, затегнут

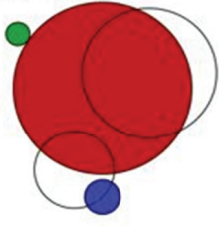
На оба штапа су завезали . Један штап су забили у  круга који представља школско двориште. Са другим су  границе преве греднице. Затим су конопцу  дужину. Штап у средини нису .

При "цртању" су пазили да све вријеме конопца буде .

Крива линија која ограничава круг, назива се **кружница**.

Колико је кругова на слици, а колико кружница?

- Пет кругова, три кружнице.
- Три круга, двије кружнице.
- Три круга, пет кружница.



Слика 4. Електронска јединица о кружници

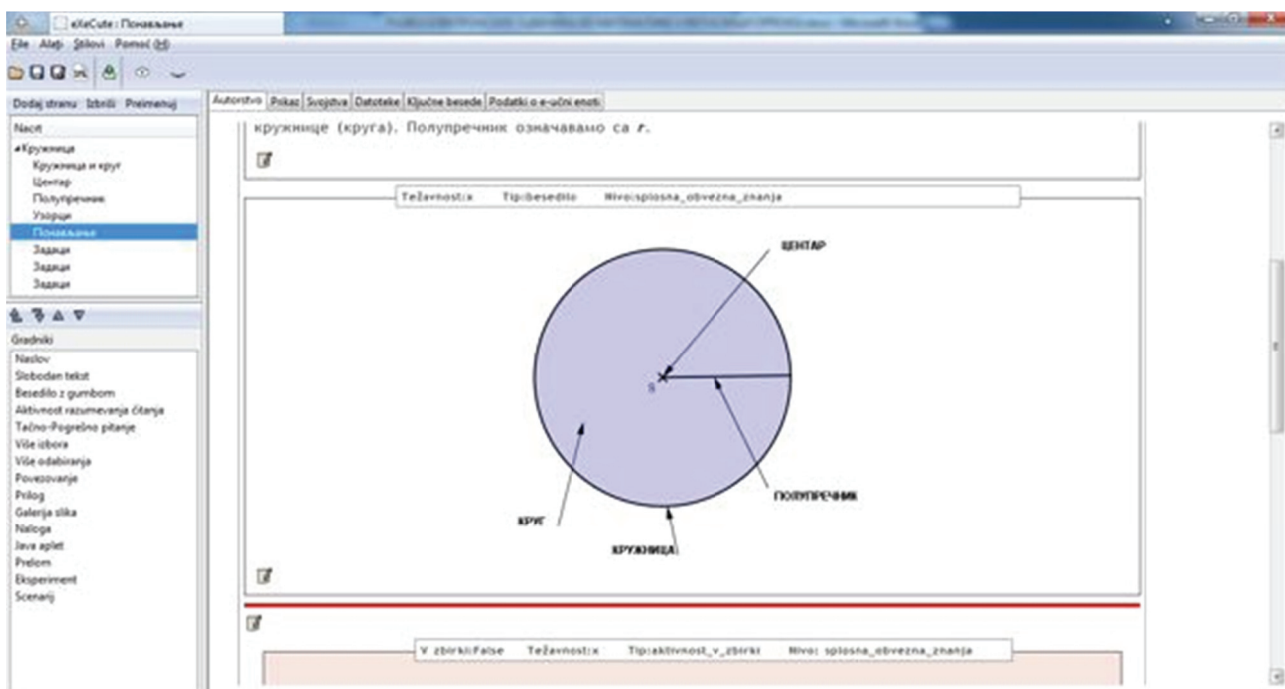
У пројектима Mobid и E-UM први алат за израду интерактивних елемената био је алат C.a.R. (<http://zirkel.sourceforge.net/>), којег је брзо замијенио алат GeoGebra (<http://www.geogebra.org>). Оба алата пружају могућност креирања одличних интерактивних елемената и веома су лаки за употребу, али се ослањају на *Java* технологију. Са појавом таблета, који немају подршку за *Java* технологију, интерактивни елементи креирани овим алатима су постали неупотребљиви. Осим тога, јављали су се стални проблеми са корисничким инсталацијама и верзијама *Java* виртуелне машине. Такође, неки од интерактивних елемената су се израђивали коришћењем *Flash* технологије, али ту се јављају исти проблеми као и са *Java* технологијом - не постоји подршка на таблетима и проблем инсталације и надоградње *Flash* читача. Невезано за неупотребљивост елемената на таблетима, у оба случаја поузданост рада електронских уџбеника се ослања на крајњег корисника, што је неприхватљиво.

Технологије које омогућавају развој свих потребних елемената електронских уџбеника, а које раде на свим платформама су *HTML*, *CSS*, *JavaScript* и *Ogg* видео формат (<http://xiph.org/>). Сви електронски уџбеници развијени у оквиру "Интерактивно електронско учење математике у културном контексту народа Словеније и Босне и Херцеговине" развијену су коришћењем поменутих технологија. За њихов приказ потребан је само веб претраживач.

Елементи класичног уџбеника (текст и слике) се лако имплементирају у *HTML*-у, док *JavaScript* и *Ogg* видео формат омогућавају имплементацију интерактивних елемената. Постоји мноштво *JavaScript* библиотека за израду математичких материјала, а у овдје су коришћене библиотека за интерактивну геометрију *JSXGraph* (<http://jsxgraph.uni-bayreuth.de/>) и библиотека *MathJax* за приказ математичког текста (<http://www.mathjax.org/>).

Тренутно постоји велики број алата за израду електронских материјала, као нпр. *W2L* (<http://www.what2learn.com/>), *xical* (<http://xical.org/>), *ClassTools* (<http://www.classtools.net/>) и *eXe* (<http://exelearning.org/wiki>). С обзиром да ниједан од поменутих алата не задовољава све потребе развоја електронског уџбеника из математике, на Универзитету у Марибору је развијена напредна верзија алата *eXe*, названа *eXeCute* [5]. Алат *eXeCute* представља

WYSIWYG<sup>2</sup> едитор у којем се на лак начин креирају електронски материјали и извозе у различите формате (*HTML*, *SCORM*, *IMS* итд.). Помоћу алата *eXeCute* аутори могу самостално да креирају већину електронских материјала, без потребе познавања напредних информатичких вјештина.

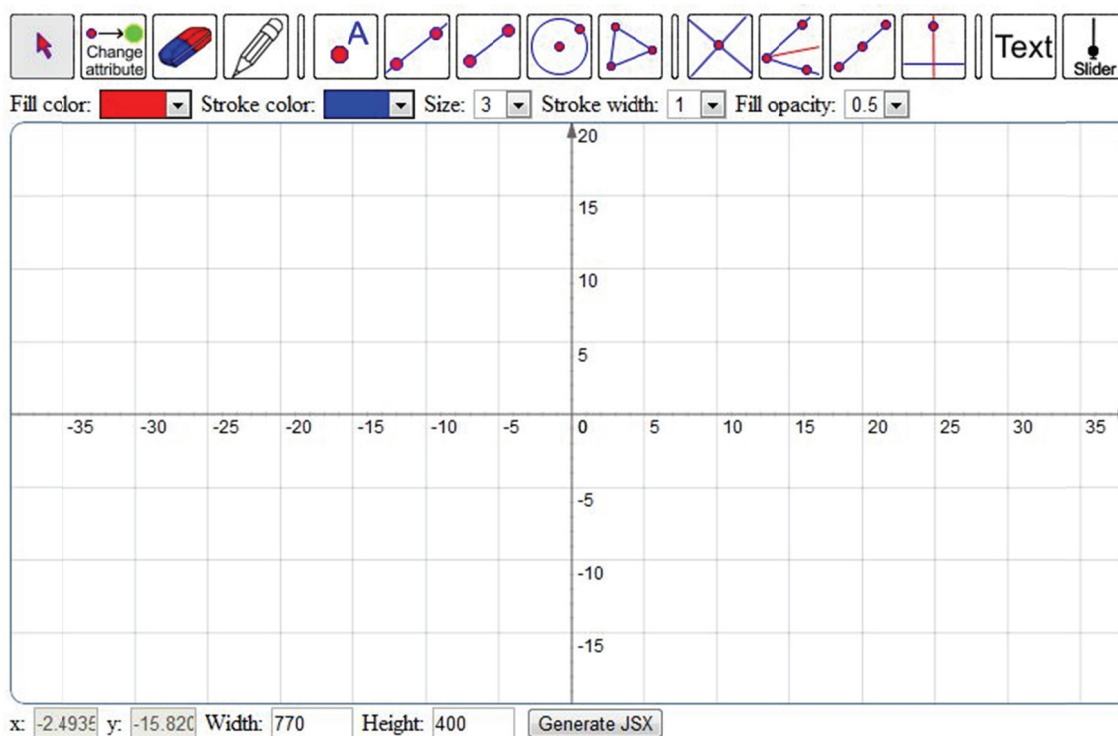


Слика 5. Алат *eXeCute*

## ПРОЦЕС ИЗРАДЕ ЕЛЕКТРОНСКИХ УЏБЕНИКА

Електронски уџбеници, као и остали уџбеници, имају уредника, ауторе, техничко особље, рецензенте и лекторе. Аутори већину елемената могу самостално да креирају коришћењем алата *eXeCute*, *GeoGebra* и програмима за обраду слике и видеа. Једини проблем ауторима представља израда интерактивних елемената у програмском језику *JavaScript*, па им је у овом послу неопходна помоћ техничког особља (програмера). Садашња пракса је да аутори креирају *GeoGebra* датотеке које представљају интерактивни елемент, које затим техничко особље конвертује у *JavaScript* (тј. поново их испрограмирају) или у *GIF* слике, уколико се ради о анимацијама. На Универзитету у Источном Сарајеву развија се неколико верзија графичког корисничког интерфејса са генераторима *JavaScript* кода за *JSXGraph* библиотеку помоћу којих аутори могу на лак начин да креирају основне *JavaScript* аплете.

<sup>2</sup> WYSIWYG - What You See Is What You Get, израз који се у рачунарству користи за опис система за уређење текста, слика и осталих елемената у којем се садржај током уређивања представља и сличној форми као оној која ће бити у коначном приказу.



Слика 6. *JSXGraph* графички кориснички интерфејс и генератор кода

Циљ прве фазе билатералног пројекта "Интерактивно електронско учење математике у културном контексту народа Словеније и Босне и Херцеговине" је израда уџбеника из математике за 4. и 6. разред основних школа у Републици Српској и Словенији. На овом послу ради преко 150 особа из обје државе (аутори, техничко особље, рецензенти, уредници и лектори) и посебан задатак јесте организација самог процеса израде уџбеника и размјена материјала између њих. У ту сврху развијен је портал за управљање електронским материјалима и садржајем електронских уџбеника и цјелокупан циклус израде електронских уџбеника се обавља преко тог портала. Циклус израде се састоји од постављања материјала од стране аутора, рада техничког особља на конвертовању *GeoGebra* датотека у *JavaScript* или *GIF* формат, рецензирања материјала, лекторисања и коначног прихватања електронске јединице од стране уредника књиге.

## ЗАКЉУЧАК

Пројекат "Интерактивно електронско учење математике у културном контексту народа Словеније и Босне и Херцеговине" је први пројекат овакве врсте на територији држава бивше Југославије и један од првих пројеката на подручју Европе. Тренутне активности на овом пројекту односе се на израду електронских књига за два разреда основних школа у Републици Српској и Словенији. Ови уџбеници су тренутно у *HTML* формату, а већ су започете активности на њиховом прилагођавању програмима за читање електронских књига на таблетима. Тренутно постоји мноштво таблет апликација за читање електронских књига (*iBooks*, *Aldiko eBook Reader*, *Kindle...*), међутим већина од њих има проблема са приказом математичких текстова и *JavaScript* аплета. Због наведеног, у плану је развој апликација за читање електронских уџбеника из математике за *Android* и *iPad* платформе.

Осим наставка рада на техничком дијелу пројекта, планирано је да се од следеће године електронски уџбеници уведу у наставу у неколико школа у Словенији и Републици Српској, како би се паралелно евалуирали резултати њиховог коришћења у наставном процесу.

## Литература

- [1] **J. Q. Anderson, L. Rainie** - Future of the Internet V - 2012 - *Pew Research Center's Internet & American Life Project*, [www.pewinternet.org](http://www.pewinternet.org), 29.2.2012.
- [2] **D. Kobal, B. Zmazek** - (E-)Mind Thinking with E-um, *Journal of Systemics, Cybernetics and Informatics*, Volume 6 - Number 2 - Year 2008, pp 56-59
- [3] **S. Kim** - South Korea ditching textbooks for tablet PCs, USA Today, [http://www.usatoday.com/tech/news/2011-07-20-south-korea-tablet-pc\\_n.htm](http://www.usatoday.com/tech/news/2011-07-20-south-korea-tablet-pc_n.htm), 20.7.2012.
- [4] **Online Publish Association** - OPA Study Reveals Attitudes of Today's Tablet User, [http://onlinepubs.ehclients.com/index.php/opa\\_news/press\\_release/opa\\_study\\_reveals\\_attitudes\\_of\\_todays\\_tablet\\_user](http://onlinepubs.ehclients.com/index.php/opa_news/press_release/opa_study_reveals_attitudes_of_todays_tablet_user), 18.6.2012.
- [5] **K. Prnaver, I. Pesek, B. Zmazek**. Online review system and authoring tools in the E-um project, 11th World Mulriconference on Systemics, Cybernetics and Informatics, Orlando, ZDA, julij 2007, WMSCI 2007 : proceedings. Vol. 1, 5 str.

# Istraživanjem do minimuma ili maksimuma

**Miodrag Mateljević**

*Matematički fakultet, Beograd, Studentski trg 16  
e-mail: miodrag@matf.bg.ac.rs*

**Nives Jozić**

*Filozofski fakultet, Split, Teslina 12  
e-mail: njozic@ffst.hr*

**Marek Svetlik**

*Matematički fakultet, Beograd, Studentski trg 16  
e-mail: svetlik@matf.bg.ac.rs*

**Apstrakt.** Život i rad u savremenom društvu zahtevaju nova znanja, veštine, sposobnosti, vrednosti i stavove, tj. nove kompetencije pojedinca, koje se odnose na razvoj stvaralaštva, kritičkog mišljenja i uvodjenja novina, informatičke pismenosti, sposobnosti rešavanja problema iz svakodnevnog života, socijalnih sposobnosti i veština itd.

U toku su promene tradicionalnog vaspitno-obrazovnog sistema koji deluje kao sredstvo prenošenja znanja.

Neki nosioci novih trendova predlažu da se matematičke kompetencije odnose na osposobljenost učenika za razvijanje i primenu matematičkog mišljenja u rešavanju problema u različitim svakodnevnim situacijama.

Problemi optimizacije se često pojavljuju u realnom životu i grubo rečeno odnose se na određivanje „najpovoljnijeg” rešenja; preciznije da na osnovu fiksiranih (datih) vrednosti nekih veličina odredimo minimalnu odnosno maksimalnu vrednost drugih veličina (u duhu novih trendova nazovimo ovu sposobnost: kompetencija „matematičkog odlučivanja”).

U ovom članku, predstavimo sadržaje koji su, po našem mišljenju, pogodni za razvijanje kompetencije „matematičkog odlučivanja”. Takođe, pokušaćemo objasniti kako obradivati te sadržaje da bi se postigao željeni ishod učenja i razvile sposobnosti „matematičkog odlučivanja”? Rutinsko rešavanje računskih zadataka utvrđuje naučene procedure, veštinu i brzinu računanja, ali vrlo malo razvija sposobnost „matematičkog odlučivanja”.

Da bi učenici razvili sposobnost nalaženja (naj)povoljnijeg rešenja, predložimo zadatke u kojima se učenik stavlja u situaciju da i) samostalno istražuje mogućnosti, sprovodi analizu i time razvija intuiciju za otkrivanjem traženog rešenja; preciznije ii) uočava činjenice, postavlja pretpostavke i izvodi zaključke; i iii) shvata potrebu za potvrđivanjem istinitosti uočenog zapažanja i postavljenih pretpostavki.

Tako na primer umesto da učenicima „serviramo” činjenicu da je od svih trouglova jednakog obima jednakostranični trougao najveće površine, možemo ih postaviti u situaciju da istraže i zaključe kako bi od date žice određene dužine napravili trougao najveće površine.

Jedan od ciljeva ovog rada je da se pokaže kako se učenici mogu voditi kroz istraživački proces u otkrivanju optimalnog rešenja i usmeravati na kritičko razmišljanje i učenje sa razumevanjem. Razmatraćemo probleme određivanja minimuma i maksimuma površine ili obima figura u ravni. Između ostalog u radu je dat i elementaran pristup izoperimetrijskoj nejednakosti.

**Ključne reči:** određivanje minimuma ili maksimuma; izoperimetrijski problem; učenje s razumevanjem

## 1. Uvod

Kada se u nastavi matematike razmatra pojam površine trougla obično se najpre pokaže (ili samo navede) da površinu trougla možemo izračunati tako što: pomnožimo dužinu stranice i dužinu visine koja odgovara toj stranici i taj proizvod podelimo sa 2. Odnosno, da važi formula:

$$P = \frac{av_a}{2} = \frac{bv_b}{2} = \frac{cv_c}{2}, \quad (1)$$

gde smo sa  $a$ ,  $b$  i  $c$  obeležili dužine stranica trougla a sa  $v_a$ ,  $v_b$  i  $v_c$  dužine odgovarajućih visina.

Dokaz je relativno jednostavan ako smatramo da su neka svojstva euklidske geometrije (npr. trougao možemo dopuniti do paralelograma) očigledna <sup>1</sup>.

No, postavlja se pitanje da li je formula (1) najpraktičnija, ako računamo površinu nekog modela trougla u realnom svetu? Na primer zamislimo sledeću situaciju: Limar je isekao komad lima oblika trougla čije su stranice  $15\text{cm}$ ,  $50\text{cm}$  i  $60\text{cm}$  i interesuje se kolika je površina isečenog lima.

Da bi izračunao površinu na osnovu formule (1) potrebno je da izmeri i dužinu bar jedne visine trougla, a to možda nije jednostavan zadatak. Naime, potrebno je znati šta je to visina, kako se konstruiše, a zatim i izmeriti dužinu visine.

Jednostavnije za limara je da izračuna površinu datog trougla samo na osnovu dužina stranica. Naime dužine stranica može jednostavno izmeriti. Formula za računanje površine trougla na osnovu dužina stranica glasi:

$$P = \sqrt{s(s-a)(s-b)(s-c)}, \quad s = \frac{a+b+c}{2}, \quad (2)$$

i naziva se Heronov obrazac.

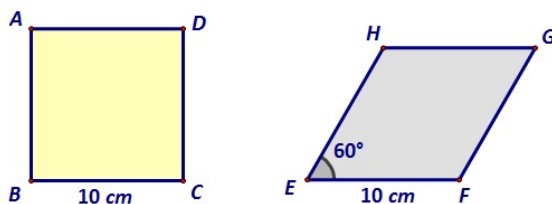
Koristeći kalkulator, limar može jednostavno izračunati površinu isečenog komada lima. Iako se obično smatra da je izvodjenje Heronovog obrasca komplikovanije od izvodjenja formule (1), u praktičnim problemima, kao što je navedeni, izgleda da je najpogodnije koristiti upravo Heronov obrazac.

Prirodno se nameće pitanje (a možemo očekivati i da ga limar postavi) da li i površinu četvorougla možemo izračunati samo na osnovu dužina stranica. Pre odgovora na postavljeno pitanje rešimo sledeći zadatak.

**Zadatak 1.1.**

- Izračunati površinu kvadrata čija je stranica dužine  $10\text{cm}$ .
- Izračunati površinu romba čija je stranica dužine  $10\text{cm}$ , a jedan ugao mere  $60^\circ$ .

**Rešenje.** Jednostavnim računom dolazimo do sledećeg rešenja: a)  $P = 100\text{cm}^2$  i b)  $P = 50\sqrt{3}\text{cm}^2$ . △



**Slika 1.** Kvadrat i romb jednakih dužina stranica

Uočimo da četvorouglovi razmatrani u zadatku 1.1 (slika 1) imaju različite površine, iako su sve stranice i jednog i drugog dužine  $10\text{cm}$ . Otuda zaključujemo da četvorougao, kao i površina četvorougla, nisu jedinstveno određeni ako su poznate samo dužine stranica. Prema tome, formula u kojoj je površina četvorougla izražena samo preko dužine stranica četvorougla ne postoji.

Rešavanjem zadatka 1.1 dobili smo još jedan rezultat. Naime, možemo primetiti da kvadrat čija je stranica dužine  $10\text{cm}$  ima veću površinu od romba čija je stranica dužine  $10\text{cm}$ , a jedan ugao mere  $60^\circ$ .

Štaviše iz formule  $P = a^2 \sin \alpha$ , gde je  $a$  dužina stranice a  $\alpha$  mera ugla romba, pomoću koje možemo izračunati površinu romba sledi da će od svih rombova, sa fiksiranom dužinom stranice, najveću površinu imati onaj kod kog je mera jednog (a samim tim i svih) ugla jednaka  $90^\circ$ , odnosno kvadrat. Zaključujemo: *U skupu svih rombova, zadate dužine stranice, najveću površinu ima kvadrat.*

Istraživanje možemo nastaviti rešavajući sledeći zadatak.

**Zadatak 1.2.** Date su četiri duži  $a$ ,  $b$ ,  $c$  i  $d$ , takve da se može konstruisati četvorougao čije su stranice upravo zadate duži. Kako od tih duži konstruisati četvorougao najveće površine?

Pre rešavanja ovog zadatka istražimo kako konstruisati, odnosno bar modelirati trougao najveće površine, ako je obim trougla unapred zadat.

<sup>1</sup>Postavlja se pitanje: Može li se analogan dokaz sprovesti u ne-euklidskim geometrijama? Za detalje videti sekciju 7.

## 2. Trougao najveće površine

U ovoj sekciji predlažemo jedan način kako u radu sa učenicima možemo postepeno, uz aktivno učešće i samostalna istraživanja učenika, doći do interesantnih činjenica vezanih za problem određivanja trougla najveće površine, pri unapred zadatom obimu.

### 2.1. Modeliranje trougla od žice

Za početak možemo svakom učeniku dati komad žice jednake dužine i zadužiti ga da modelira trougao. Očekujemo da će učenici napraviti modele trouglova različitih vrsta: raznostranične, jednakokrake i jednakostranične (slika 2).



Slika 2. Modeli trougla od žice

Možemo li takve modele trouglova uporediti? Po obliku? Po dužini stranica? Po veličini uglova? Po veličini površine? Razmotrimo obim i površinu.

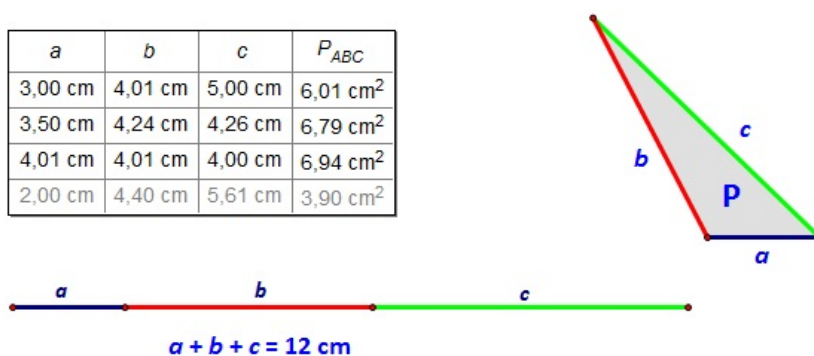
Šta možemo reći za obim? Kako su modelirani od žice jednake dužine, trouglovi imaju jednak obim. Da li su im i površine jednake? Pokušajmo doći do odgovora istraživanjem. U zavisnosti od toga sa kojim uzrastom učenika radimo, koriste se odgovarajuće formule za izračunavanje površine. U fazi istraživanja učenici mogu meriti dužine stranica i izračunavati odgovarajuće površine i na osnovu dobijenih rezultata izvoditi zaključke.

Primetimo da modeliranjem trougla od žice zadate dužine možemo dobiti beskonačno mnogo raznostraničnih trouglova, beskonačno mnogo jednakokrakih trouglova i tačno jedan jednakostranični trougao. Kako uporediti površine trouglova koje možemo dobiti na opisani način?

### 2.2. Modeliranje trougla u programu dinamičke geometrije

Uz modeliranje trougla pomoću žice za potrebe modeliranja možemo koristiti i program dinamičke geometrije<sup>2</sup>. Korišćenje programa dinamičke geometrije je pogodno, posebno stoga što u kratkom vremenu možemo modelirati mnogo trouglova i izračunati odgovarajuću površinu (slika 3).

$a$	$b$	$c$	$P_{ABC}$
3,00 cm	4,01 cm	5,00 cm	6,01 cm <sup>2</sup>
3,50 cm	4,24 cm	4,26 cm	6,79 cm <sup>2</sup>
4,01 cm	4,01 cm	4,00 cm	6,94 cm <sup>2</sup>
2,00 cm	4,40 cm	5,61 cm	3,90 cm <sup>2</sup>



Slika 3. Modeliranje i izračunavanje površine trougla

Na kraju ovog procesa možemo naslutiti sledeće: *Od svih trouglova datog obima najveću površinu ima jednakostranični trougao.*

<sup>2</sup>Pod programom dinamičke geometrije ovde podrazumevamo računarske programe namenjene statičkoj i dinamičkoj vizuelizaciji geometrijskih objekata kao što su na primer GeoGebra, Sketchpad, GCLC i drugi.



### 2.3. Upoređivanje površina jednakokrakog i raznostraničnog trougla

Obeležimo sa  $l$  zadati obim trougla i primetimo da je zbog nejednakosti trougla dužina svake stranice svakog trougla obima  $l$  strogo manja od  $l/2$ . Neka je  $a$  takvo da je  $0 < a < l/2$  i obeležimo sa  $\mathcal{F}_a$  familiju svih trouglova čiji je obim  $l$  i bar jedna stranica dužine  $a$ . Zbir dužina preostale dve stranice tada je  $l - a$ . Obeležimo razliku  $l - a$  sa  $l_a$ .

Pokušajmo najpre naslutiti koji od trouglova iz familije  $\mathcal{F}_a$  ima najveću površinu. Familiji  $\mathcal{F}_a$  pripada tačno jedan (do na podudarnost) jednakokraki trougao čija je osnovica dužine  $a$  (nazovimo taj trougao  $\tau_0$ , preciznije  $\tau_0(a)$ ) i beskonačno mnogo drugih trouglova. Primetimo da ako je  $a = l/3$ , trougao  $\tau_0$  je jednakokranični.

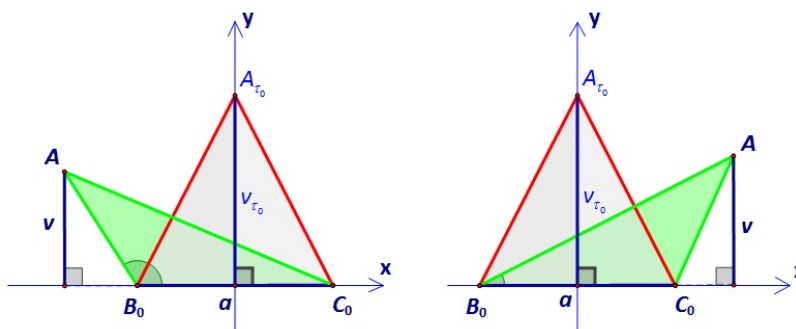
Pomoću programa dinamičke geometrije možemo eksperimentalno doći do zaključka koji od trouglova iz familije  $\mathcal{F}_a$  ima najveću površinu. U tom cilju na radnoj površini programa zadajmo koordinatni sistem  $xOy$  i konstruišimo tačke  $B_0(-a/2, 0)$  i  $C_0(a/2, 0)$ . Kako je  $|B_0C_0| = a$ , za proizvoljan trougao  $\tau$  iz familije  $\mathcal{F}_a$  možemo u gornjoj poluravni konstruisati tačku  $A_\tau$  takvu da je trougao  $\triangle A_\tau B_0 C_0$  podudaran trouglu  $\tau$ . Specijalno,  $\triangle A_{\tau_0} B_0 C_0$  je podudaran jednakokrakom trouglu  $\tau_0$ . Obeležimo sa  $v_\tau$  dužinu visine trougla  $\triangle A_\tau B_0 C_0$  koja odgovara stranici  $\overline{B_0 C_0}$ . Primetimo da je pri uvedenim oznakama  $A_{\tau_0} = A_{\tau_0}(0, v_{\tau_0})$ .

Neka je  $E_a = \{A_\tau : \tau \in \mathcal{F}_a\}$ .

Kako smo prethodno opisanim postupkom svakom trouglu iz familije  $\mathcal{F}_a$  dodelili podudaran trougao u koordinatnom sistemu programa, dovoljno je odrediti za koji „položaj” tačke  $A \in E_a$ , trougao  $\triangle A B_0 C_0$  ima najveću površinu. U tom cilju dopustimo da se tačka  $A$  kreće duž skupa  $E_a$  (kasnije ćemo videti da je putanja tačke  $A$  poluelipsa).

Obeležimo sa  $v$  dužinu visine trougla  $\triangle A B_0 C_0$  koja odgovara stranici  $\overline{B_0 C_0}$ .

Ako se tačka  $A$  kreće iz položaja  $A_{\tau_0}$  tako da se  $\angle A B_0 C_0$  povećava, stranica  $\overline{B_0 A}$  i  $v$  se smanjuju. S druge strane, ako se tačka  $A$  kreće iz „položaja”  $A_{\tau_0}$  tako da se  $\angle A B_0 C_0$  smanjuje, stranica  $\overline{B_0 A}$  se povećava a  $v$  se smanjuje (videti sliku 4).



Slika 4. Kretanje tačke  $A$

Otuda zaključujemo da je  $v \leq v_{\tau_0}$  (pri čemu jednakost važi ako i samo ako je  $A = A_{\tau_0}$ ), odnosno korišćenjem formule (1), zaključujemo da je površina trougla  $\tau_0$  veća do jednaka od površine proizvoljnog trougla  $\tau$  iz familije  $\mathcal{F}_a$ . Pri tome površina trougla  $\tau_0$  je strogo veća od površine trougla  $\tau$ , ako trougao  $\tau$  nije podudaran trouglu  $\tau_0$ .

Na osnovu prethodnih razmatranja formulišemo sledeće tvrdjenje.

**Tvrdjenje 1.** Od svih trouglova zadatog obima  $l$ , kojima je dužina jedne stranice  $a$  ( $0 < a < l/2$ ) najveću površinu ima jednakokraki trougao čija je osnovica dužine  $a$ .

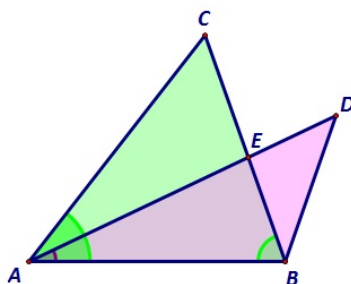
Sintetički („čisto” geometrijski, bez korišćenja algebre i analitičke geometrije) dokaz tvrdjenja 1 može se bazirati na sledećoj lemi (slika 5).

**Lema 1.** Neka su  $\triangle ABC$  i  $\triangle ABD$  dva trougla takva da važi:

- tačke  $C$  i  $D$  su sa iste strane prave  $AB$ ,
- $|AC| + |CB| = |AD| + |DB|$ ,
- $\angle BAD < \angle BAC \leq \angle CBA$ .

Tada je površina trougla  $\triangle ABC$  strogo veća od površine trougla  $\triangle ABD$ .





Slika 5. Uporedjivanje površine trouglova

*Dokaz leme 1.* Pre svega, primetimo da iz uslova  $\angle BAD < \angle BAC$  sledi  $C \neq D$ , pa se prave  $AD$  i  $BC$  seku u jednoj tački. Tačku preseka pravih  $AD$  i  $BC$  obeležimo sa  $E$ .

Nadalje tvrdimo da je  $\angle CBA < \angle DBA$ . Drugim rečima, od četiri ugla ovih trouglova, kojima jedan krak pripada pravoj  $AB$ , najmanji i najveći od njih pripadaju istom trouglu ( $\triangle ABD$ ). Da bismo ovo dokazali pretpostavimo suprotno, tj. da važi:  $\angle CBA \geq \angle DBA$ . Moguća su dva slučaja:

i)  $\angle CBA = \angle DBA$  (slika 6)

U ovom slučaju tačka  $D$  pripada duži  $\overline{BC}$ , tj.  $D \equiv E$ . Koristeći nejednakost trougla za dužine stranica trougla  $\triangle ADC$  dobijamo  $|AD| < |AC| + |CD|$  i otuda:

$$|AD| + |DB| < (|AC| + |CD|) + |DB| = |AC| + (|CD| + |DB|) = |AC| + |CB|,$$

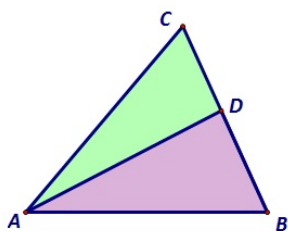
a to je u kontradikciji sa uslovom iskazanim pod b).

ii)  $\angle CBA > \angle DBA$  (slika 7)

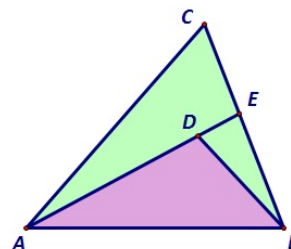
U ovom slučaju tačke  $A$  i  $D$  su sa iste strane prave  $BC$ . Koristeći dva puta nejednakost trougla (u trouglovima  $\triangle DBE$  i  $\triangle AEC$ ) dobijamo:

$$|AD| + |DB| < |AD| + (|DE| + |EB|) = |AE| + |EB| < (|AC| + |CE|) + |EB| = |AC| + |CB|,$$

što je opet u kontradikciji sa uslovom iskazanim pod b).



Slika 6. Uporedjivanje uglova 1



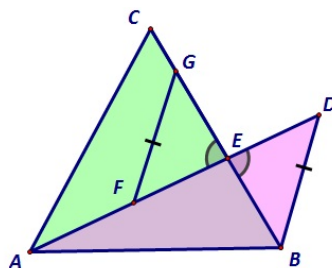
Slika 7. Uporedjivanje uglova 2

Budući, na osnovu pokazanog, pretpostavka  $\angle CBA \geq \angle DBA$  nije održiva sledi da je  $\angle CBA < \angle DBA$ , što je i trebalo dokazati.

Konstruišimo sada tačke  $F$  i  $G$  na sledeći način: tačka  $F$  pripada polupravoj  $EA$  i važi  $|EF| = |EB|$ , a tačka  $G$  pripada polupravoj  $EC$  i važi  $|EG| = |ED|$  (slika 8). Kako je  $\angle BED = \angle GEF$  (unakrsni uglovi),  $|EF| = |EB|$  i  $|EG| = |ED|$  sledi  $\triangle FEG \cong \triangle BED$ , a samim tim i  $|FG| = |BD|$ .

Dalje tvrdimo: tačka  $F$  pripada duži  $\overline{AE}$  i različita je od tačke  $A$ , a tačka  $G$  pripada duži  $\overline{EC}$  i različita je od tačke  $C$ .

Naime, u trouglu  $\triangle ABE$  na osnovu uslova c) važi  $\angle BAE < \angle EBA$  te je  $|EB| < |EA|$ , tj.  $|EF| < |EA|$ . To znači da je tačka  $F$  između tačaka  $A$  i  $E$ .

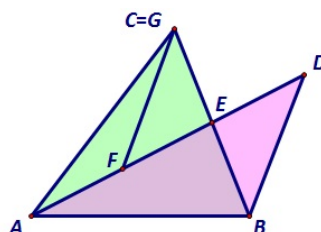


Slika 8. Tačke  $F$  i  $G$

Ako bi važiolo  $G \equiv C$  (slika 9), onda bismo koristeći nejednakost trougla  $\triangle AFC$ , odnosno trougla  $\triangle AFG$  i uslove konstruisanja tačaka  $F$  i  $G$ , dobili sledeće:

$$|AC| + |CB| < (|AF| + |FC|) + |CB| = |AF| + |BD| + |FD| = |AD| + |DB|,$$

što je u kontradikciji sa uslovom iskazanim pod b).

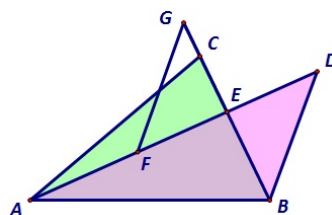


Slika 9. Slučaj  $G \equiv C$

Ako bi važiolo da je tačka  $C$  između tačaka  $G$  i  $E$  (slika 10), onda bismo, koristeći uslov b) i uslove konstruisanja tačaka  $F$  i  $G$ , dobili sledeće:

$$\begin{aligned} |AC| + |CB| &= |AD| + |DB| \\ |AC| + (|BE| + |EG| - |GC|) &= (|AF| + |FE| + |ED|) + |FG| \\ |AC| + |FE| + |ED| - |GC| &= |AF| + |FE| + |ED| + |FG| \\ |AC| &= |AF| + |FG| + |GC|. \end{aligned}$$

To bi značilo da tačke  $F$  i  $G$  pripadaju duži  $\overline{AC}$ , što je u suprotnosti sa uslovima konstruisanja tih tačaka.



Slika 10. Slučaj kada je tačka  $C$  između tačaka  $G$  i  $E$

Konačno, kako su trouglovi  $\triangle FEG$  i  $\triangle BED$  podudarni, kako tačka  $F$  pripada duži  $\overline{AC}$  i različita je od tačke  $A$ , i kako tačka  $G$  pripada duži  $\overline{BC}$  i različita je od tačke  $C$ , zaključujemo da je površina trougla  $\triangle ABC$  strogo veća od površine trougla  $\triangle ABD$ .  $\square$

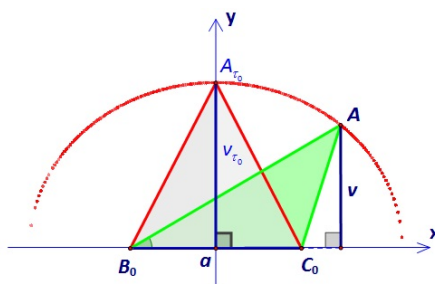
Vratimo se sada dokazu tvrdjenja 1.

*Dokaz tvrdjenja 1.* Neka je  $\triangle ABC$  jednakokraki trougao obima  $O$  i osnovice  $\overline{BC}$  dužine  $a$ . Dalje, neka je trougao  $\triangle DBC$  različit od trougla  $\triangle ABC$ , ali jednakog obima kao i trougao  $\triangle ABC$ . Bez umanjjenja opštosti možemo pretpostaviti da su tačke  $A$  i  $D$  sa iste strane prave  $BC$ .

Bar jedan od uglova  $\angle CBD$  i  $\angle DCB$  je manji od ugla  $\angle CBA$  (koji je jednak uglu  $\angle ACB$ ). Zaista, ako bi oba ugla  $\angle CBD$  i  $\angle DCB$  bila veća od ugla  $\angle CBA$  onda ne bi važila jednakost  $|CA| + |AB| = |CD| + |DB|$ .

Time su ispunjena sva tri uslova leme 1 i korišćenjem iste zaključujemo da je površina trougla  $\triangle ABC$  strogo veća od površine trougla  $\triangle DBC$ .  $\square$

Dokaz tvrdjenja 1 se može bazirati i na definiciji elipse kao geometrijskog mesta tačaka. Kako za svaku tačku  $A_\tau$  važi da je zbir dužina duži  $\overline{B_0A_\tau}$  i  $\overline{C_0A_\tau}$  konstantan i jednak  $l_a$  (ponovimo,  $l_a = l - a$ ), zaključujemo da svaka tačka  $A_\tau$  pripada elipsi čiji su fokusi tačke  $B_0$  i  $C_0$  tj. elipsi  $\mathcal{E} = \{T : |B_0T| + |C_0T| = l_a\}$  (slika 11).

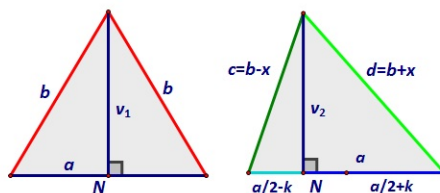


Slika 11. Putanja tačke  $A$

Ako elipsu  $\mathcal{E}$  predstavimo u koordinatnom sistemu  $xOy$  tako da su koordinate fokusa  $B_0(-a/2, 0)$  i  $C_0(a/2, 0)$ , onda je jednačina elipse  $\frac{x^2}{\alpha^2} + \frac{y^2}{\beta^2} = 1$ , pri čemu je  $\alpha = l_a/2$  a  $\beta = \sqrt{(l_a/2)^2 - (a/2)^2}$ . Primitimo da na osnovu Pitagorine teoreme važi  $\beta = v_{\tau_0}$ .

Nadalje, imajući u vidu razmatranje koje smo sproveli korišćenjem programa dinamičke geometrije, možemo posmatrati samo gornju polovinu elipse. Neka je  $\mathcal{E}_+ = \{T(x, y) \in \mathcal{E} : y > 0\}$  i neka je  $A \in \mathcal{E}_+$ . Imajući u vidu koordinate tačaka  $B_0$  i  $C_0$  na osnovu formule (1), sledi da trougao  $\triangle AB_0C_0$  ima najveću površinu kada je vrednost  $y$  koordinate tačke  $A$  najveća. Kako je  $y = \beta\sqrt{1 - \frac{x^2}{\alpha^2}}$ , zaključujemo da  $y$  najveće kada je vrednost izraza pod korenem najveća, odnosno kada je  $x = 0$ . Otuda sledi da trougao  $\triangle AB_0C_0$  ima najveću površinu kada je  $A = A(0, \beta)$ , odnosno trougao  $\triangle AB_0C_0$  ima najveću površinu kada je visina koja odgovara stranici  $\overline{B_0C_0}$  podudarna sa malom poluosom poluelipse  $\mathcal{E}_+$ . Kada je visina trougla  $\triangle AB_0C_0$  koja odgovara stranici  $\overline{B_0C_0}$  podudarna sa malom poluosom poluelipse  $\mathcal{E}_+$  trougao  $\triangle AB_0C_0$  je jednakokraki (sa osnovicom  $\overline{B_0C_0}$ ) a samim tim i podudaran sa trouglom  $\tau_0$ .

Algebarski, problem određivanja trougla najveće površine u familiji  $\mathcal{F}_a$ , možemo razmatrati na sledeći način. Pri tome koristimo oznake kao na slici 12.



Slika 12. Jednakokraki i raznostranični trougao

Bez umanjjenja opštosti možemo pretpostaviti da je  $x > 0$ . Kako je  $x > 0$  sledi da je i  $k > 0$ . Koristeći formulu (1) za površinu jednakokrakog trougla  $P_1$  dobijamo  $P_1^2 = \frac{a^2}{4} \left(b^2 - \frac{a^2}{4}\right)$ , a za površinu odgovarajućeg

raznokraničnog trougla  $P_2$  dobijamo  $P_2^2 = \frac{a^2}{4} \left( (b+x)^2 - \left( \frac{a}{2} + k \right)^2 \right)$ . Primenom Pitagorine teoreme dobijamo  $v_2 = \sqrt{(b-x)^2 - (a/2 - k)^2} = \sqrt{(b+x)^2 - (a/2 + k)^2}$ , odnosno  $2bx = ak$ . No kako je  $2b > a$  (nejednakost trougla) na osnovu jednakosti  $2bx = ak$  sledi da je  $k > x$ . Raspisivanjem izraza za  $P_2^2$  i koristeći prethodne zaključke dobijamo  $P_2^2 = P_1^2 - \frac{a^2}{4}(k^2 - x^2)$ . Kako je  $k > x$  to je  $k^2 - x^2 > 0$  pa je površina  $P_2$  manja od površine  $P_1$ , tj. površina jednakokrakog trougla veća je od površina svih odgovarajućih raznostraničnih trouglova.

#### 2.4. Upoređivanje površina jednakokraničnog i jednakokrakog trougla

Primitimo da za svaki raznostranični trougao postoji jednakokraki trougao istog obima, čija je osnovica jednaka jednoj stranici datog raznostraničnog trougla. Otuda možemo zaključiti da za svaki raznostranični trougao postoji odgovarajući jednakokraki trougao jednakog obima a veće površine.

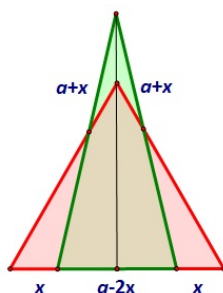
Dakle, da bismo odredili trougao najveće površine pri fiksiranom obimu, preostaje da uporedimo površinu jednakokraničnog (jedinственog za fiksirani obim) trougla sa površinom proizvoljnog jednakokrakog trougla istog obima.

Neka je stranica jednakokraničnog trougla dužine  $a$ . Uočimo da su moguća dva slučaja:

- i) Dužina osnovice jednakokrakog trougla je manja od  $a$ .
- ii) Dužina osnovice jednakokrakog trougla je veća od  $a$ .

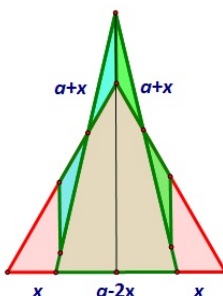
Kako bismo uporedili odgovarajuće površine, „položimo” osnovicu jednakokrakog trougla na jednu stranicu jednakokraničnog trougla tako da se odgovarajuće visine „polaže” jedna na drugu.

i) Ako je dužina osnovice jednakokrakog trougla manja od  $a$  za  $2x$  ( $x > 0$ ), tj. ako je dužina osnovice  $a - 2x$ , tada će krakovi biti dužine  $a + x$  (slika 13).



Slika 13. Slaganje trouglova

Deo jednakokrakog trougla nalazi se izvan jednakokraničnog trougla, a razmatrana visina ga deli na dva manja (podudarna) trougla. Rotirajmo ta dva trougla za  $180^\circ$  oko tačke preseka stranica jednakokrakog i jednakokraničnog trougla (tačke preseka se nalaze iznad središta stranica jednakokraničnog trougla). Ovom transformacijom smo tim trouglovima, koji su se nalazili izvan jednakokraničnog trougla, „popunili” još jedan njegov deo, ali ne i ceo jednakokraničan trougao. Otuda možemo zaključiti da je u ovom slučaju površina jednakokraničnog trougla veća od površine odgovarajućeg jednakokrakog trougla jednakog obima (slika 14).



Slika 14. Trouglovi nakon rotiranja

Algebarski bismo problem mogli razmatrati na sledeći način. Ako je stranica jednakostraničnog trougla dužine  $a$ , tada je dužina visine  $v = \frac{a\sqrt{3}}{2}$ , tj.  $v^2 = \frac{3a^2}{4}$ , a dužina visine jednakokrakog trougla je  $v_1 = \sqrt{v^2 + 3ax}$ , tj.  $v_1^2 = v^2 + 3ax$ . Uporedimo kvadrate površina koristeći formulu (1). Naime,

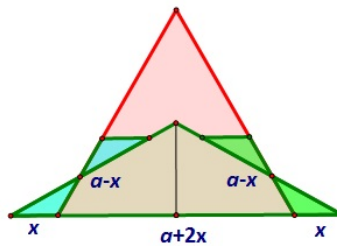
$$P_1^2 = \frac{a^2 v^2}{4},$$

odnosno

$$P_2^2 = \frac{a_1^2 v_1^2}{4} = \left(\frac{a}{2} - x\right)^2 v_1^2 = \left(\frac{a}{2} - x\right)^2 (v^2 + 3ax) = P_1^2 - 3ax^2 \left(\frac{3a}{4} - x\right),$$

pa kako je  $2x < a$  to je  $x < \frac{a}{2} < \frac{3a}{4}$ , pa se kvadrat površine  $P_2$  dobija oduzimanjem pozitivnog broja od kvadrata površine  $P_1$ . Zaključujemo da je  $P_1^2 > P_2^2$  a samim tim i  $P_1 > P_2$ .

ii) Ako je dužina osnovice jednakokrakog trougla veća od  $a$  za  $2x$  ( $x > 0$ ), tj. ako je dužina osnovice  $a + 2x$ , tada će krakovi biti dužine  $a - x$ . Analognim razmatranjem kao u slučaju i) dolazimo do istog zaključka kao u slučaju i) (slika 15).

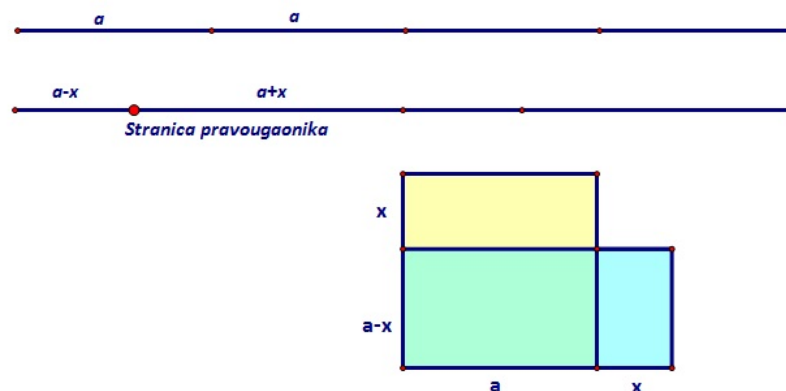


Slika 15. Trouglovi nakon rotiranja

Dakle, od svih jednakokrakih trouglova jednakog obima najveću površinu ima jednakostranični trougao. Konačno smo dokazali: *Od svih trouglova jednakog obima najveću površinu ima jednakostranični trougao.*

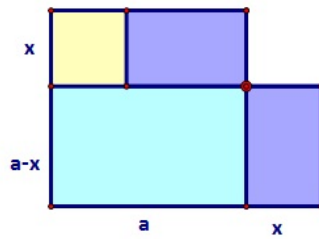
### 3. Paralelogram najveće površine

Uzmimo sada dva komada žice jednake dužine i modelirajmo jedan kvadrat i jedan pravougaonik. Stavimo „jedan preko drugog” kao na slici 16. i uporedimo ih.



Slika 16. Uporedjivanje kvadrata i pravougaonika

Deo pravougaonika koji je izvan kvadrata premostimo u deo kvadrata koji je preostao izvan pravougaonika. Time smo „popunili” još jedan deo kvadrata, ali ne i ceo kvadrat (slika 17).



Slika 17. Pravougaonik i kvadrat nakon transformacije

Primitimo da modeliranjem kvadrata i pravougaonika, od žice zadate dužine, možemo dobiti beskonačno mnogo pravougaonika, ali samo jedan kvadrat. Otuda zaključujemo: *Površina kvadrata veća je od površine bilo kog drugog pravougaonika jednakog obima.*

Algebarski bismo problem mogli razmatrati na sledeći način. Ako je dužina stranice kvadrata jednaka  $a$  onda su dužine stranica odgovarajućeg pravougaonika jednake  $a + x$  i  $a - x$ , pri čemu je  $0 < x < a$ . Nadalje, neka je  $P_1$  površina kvadrata, a  $P_2$  površina pravougaonika. Tada važi:

$$P_1 = a^2$$

$$\text{i } P_2 = (a + x)(a - x) = a^2 - x^2.$$

Otuda, kako je  $0 < x < a$ , sledi da je  $P_1 > P_2$ .

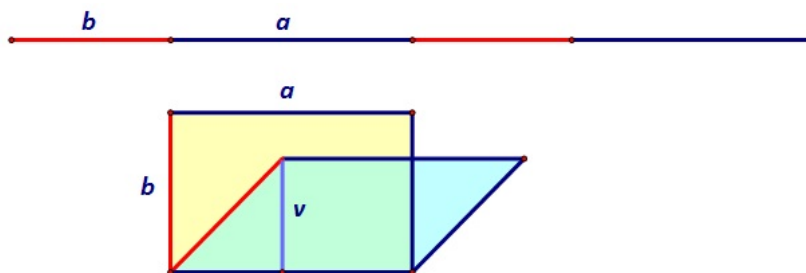
Razmotrimo ostale vrste paralelograma. Uporedimo npr. pravougaonik i paralelogram jednakog obima i jednakih dužina stranica (slika 18). Ako sa  $a$  i  $b$  obeležimo dužine stranica pravougaonika i paralelograma, a sa  $P_1$  i  $P_2$  odgovarajuće površine onda važi:

$$P_1 = ab$$

$$\text{i } P_2 = av_a,$$

pri čemu je  $v_a$  dužina odgovarajuće visine paralelograma.

Kako je  $b > v_a$ , zaključujemo da je  $P_1 > P_2$ .



Slika 18. Uporedjivanje pravougaonika i paralelograma

Uočimo da u odnosu na razmatrani pravougaonik imamo beskonačno mnogo odgovarajućih paralelograma. Dakle, pri fiksiranim dužinama stranica, od svih paralelograma najveću površinu ima pravougaonik. Dalje, kako za svaki pravougaonik postoji kvadrat istog obima i kako je površina tog kvadrata veća od površine pravougaonika zaključujemo: *Od svih paralelograma jednakog obima najveću površinu ima kvadrat.*

Daljim istraživanjem mogli bismo doći do zaključka: *Od svih četvorouglova jednakog obima najveću površinu ima kvadrat.*

Vratimo se sada na zadatak 1.2. Najpre primetimo da su u postavci zadatka fiksirane i dužine stranica četvorougla a ne samo obim.

Na osnovu Bretšnajerove formula za određivanje površine četvorougla nalazimo

$$P = \sqrt{(s-a)(s-b)(s-c)(s-d) - abcd \cos^2 \frac{\alpha + \gamma}{2}},$$

pri čemu su  $a, b, c$  i  $d$  dužine stranica četvorougla,  $s = \frac{a+b+c+d}{2}$ , a  $\alpha$  i  $\gamma$  dva naspramna ugla u četvorouglu.

Primetimo da se površina četvorougla, ako su dužine stranica fiksirane (što je ispunjeno u postavci zadatka 1.2), menja samo u zavisnosti od promene poluzbira dva napramna ugla. Tako će površina biti veća, ako je vrednost izraza  $\cos^2 \frac{\alpha + \gamma}{2}$  manja. Odnosno površina je će biti najveća kada je  $\alpha + \gamma = 180^\circ$ .

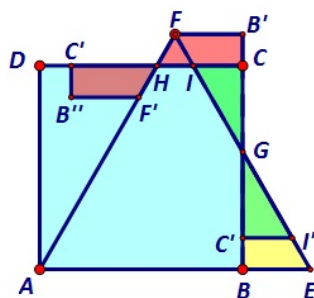
No, kako je uslov  $\alpha + \gamma = 180^\circ$  potreban i dovoljan da četvorougao bude tetivan sledi da od svih četvorouglova čije su stranice zadate dužine najveću površinu ima tetivan četvorougao.

Zainteresovanom čitaocu ostavljamo da dokaže: Ako su date četiri duži takve da se može konstruisati četvorougao čije su to stranice, onda se može konstruisati i tetivan četvorougao čije su stranice zadate duži.

#### 4. Jednakostranični trougao i kvadrat

Uporedimo sad jednakostranični trougao (on je medju svim trouglovima najveće površine) i kvadrat (on je medju svim četvorouglovima najveće površine) jednakog obima.

Kako bismo geometrijski uporedili njihove površine, trougao „položimo” na kvadrat kao na slici 19, a zatim vršimo rotaciju onih delova trougla koji se nalaze izvan kvadrata tako da se preslikaju unutar dela kvadrata koji je izvan trougla.



Slika 19. Uporedjivanje trougla i kvadrata

Sa slike možemo uočiti da je površina kvadrata veća od površine trougla za površinu petougla  $AF'B''C'D$ . Konačno se može izvesti zaključak da kvadrat ima veću površinu od jednakostraničnog trougla jednakog obima.

Dalje se mogu upoređivati kvadrat i pravilni petougao jednakog obima, pravilni petougao i pravilni šestougao jednakog obima itd. U svim slučajevima došli bismo do zaključka da veću površinu ima onaj  $n$ -tougao koji ima više stranica.

Konačno, kako kružnicu intuitivno, ali ne i u potpunosti strogo, možemo videti kao pravilan  $n$ -tougao za  $n = \infty$ , naslućujemo da će krug imati veću površinu od ma kog  $n$ -tougla istog obima.

#### 5. Izoperimetrijski problem

Problem da se medju svim ravanskim figurama datog obima odredi figura najveće površine naziva se izoperimetrijski problem.

U prethodnoj sekciji smo naslutili da za zadati obim, krug ima veću površinu od ma kog  $n$ -tougla tog obima. Prirodno se nameće pitanje: Da li ako razmatramo sve ravanske figure datog obima i dalje krug ima najveću površinu? Odgovor je potvrđan. Štaviše, važi izoperimetrijska nejednakost:

$$4\pi P \leq l^2,$$

gde smo sa  $l$  obeležili dužinu proste zatvorene ravanske krive a sa  $P$  površinu figure koju ta kriva ograničava. Pri tome, u izoperimetrijskoj nejednakosti, jednakost važi ako i samo ako je razmatrana prosta zatvorena kriva

upravo kružnica.

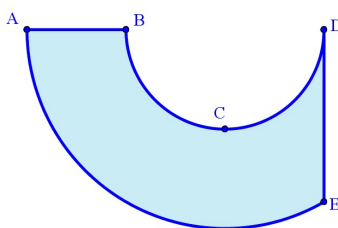
Dajemo sada i skicu jednog rešenja izoperimetrijskog problema.

Neka je  $\mathcal{F}$  familija svih ravanskih figura čiji je obim jednak  $l$ . Dalje, sa  $P(\Phi)$  obeležavamo površinu figure  $\Phi$  a sa  $l(\gamma)$  dužinu krive  $\gamma$ . Tako da, ako je kriva  $\gamma$  granica figure  $\Phi$  onda je  $l(\gamma)$  obim figure  $\Phi$ .

Dokažimo da za svaku figuru  $\Phi \in \mathcal{F}$  koja nije krug postoji figura  $\Omega \in \mathcal{F}$  takva da je  $P(\Omega) > P(\Phi)$ .

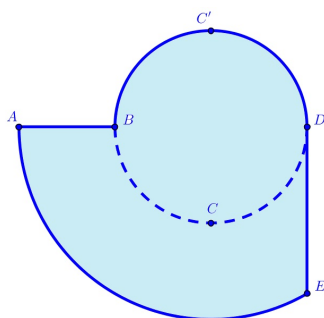
i) Pretpostavimo prvo da figura  $\Phi$  nije konveksna. U tom slučaju postoji konveksna figura  $\Omega$  takva da je  $P(\Omega) > P(\Phi)$ . Dokaz ove činjenice, u opštem slučaju, nije jednostavan. U ovom radu ilustrujemo samo ideju dokaza na jednom primeru.

Neka je  $\Phi$  figura na slici 20.



**Slika 20.** Primer figure koja nije konveksna

Figuru  $\Phi_1$  na slici 21 dobijamo tako što luk  $\widehat{BD}$  preslikamo refleksijom u odnosu na pravu  $BD$ . Jasno je da figura  $\Phi_1$  ima isti obim kao i figura  $\Phi$ , veću površinu od figure  $\Phi$ , ali nije konveksna.



**Slika 21.** Figura  $\Phi_1$

Dalje, izvršimo „preslaganje” duži  $\overline{AB}$  i lukova  $\widehat{C'B}$  i  $\widehat{DC'}$ , tako da je  $\overline{AB} = \overline{XY}$ ,  $\widehat{XA} = \widehat{C'B}$  i  $\widehat{DC'} = \widehat{DY}$ . Time dobijamo figuru  $\Phi_2$  koja je prikazana na slici 22. Figura  $\Phi_2$  ima isti obim kao i figura  $\Phi_1$ , veću površinu od figure  $\Phi_1$  i konveksna je.

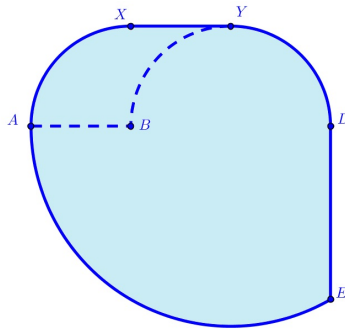
Konačno, figura  $\Phi_2$  ima isti obim kao i figura  $\Phi$  i veću površinu od figure  $\Phi$ . Dakle, možemo uzeti  $\Omega = \Phi_2$ .

ii) Pretpostavimo sada da je figura  $\Phi$  konveksna. Neka je  $\gamma$  granica figure  $\Phi$ . Uočimo na krivoj  $\gamma$  tačke  $A$  i  $B$  takve da dele krivu  $\gamma$  na dva dela jednakih dužina. Obeležimo te delove sa  $\gamma_1$  i  $\gamma_2$  (videti sliku 23). Takodje, obeležimo sa  $\Phi_1$  figuru ograničenu sa  $\gamma_1$  i duži  $\overline{AB}$ , odnosno sa  $\Phi_2$  figuru ograničenu sa  $\gamma_2$  i duži  $\overline{AB}$ .

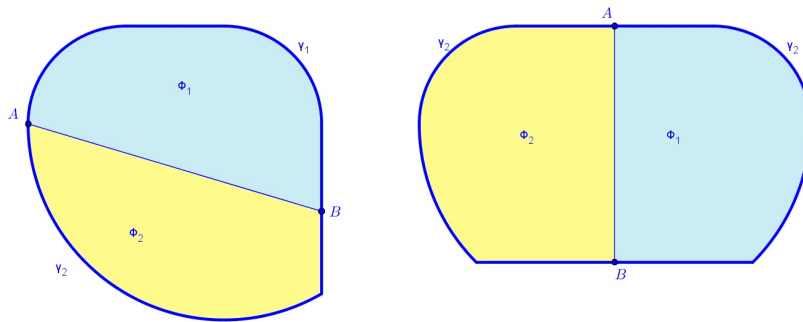
Nadalje bez umanjnja opštosti možemo pretpostaviti da važi  $P(\Phi_1) \geq P(\Phi_2)$ .

Ako je  $P(\Phi_1) > P(\Phi_2)$  (kao što je to slučaj na slici 23 levo) konstruišimo figuru  $\Omega$  na sledeći način:  $\Omega$  je unija figure  $\Phi_1$  i slike figure  $\Phi_1$  pri refleksiji u odnosu na pravu  $AB$  (videti sliku 24). Jasno je da važi  $P(\Omega) > P(\Phi)$  i da figure  $\Phi$  i  $\Omega$  imaju jednak obim.

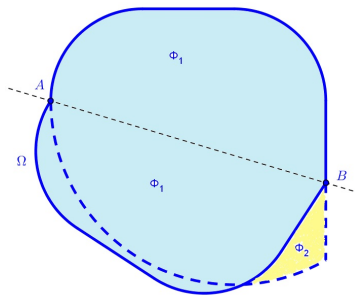




Slika 22. Figura  $\Phi_2$



Slika 23. Primer konveksnih figura



Slika 24. Slučaj  $P(\Phi_1) > P(\Phi_2)$

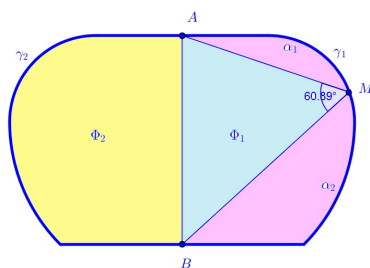
Ako je  $P(\Phi_1) = P(\Phi_2)$  (kao što je to slučaj na slici 23 desno) onda bar jedna od figura  $\Phi_1$  i  $\Phi_2$  nije polukrug (u suprotnom, kako je  $\Phi = \Phi_1 \cup \Phi_2$ , figura  $\Phi$  bi bila krug). Bez umanjenja opštosti možemo pretpostaviti da  $\Phi_1$  nije polukrug. Kako figura  $\Phi_1$  nije polukrug, sledi da na krivoj  $\gamma_1$  postoji tačka  $M$ , različita od tačaka  $A$  i  $B$ , takva da ugao  $\angle AMB$  nije prav (videti sliku 25). Neka je  $\alpha_1$  deo krive  $\gamma_1$  od tačke  $A$  do tačke  $M$  a  $\alpha_2$  deo krive  $\gamma_1$  od tačke  $M$  do tačke  $B$ .

Rotirajmo sada figuru ograničenu sa duži  $\overline{MB}$  i krivom  $\alpha_2$  za ugao  $90^\circ - \angle AMB$  oko tačke  $M$  i obeležimo sa  $\alpha'_2$  i  $B'$  sliku krive  $\alpha_2$  odnosno tačke  $B$  pri toj transformaciji. Dalje, neka je  $\Omega_1$  figura ograničena sa krivama  $\alpha_1$  i  $\alpha'_2$  i duži  $\overline{AB'}$  (videti sliku 26).

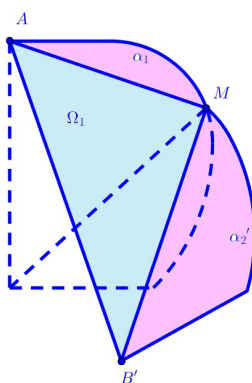
Kako je površina trougla  $\triangle AMB'$  strogo veća od površine trougla  $\triangle AMB$  sledi da je  $P(\Omega_1) > P(\Phi_1)$  a takodje važi i  $l(\alpha_1) + l(\alpha'_2) = l(\gamma_1)$ .

Na kraju konstruišimo figuru  $\Omega$  na sledeći način:  $\Omega$  je unija figure  $\Omega_1$  i slike figure  $\Omega_1$  pri simetriji u odnosu na pravu  $\overline{AB'}$ . Jasno je da važi  $P(\Omega) > P(\Phi)$  i da figure  $\Omega$  i  $\Phi$  imaju jednak obim (videti sliku 27).

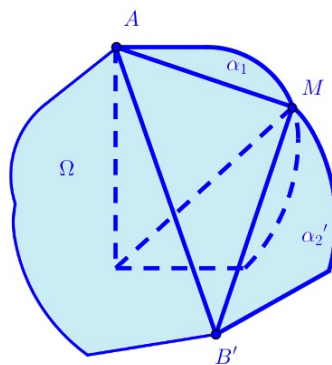
Ovo rešenje izoperimetrijskog problema je dao švajcarski matematičar Jacob Štajner (1796-1863).



Slika 25. Slučaj  $P(\Phi_1) = P(\Phi_2)$



Slika 26. Transformacija figure  $\Phi_1$



Slika 27. Figura  $\Omega$

Medjutim, ovo rešenje nije sasvim korektno. Zaista, pažljivom analizom možemo doći do zaključka da smo jedino dokazali da u familiji  $\mathcal{F}$  za svaku figuru, koja nije krug postoji figura strogo veće površine. Dakle, nismo dokazali i da od svih figura koje pripadaju familiji  $\mathcal{F}$  krug ima najveću površinu. Otuda, ovo rešenje je korektno jedino ako dokažemo da u familiji  $\mathcal{F}$  postoji figura najveće površine.

## 6. Zadaci odredjivanja minimuma ili maksimuma

Zadatak 6.1. U skupu pravouglanih trouglova kojima je zbir dužina kateta jednak  $d$ , odrediti onaj:

- sa najvećom površinom.
- sa najkraćom hipotenuzom.

Rešenje. a) Odredimo dužine kateta  $a$  i  $b$  za koje će površina trougla  $P$  biti najveća, pri čemu je  $a + b = d$  zadato.

*Prvi način.* Kako je pri jednakom obimu kvadrat veće površine od pravougaonika, to će važiti i za njihove polovine pa zaključujemo da će pravougli trougao imati najveću površinu kada predstavlja polovinu kvadrata, tj. kada su mu katete jednake dužine. Konačno, površina  $P$  je najveća kada je  $a = b = \frac{d}{2}$ .

*Drugi način.* Kako razmatramo samo pravougle trouglove i kako su dužine kateta  $a$  i  $b$  imamo da je

$$P = \frac{ab}{2} = \frac{4ab}{8} = \frac{(a+b)^2 - (a-b)^2}{8}.$$

Dalje, kako je  $a + b = d$  imamo da je  $P = \frac{d^2 - (a-b)^2}{8}$  i zaključujemo da će površina biti veća ako je kvadrat razlike  $(a-b)^2$  manji. Konačno, površina  $P$  je najveća kada je  $a = b = \frac{d}{2}$ .

*Treći način.* Zadatak možemo rešiti ispitivanjem osobina pogodno izabrane kvadratne funkcije (slika 28).

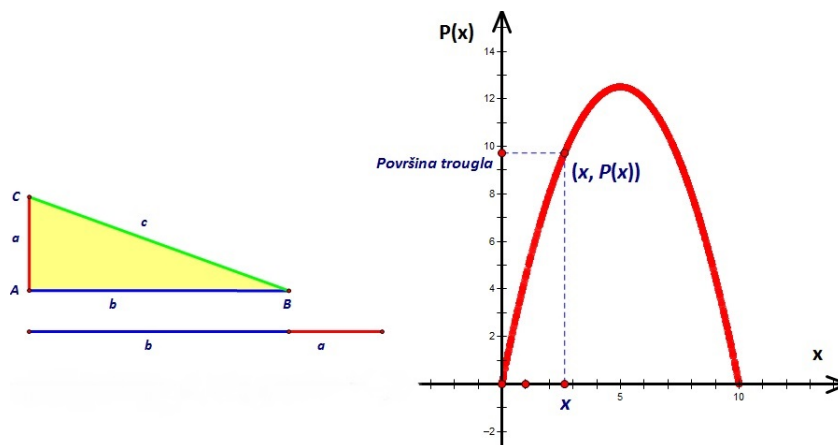
Kako je  $a + b = d$ , a  $d$  je konstanta, veličinu  $b$  možemo izraziti preko  $a$  ( $b = d - a$ ), i izračunati površinu trougla u zavisnosti od  $a$ . Naime,

$$P = \frac{ab}{2} = \frac{a(d-a)}{2} = -\frac{a^2}{2} + \frac{ad}{2}.$$

Pogodno je smatrati da je  $x = a$  promenljiva,  $0 \leq x \leq d$ . Tada površinu  $P$  možemo posmatrati kao funkciju promenljive  $x$ . Zaista,  $P = P(x) = -\frac{1}{2}x^2 + \frac{d}{2}x$ . Iz  $(x - \frac{d}{2})^2 = x^2 - xd + \frac{d^2}{4}$  sledi

$$P(x) = -\frac{1}{2} \left( \left( x - \frac{d}{2} \right)^2 - \frac{d^2}{4} \right).$$

Otuda dobijamo da  $P$  kao funkcija promenljive  $x$  ima maksimum  $\frac{d^2}{8}$  kada je  $x = \frac{d}{2}$ . Konačno, površina  $P$  je najveća kada je  $a = b = \frac{d}{2}$ .



**Slika 28.** Površina trougla kao kvadratna funkcija

b) Odredimo dužine kateta  $a$  i  $b$  za koje će dužina hipotenuze  $c$  biti najmanja, pri čemu je  $a + b = d$  zadato. Iz Pitagorine teoreme sledi da je  $c^2 = a^2 + b^2$ . Takođe, kako je dužina uvek nenegativna, sledi da je veličina  $c$  najmanja ako i samo ako je veličina  $c^2$  najmanja. Dakle, odredimo dužine kateta  $a$  i  $b$  za koje će kvadrat dužine hipotenuze biti najmanji.

*Prvi način.* Ako sa  $P$  obeležimo površinu trougla, imamo da je

$$c^2 = a^2 + b^2,$$

odnosno,

$$c^2 = (a + b)^2 - 2ab,$$

odnosno kako je  $P = \frac{ab}{2}$ ,

$$c^2 = (a + b)^2 - 4P,$$

odnosno kako je  $a + b = d$ ,

$$c^2 = d^2 - 4P.$$

Iz poslednje jednakosti neposredno sledi da će kvadrat dužine hipotenuze biti manji ako je površina veća. Kako smo u delu pod a) dokazali da je površina najveća kada je  $a = b = \frac{d}{2}$  sledi da je kvadrat dužine hipotenuze a samim tim i dužina hipotenuze najmanja kada je  $a = b = \frac{d}{2}$ .

*Drugi način.* I ovaj zadatak možemo rešiti ispitivanjem osobina pogodno izabrane kvadratne funkcije (slika 29). Kako je  $a + b = d$ , a  $d$  je konstanta, veličinu  $b$  možemo izraziti preko  $a$  ( $b = d - a$ ), i izračunati kvadrat dužine hipotenuze u zavisnosti od  $a$ . Naime,

$$c^2 = a^2 + b^2 = a^2 + (d - a)^2 = 2a^2 - 2ad + d^2.$$

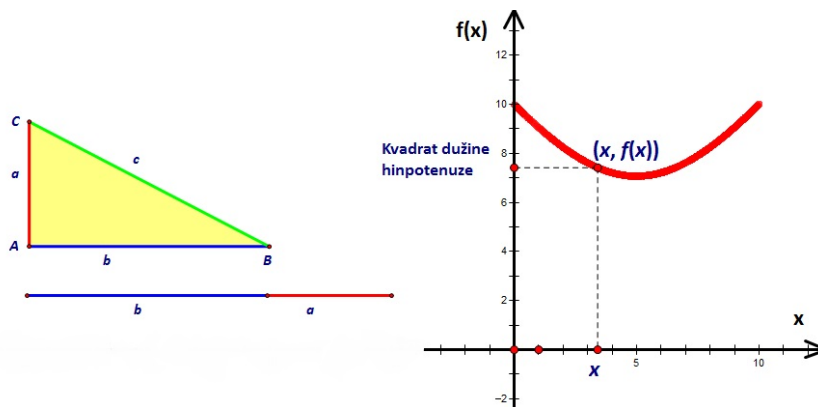
Pogodno je smatrati da je  $x = a$  promenljiva,  $0 \leq x \leq d$ . Tada kvadrat dužine hipotenuze  $c^2$  možemo posmatrati kao funkciju promenljive  $x$ . Zaista,  $c^2 = f(x) = 2x^2 - 2dx + d^2$ . Iz

$$\left(x - \frac{d}{2}\right)^2 = x^2 - xd + \frac{d^2}{4}$$

sledi

$$f(x) = 2 \left( \left(x - \frac{d}{2}\right)^2 + \frac{d^2}{4} \right).$$

Otuda dobijamo da  $f$  kao funkcija promenljive  $x$  ima minimum  $\frac{d^2}{2}$  kada je  $x = \frac{d}{2}$ . Konačno, kvadrat dužine hipotenuze  $c^2$  a samim tim i dužina hipotenuze  $c$  je najmanja kada je  $a = b = \frac{d}{2}$ .



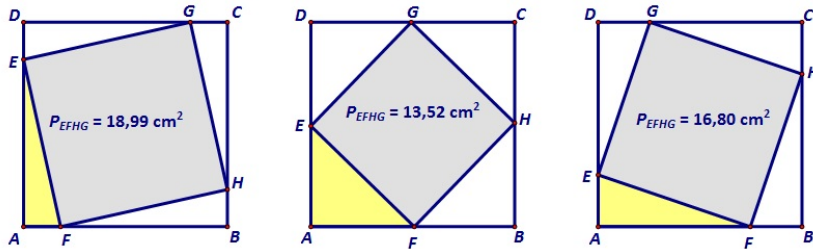
**Slika 29.** Kvadrat dužine hipotenuze kao kvadratna funkcija

Napomenimo da je metod koji smo koristili u trećem načinu rešavanja dela a) i u drugom načinu rešavanja dela b) na neki način „opšti metod” kako rešavati ovakve zadatke. Naime, cilj je predstaviti jednu veličinu kao funkciju druge veličine a onda ispitivanjem svojstava te funkcije doći do rešenja. U ovom zadatku ta funkcija je kvadratna funkcija, pa se maksimum i minimum takve funkcije može elementarno odrediti. U slučaju da funkcija nije kvadratna a ima prvi i drugi izvod za određivanje maksimuma i minimuma možemo koristiti poznate teoreme diferencijalnog računa.

△

**Zadatak 6.2.** U zadati kvadrat  $ABCD$  upisati kvadrat  $EFHG$  najmanje površine.

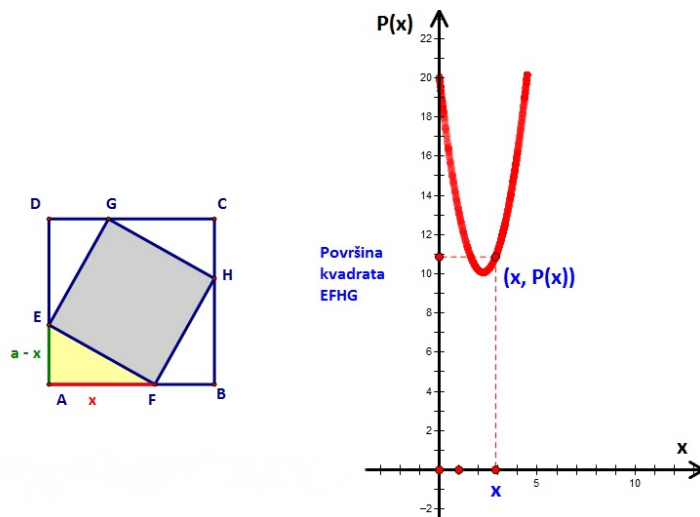
**Rešenje.** Koristeći se prethodnim saznanjima, uočavamo sledeće: površina kvadrata  $EFHG$  biće najmanja kada je površina preostalog dela kvadrata  $ABCD$  najveća. Budući su ostaci pravougli trouglovi, oni će biti najveće površine kada su jednakokraki (slika 30). Zaključujemo, kvadrat  $EFHG$  ima najmanju površinu kada su njegova temena jednako udaljena od temena polaznog četvorougla  $ABCD$ , tj. kada su u središtima stranica četvorougla  $ABCD$ .



**Slika 30.** Upisani kvadrat

Kao i u prethodnom zadatku, rešenje bismo mogli dati i pomoću kvadratne funkcije.

Ako rastojanje temena  $F$  kvadrata  $EFHG$  do temena  $A$  polaznog kvadrata  $ABCD$  obeležimo sa  $x$ , tada je rastojanje tog istog temena do temena  $B$  jednako  $a - x$ , pri čemu je  $a$  dužina stranice kvadrata  $ABCD$ . Dužinu stranice kvadrata  $EFHG$  označimo sa  $b$  (slika 31).



**Slika 31.** Površina upisanog kvadrata kao kvadratna funkcija

Primetimo da površinu kvadrata  $EFHG$  možemo videti kao funkciju promenljive  $x$ ,  $0 \leq x \leq a$ . Zaista,

$$P = b^2$$

$$P = (a - x)^2 + x^2$$

$$P = 2x^2 - 2ax + a^2.$$

Vidimo da je površina  $P$  kvadratna funkcija promenljive  $x$ , tj. imamo da je  $P = P(x) = 2x^2 - 2ax + a^2$ . Kako je  $(x - \frac{a}{2})^2 = x^2 - ax + \frac{a^2}{4}$ , sledi da je

$$P(x) = 2 \left( \left( x - \frac{a}{2} \right)^2 + \frac{a^2}{4} \right).$$

Funkcija  $P$  ima minimum  $\frac{a^2}{2}$  kada je  $x = \frac{a}{2}$ , pa je i površina kvadrata  $EFHG$  najmanja kada su mu temena u središtima stranica polaznog kvadrata  $ABCD$ .

△

### 6.1. Zadaci za samostalan rad

**Zadatak 6.3.** U jednakokraki pravougli trougao sa katetom dužine  $2cm$  upisan je pravougaonik tako da mu je jedno teme u temenu pravog ugla trougla. Koji od ovih pravougaonika ima najveću površinu? (Rešenje: Kada je pravougaonik kvadrat, nezavisno od dužine katete)

**Zadatak 6.4.** U jednakokraničan trougao stranice dužine  $a$  upisan je pravougaonik čija je jedna stranica na stranici trougla, a ostala dva temena pripadaju po jednoj od dveju ostalih stranica. Koji od tih pravougaonika ima najveću površinu? (Rešenje: Kvadrat)

**Zadatak 6.5.** Žicu dužine  $d$  treba podeliti na dva dela. Od jednog dela treba oblikovati kvadrat stranice dužine  $a$ , a od drugog dela treba oblikovati pravougaonik dužine  $b$  i širine  $c$ , tako da je dužina tri puta veća od širine. Gde treba preseći žicu tako da zbir površina kvadrata i pravougaonika bude najmanji? (Rešenje:  $a = 6d/56$ ,  $c = 4d/56$ )

**Zadatak 6.6.** Zadat je trapez obima  $d$ , pri čemu su osnovice dužina  $a$  i  $c$ , a uglovi uz veću osnovicu trapeza  $60^\circ$ . Odrediti dužine osnovica trapeza za koje će površina trapeza biti maksimalna. (Rešenje:  $a = 3d/8$ ,  $c = d/8$ )

**Napomena .** Pogodno je zadatke rešavati na različitim nivoima: za učenike osnovne škole i za učenike srednje škole.

### 7. Komentari

Interesantno je detaljnije razmotriti formulu (1). U tom cilju, neka je zadat trougao  $\triangle ABC$ . Dokaz da za površinu trougla  $\triangle ABC$  važi formula (1) obično počinjemo frazom:

A) „Dopunimo trougao  $\triangle ABC$  do paralelograma”.

Preciznije, umesto A) možemo reći

B) „Postoji tačka  $D$  tako da je četvorougao  $ABDC$  paralelogram”.

Otuda, dokaz da važi formula (1) je relativno jednostavan ako smatramo da su neka svojstva euklidske geometrije kao na primer B) „očigledna”.

Ako počnemo da razmišljamo o svojstvu B) sledeća pitanja su prirodna.

- 1) Da li svojstvo B) ima veze sa aksiomom paralelnosti?
- 2) Koje od aksioma euklidske geometrije koristimo da dokažemo svojstvo B)?
- 3) Da li postoji paralelogram u ne-euklidskim geometrijama?
- 4) Zašto svojstvo B) nije „očigledno” (ako je uopšte tačno) u ne-euklidskim geometrijama?
- 5) Može li se pojam površine uvesti u apsolutnoj geometriji?

Planiramo da se ovim pitanjima bavimo u daljem istraživanju. Ovde samo navodimo neke činjenice i ideje.

Oko 300. p.n.e. antički matematičar Euklid je u svom delu Elementi geometriju zasnovao na sledećih pet postulata (polaznih tvrdjenja):

- p1) Za svaku tačku  $P$  i za svaku tačku  $Q$ , različitu od tačke  $P$ , postoji jedinstvena prava  $l$  koja sadrži tačke  $P$  i  $Q$ .
- p2) Za svaku duž  $\overline{AB}$  i svaku duž  $\overline{CD}$  postoji jedinstvena tačka  $E$  takva da je tačka  $B$  između tačaka  $A$  i  $E$  i duž  $\overline{CD}$  je podudarna duži  $\overline{BE}$ .

p3) Za svaku tačku  $O$  i svaku tačku  $A$ , različitu od tačke  $O$ , postoji krug sa centrom  $O$  i poluprečnikom  $\overline{OA}$ .

p4) Svi pravi uglovi su međusobno podudarni.

p5) Za svaku pravu  $p$  i svaku tačku  $A$  koja ne pripada pravoj  $p$ , u ravni određenoj pravom  $p$  i tačkom  $A$ , postoji tačno jedna prava  $q$  takva da sadrži tačku  $A$  i disjunktna je sa pravom  $p$ .

Tokom istorije se pokazalo da je Euklid imao nekih propusta u zasnivanju geometrije (npr. u Elementima nema strogo zasnivanja pojma neprekidnosti). To naravno ne umanjuje značaj njegovog dela. Štaviše sve do 19. veka Elementi su bili glavni udžbenik iz geometrije a do danas su verovatno najznačajnije delo u istoriji matematike.

Mnogi matematičari su vekovima radili na preciziranju Euklidovih ideja. Tako danas, kada u nastavi matematike razmatramo geometriju najčešće podrazumevamo sledećih pet grupa aksioma (polaznih tvrdjenja):

e1) Aksiome incidencije;

e2) Aksiome rasporeda;

e3) Aksiome podudarnosti;

e4) Aksioma neprekidnosti;

e5) Aksioma paralelnosti.

Na ovom mestu izdvojimo samo aksiomu e5):

*Za svaku pravu  $p$  i svaku tačku  $A$  koja ne pripada pravoj  $p$  u ravni određenoj pravom  $p$  i tačkom  $A$  postoji tačno jedna prava  $q$  takva da sadrži tačku  $A$  i disjunktna je sa pravom  $p$ .*

Korišćenjem aksioma e1)-e4) može se dokazati egzistencija prave  $q$  o kojoj govori aksioma e5), ali ne i jedinstvenost. Otuda se tokom istorije matematike pokazalo da ima smisla razmatrati i geometriju zasnovanu na aksiomama e1)-e4) i aksiomi h5) koja glasi:

*Za svaku pravu  $p$  i svaku tačku  $A$  koja ne pripada pravoj  $p$  u ravni određenoj pravom  $p$  i tačkom  $A$  postoje bar dve prave  $q_1$  i  $q_2$  takve da sadrže tačku  $A$  i disjunktna su sa pravom  $p$ .*

Geometriju zasnovanu na aksiomama e1)-e4) zovemo *apsolutna geometrija*, geometriju zasnovanu na aksiomama e1)-e5) zovemo *euklidska geometrija*, dok geometriju zasnovanu na aksiomama e1)-e4) i aksiomi h5) zovemo *hiperbolička geometrija*.

U euklidskoj geometriji paralelogram obično definišemo kao *četvorougao kome su parovi naspramnih stranica međusobno paralelne duži*. U pomenutoj definiciji implicitno koristimo aksiomu e5). Međutim, ispostavlja se da paralelogram možemo definisati i kao *četvorougao kome se dijagonale polove*. Prihvatimo li ovu drugu definiciju ispostavi se da svojstvo B) zavisi samo od aksioma e1) do e4).

Preciznije, neka je u apsolutnoj ravni dat trougao  $\triangle ABC$  i neka je  $S$  središte duži  $\overline{BC}$ . Na polupravoj  $AS$  ( $A$  je početna tačka) postoji jedinstvena tačka  $D$ , različita od tačke  $A$  takva da je  $\overline{AS} \cong \overline{SD}$ .

Dakle svojstvo B) možemo prihvatiti kao svojstvo apsolutne geometrije. Ali sada se postavlja novo pitanje: Kako odrediti površinu četvorougla dobijenog takvom konstrukcijom?

## 8. Zaključak

U radu je obradjen jedan deo sadržaja vezanih za izoperimetrijski problem koji bi se mogao proučavati u nastavi matematike osnovne i srednje škole. Cilj je bio pokazati kako se ti sadržaji mogu obradjevati u svrhu povezivanja različitih matematičkih sadržaja, u svrhu uvodjenja učenika u samostalni istraživački rad i razvijanja sposobnosti „matematičkog odlučivanja”. Pokazano je kako se u cilju eksperimentalnog zaključivanja i istraživanja, kao i kvalitetnijeg vizuelnog prikazivanja pojmova i „procesa dokaza”, prilikom obradjevanja ovih sadržaja, mogu koristiti programi dinamičke geometrije. Naglasak je s jedne strane stavljen na vizualizaciju u

svrhu boljeg razumevanja pojmova što za posledicu ima i trajnije znanje, a s druge strane naglašava se istraživanje i zaključivanje kroz proces eksperimentisanja što za posledicu može imati veću aktivnost i samostalni rad učenika.

### Literatura

- [1] **C. Bandle.** Isoperimetric inequalities and applications, London, 1980.
- [2] **A. Bogomolny.** Isoperimetric Theorem and Inequality from Interactive Mathematics Miscellany and Puzzles  
<http://www.cut-the-knot.org/do-you-know/isoperimetric.shtml>, Accessed 02 December 2012.
- [3] **N. Jozić.** Učenje usmjerenim opažanjem, *Zbornik povzetkov, 1. mednarodna Konferenca o učenju in poučevanju matematike*, Maribor, 23. in 24. avgust 2012.
- [4] **C. Donald.** Hyperbolic Geometry in the High School Geometry Classroom Iowa State University  
<http://www.math.iastate.edu/thesisarchive/MSM/Donald-C-MSM-F05.pdf>, Accessed 06 March 2013.
- [5] **D. A. Kryjanovsky.** Izoperimetry, *Fizmatgiz*, Moskva, 1959.
- [6] **M. Mateljević.** The isoperimetric inequality and some extremal problems in  $H^1$ , *Lect. Notes Math.* 798, 1980, 364-369.
- [7] **M. Mateljević.** The isoperimetric inequality in the Hardy class II, *Mat. Vesnik* 3 (16) (31), 1979, 169-178.
- [8] **M. Mateljević.** Area-modulus inequality, rukopis.
- [9] **G. Szekeres.** The Isoperimetric Problem, *Parabola* Vol. 37, Issue 2 (2001).
- [10] **A. Treibergs.** Inequalities that Imply the Isoperimetric Inequality  
<http://www.math.utah.edu/treiberg/isoperim/isop.pdf>
- [11] **E. W. Weisstein.** Isoperimetric Problem, From MathWorld-A Wolfram Web Resource  
<http://mathworld.wolfram.com/IsoperimetricProblem.html>

Received



# УПОЗНАВАЊЕ УЧЕНИКА СА ОСОБИНАМА ФУНКЦИЈА ПОМОЋУ ГЕОГЕБРЕ И ПРИМЕРА ИЗ РЕАЛНОГ КОНТЕКСТА

мр Наталија Будински

*Основна и средња школа са домом ученика „Петро Кузмјак“, Русинска 63, Руски Крстур  
[nbudinski@yahoo.com](mailto:nbudinski@yahoo.com)*

**Резиме.** Овај рад даје методички приказ како се особине експоненцијалне, логаритамске и логистичке функције могу обрадити помоћу Геогebre и ситуација из реалног контекста. Осим теоријског прегледа, дат је и конкретан пример који се односи на представљање популације Београда у математичком контексту.

**Кључне речи:** експоненцијална, логаритамска, логистичка функција, Геогebra, популација

## 1. Увод

Развојем технологије и њеном употребом у наставном процесу, функционалан приступ учењу алгебре постаје све доминантнији, а то подразумева да су функције централни концепт око ког се организује настава. Овај приступ повезује елементе традиционалне наставе алгебре, као што је сређивање израза, решавање једначина и проучавање особина функција са савременим начином учења математике у контексту. Такође, то значи да разне „зависности“ треба приказивати погодним функцијама и на тај начин обрађивати њихове особине [1]. Орјентисаност наставе према решавању реалних проблема уз помоћ рачунара омогућава да се садржаји из алгебре не свде само на технику решавања, на пример једначина, већ да се уче са дубљим разумевањем и смислом.

Технологија у функционалном приступу омогућава експериментисање и визуелизацију великог броја случајева истовремено, а то доприноси бољем усвајању појмова. На пример, ученици у току другог разреда средње школе теоријски изучавају својства експоненцијалне функције. Да би се приказало понашање експоненцијалне функције  $y = e^{ax}$  или логаритамске функције  $y = \log_a x$  у зависности од параметра  $a$ , математички софтвер омогућава да се експериментише са великим бројем случајева. На основу тога ученици уз помоћ наставника изводе одређене закључке, а представљање математичким симболима се проширује на динамичко представљање.

Главна идеја функционалног учења алгебре је разумевање на који начин промена независне променљиве утиче на промену зависне. Најпогоднији начин за такво учење је моделирање реалних ситуација, јер тако ученици пролазе кроз искуство у ком описују зависност математичких појмова и реалних ситуација и постепено усвајају особине функција [3].

## 2. Функције у средњој школи

Функције се у наставном процесу уводе постепено. Будински [2] предлаже теорију АПОШ (APOS) која описује кораке кроз које би наставници требали провести ученике, а у циљу бољег савладавања предвиђеног програма везаног за функције. Ова теорија је заснована на претпоставки да се математичко знање састоји од тежње појединца да се бави

математичким проблемом користећи смислене менталне *акције* (action), *процесе* (process), *објекте* (objects) и *шеме* (schemas), да би поједноставио ситуацију и решио проблем.

У примеру експоненцијалних и логаритамских функција то подразумева следеће кораке:

- **Обрада експонента и логаритма као акција.** Ученици прво усвајају појам експонента или изложиоца који у првим корацима учења природан број, а затим га проширују на поље реалних бројева. Што се тиче логаритама, ученици треба прво да усвоје дефиницију и увежбају основна правила логаритмовања.
- **Обрада експоненцијалне и логаритамске функције као процеса** подразумева да након примене активности, тј увежбавања једноставнијих правила и задатка, акција прераста у процес. Ученици који савладају тај ниво, у могућности су да се упознају са експоненцијалном и логаритамском функцијом и да дођу до одређених закључака, као, на пример, о особинама ових функција. Погодно је да се функције уводе преко примера из реалног живота.
- **Обрада особина експоненцијалне и логаритамске функције као објекта** значи да ученици знају да одреде домен одређене функције, као и друге особине које карактеришу функцију, на пример, нуле, знак функције, монотоност, екстремне тачке, као и конвексност, конкавност и превојне тачке функције.
- **Генерализација или шема** подразумева да на крају предавања ученици буду оспособљени да се баве функцијама на реалном домену.

У следећем делу рада се посебно разматра обрада функције као процеса и објекта, где ученици проширују своје знање функција и проучавају особине функције преко примера из реалног живота. Осим на логаритамску, направљен је осврт на линеарну и експоненцијалну функцију. Како је реална ситуација везана за људску популацију, направљен је осврт и на логистичу функцију. Математички модели су направљени у Геогебри, јер је то едукативни софтвер који има веома добре могућности за обраду средњошколских математичких садржаја.

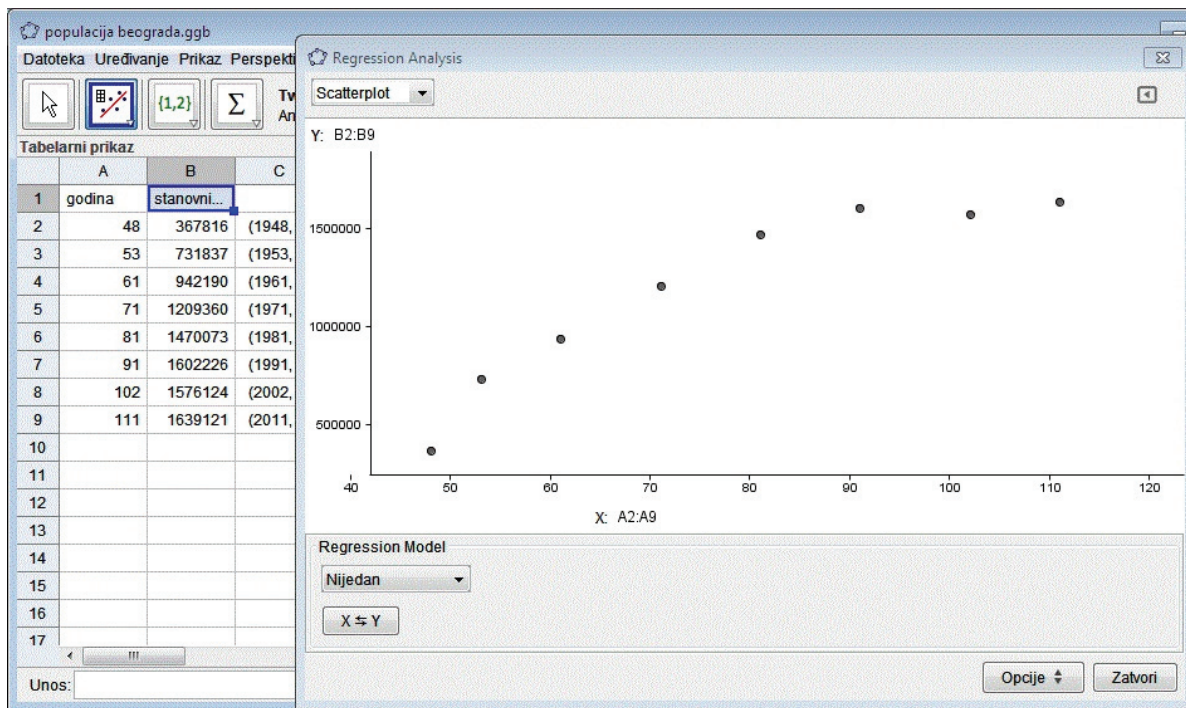
### 3. Модели популације Београда у Геогебри

Математичко моделирање у настави математике се састоји од следећих фаза:

- реална ситуација и проблем,
- математички модел и решење,
- решење реалног проблема, његова провера и извештај

*Реална ситуација* којом се уводе ученици у моделирање је попис становништва, док је *реалан проблем* предвиђање популације Београда на основу података из предходних и тренутно актуелног пописа 2011. Подаци су преузети са [www.sr.wikipedia.org](http://www.sr.wikipedia.org) и унешени су у Геогебри као тачке, где апсциса представља годину пописа, а ордината број становника. На Слици 1 је приказан табеларни унос података у Геогебри, као и тачкасти дијаграм.

У овој фази моделирања, уз помоћ рачунара, могуће је ученике навести да повежу графички приказ података са графиком неке функције коју су раније обрадили. Тако конкретни подаци за Београд „сличе“ графику логаритамске функције  $y = \log_a x$ , где је  $a > 0$ .



Слика 1 Приказ података пописа у Геогеври

Међутим, ученике треба заинтересовати за конкретан облик функције који описује промену броја становника. У тој фази моделирања треба прећи на посебно важан део процеса који се односи на математички модел и математичко решење које доводи до решења реалног проблема. Фаза математичког модела и решења подразумева добијање функције која описује реалне податке, а која омогућава предвиђање броја становника у наредном периоду. За добијање математичког решења Геогевра има уграђене опције које омогућавају фитовање кривих на основу унесених података. Са опцијама *FitLin*, *FitExp*, *FitLog* на основу добијених података се добијају следеће функције:

$$12374504922367x - 567558471y = 280752735546167 \quad (1),$$

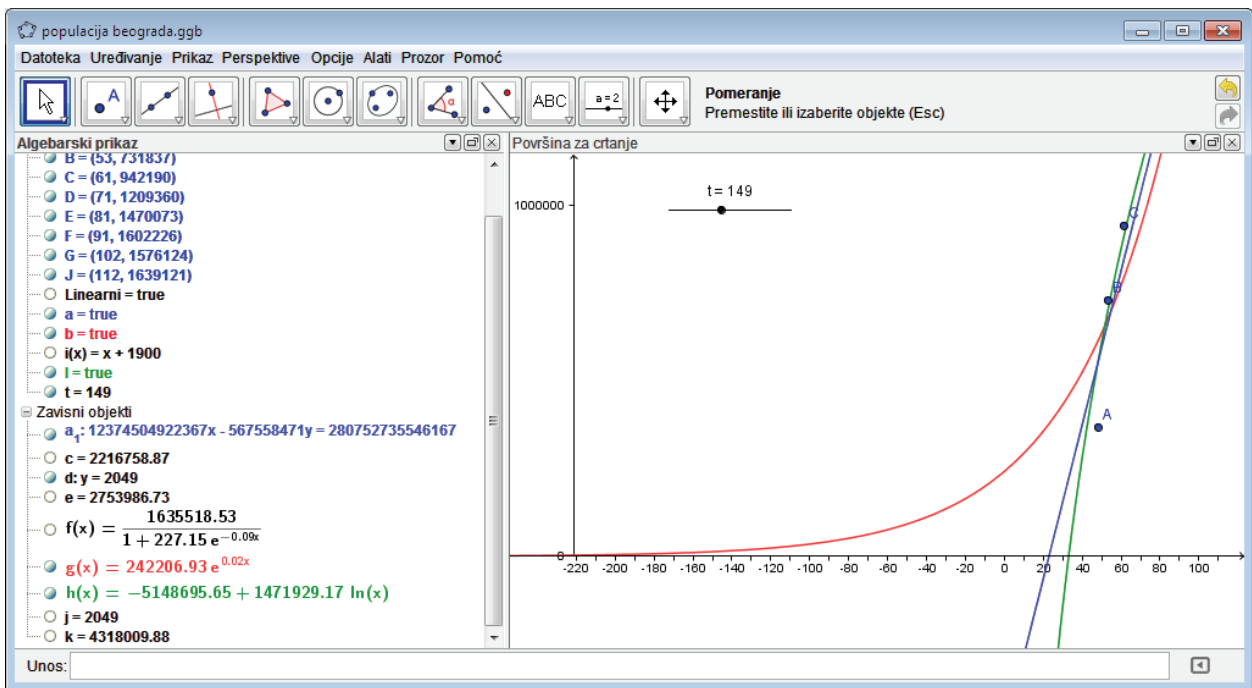
$$y = 242206,93e^{0,02x} \quad (2),$$

$$y = -5148695,65 + 1471929\ln x \quad (3)$$

које представљају математичке моделе становништва Београда. На Слици 2 су графички приказане линеарна (1), експоненцијална (2) и логаритамска (3) функција.

Једноставно одређивање функција уз помоћ Геогевре ученицима и наставнику остаје више времена за анализирање њихових особина током школског часа. Аналитички изрази се могу користити за одређивање нула функција или пресека са у-осом. На тај начин се ствара веза између предавања са рачунаром и предходно класично обрађених лекција, а ученици стичу увид у повезаност математичких појмова и разноврсност њиховог представљања [4].

Приликом анализе функција, може се уочити да су све функције растуће, али да не одговарају све функције подједнако задатим подацима. Због тога би требало ученике упознати и са логистичком функцијом.



Слика 2 Приказ функција које описују податке пописа

Логистичка функција се користи за описивање раста популације, али се не обрађује посебно у средњој школи. Међутим, њене особине, као што је асимптотско понашање, могу да допринесу бољем разумевању појединих математичких појмова. Логистичка функција која описује становништво Београда може се добити у Геогеври помоћу функције *FitLogistički* и гласи:

$$y = \frac{1635518,53}{1 + 227,15e^{-0,09x}} \quad (4)$$

На примеру логистичке функције ученицима другог разреда средње школе се може објаснити појам асимптоте, како у вези са реалном ситуацијом, тако и математички. Прво што им треба предочити да није природно да становништво може да расте до „бесконачности“, већ да се број становника „негде у будућности“ устали због ограничених ресурса. Броју на ком ће се усталити број становника главног града представља асимптоту функције (4). Ово је методичка ситуација када је погодно увести ученицима дефиницију хоризонталне асимптоте.

Након постављања математичких модела и решења, треба наћи решење за реалан проблем. У овом случају то је предвиђање популације Београда у наредном периоду. Опција клизача у Геогеври, означена у овом примеру као број година  $t$ , омогућава да се за одреди број становника за две, три или сто година. Геогевра има могућности уношења динамичког текста, тако да се број становника мења и исписује у зависности од задатог броја година.

Веома важна наставна ситуација у процесу моделирања је провера решења. У овом примеру одређивање која функција најбоље описује дате податке, односно која најбоље предвиђа број становника.

Геогевра има опцију израчунавања коефицијента детерминације  $R^2$  који је показатељ заједничких обележја која учествују у корелацији. У овом случају то су године и број

становника. У опцијама *Prikaz* и *Tabelarni prikaz*, постоји опција *Two Variable Regression Analysis* која даје могућности преко опције *Regression Model* избор линеарног (1), експоненцијалног (2), логаритамског (3) и логистичког (4) модела. Притиском на дугме *Опције* и избором *Prikaži statistiku* добијамо табеларни приказ различитих параметара модела. У претпоследњој врсти дата је вредност  $R^2$ . На основу задатих података за моделе (1), (2), (3) и (4) редом добијамо вредности 0,868, 0,6654, 0,9318 и 0,984 добијамо да су логаритамски и логистички модел најпогоднији за дате податке. Самим тим они и најбоље предвиђају број становника.

#### 4. Закључак

Искуства показују да учење математичких појмова, посебно функција, моделирањем даје добре резултате [5]. Описане радионице се баве једноставнијим примерима математичког моделирања и математички модели су направљени помоћу Геогebre. Ови примери су прилагођени ученицима других разреда средњих школа за упознавање ученика са функцијама, као и са њиховом применом у реалном контексту. Геогebra омогућава визуелизацију проблема, као и брзу манипулацију са великим бројем података, експериментисање и рачунање, што наставнику отвара простор за предавање математичких садржаја у ширем, не само уском математичком, контексту.

#### Литература

- [1] **C. Kieran.** The changing face of school algebra. In C. Alsina, J. M. Alvares, B. Hodgson, C. Laborde, & A. Pérez (Eds.), *Proceedings of the 8th International Congress on Mathematics Education: selected lectures*, Seville: S.A.E.M. 'Thales', 1996, pp. 271-290.
- [2] **E. Dubinsky and M. A. McDonald.** APOS: A Constructivist Theory of Learning in Undergraduate Mathematics Education Research, *New ICMI Study Series*, 7(3), 2002, pp.275-282.
- [3] **M. K. Heid.** A technology-intensive functional approach to the emergence of algebraic thinking, In N. Berdnarz, C. Kieran & L. Lee (Eds.), *Approaches to Algebra*, Netherlands: Kluwer Academic Publishers, 1996, pp. 239.
- [4] **J-B. Lagrange and J-M. Gelis.** The Casyopée project: a Computer Algebra Systems environment for students' better access to algebra' better access to algebra, *Int. J. Continuing Engineering Education and Life-Long Learning*, 18(5/6), 2008, pp.575–584.
- [5] **N. Budinski and D. Takači.** Learning and teaching mathematics through real life models, *Int. J. Technol. Math. Educ*, 18(1), 2011, pp. 33-37.



# Rješavanje nekih organizacionih problema u nastavi primjenom pronalaženja maksimalno balansirane povezane particije u grafu

**Dragan Matić**

*Prirodno matematički fakultet Banja Luka, Mladena Stojanovića 2  
e-mail: matic.dragan@gmail.com*

**Milan Božić**

*Matematički fakultet Beograd, Studentski trg 16  
e-mail: mbozic@matf.bg.ac.rs*

**Rezime.** Problemi kombinatorne optimizacije pronalaze široku primjenu u rješavanju raznih organizacionih problema. Poznati NP težak problem maksimalno balansirane povezane particije u grafu (Maximally Balanced Connected Partition problem - MBCP) je iskorišten za određivanje podjele nastavnih jedinki jednog kursa na dva dijela, tako da se očuva povezanost jedinki unutar svakog dijela, a da težina gradiva u oba dijela bude ujednačena. U radu je prikazana metodologija određivanja grafa koji odgovara datom kursu, te su opisane neke metode rješavanja ovog problema.

**Ključne riječi:** Particije grafa; računari u obrazovanju; organizacija kursa

## 1. Uvod

U vremenu intenzivnih reformi u obrazovanju javljaju se organizacioni problemi koji iniciraju razvoj složenih i naprednih metoda za njihovo rješavanje. Kako bi se postigla bolja efikasnost, veća ušteda resursa ili smanjenje troškova, pred osobe uključene u upravljanje i organizovanje nastavnog procesa se stavljaju zadaci čije rješavanje podrazumijeva pronalaženje najboljeg ili barem dovoljno dobrog rješenja. Ovakvi problemi se u praksi često sreću i predmet su izučavanja ne samo matematičara, već i istraživača iz drugih naučnih oblasti, kao što su ekonomija, menadžment i inženjerstvo.

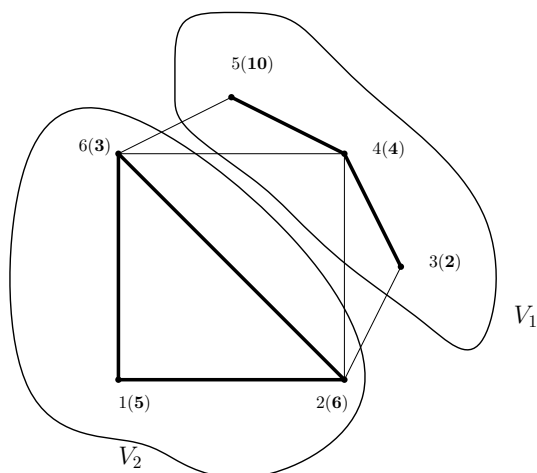
U većem broju slučajeva, pokazuje se da organizacioni problemi u nastavnom procesu u svom izvornom obliku zapravo jesu problemi matematičke prirode i kao takvi se javljaju i u drugim organizacionim problemima. Stoga se provjerene i dokazane metode rješavanja sličnih problema, kao i iskustva stečena u drugim oblastima nauke i privrede, (kao što su upravljanje proizvodnjom, organizovanje transporta, upravljanje računarskim mrežama), mogu primijeniti i u organizovanju nastavnog procesa. Time se otvara široka paleta alata i sistema za rješavanje, što generalno može da doprinese pronalaženju dobrih rješenja i kod problema organizovanja nastave.

Kao što se može očekivati, veliki broj ovakvih problema su veoma složeni i iako u današnje vrijeme upotreba računara u značajnoj mjeri olakšava njihovo rješavanje, pokazuje se da se za ove probleme moraju razvijati specifične metode rješavanja, koje su zasnovane na razvoju novih matematičkih modela i tzv. metaheurističkom pristupu, koji podrazumijeva usmeravanje procesa pronalaska rješenja ka onim oblastima prostora pretraživanja za koje se pretpostavlja da sadrže bolja rješenja.

Da bi se napravio matematički model za dati problem, svaki element problema mora biti predstavljen nekom matematičkom strukturom, dok odnosi između elemenata indukuju odgovarajuće relacije između tih matematičkih struktura. Za rješavanje ovih problema, istraživači teže upotrebi efikasnih i funkcionalnih matematičkih struktura, za koje već postoje teorijski rezultati i za koje su već razvijeni alati za rješavanje. Za rješavanje mnogih problema koji su kombinatorne prirode, koriste se matematičke strukture iz oblasti kombinatorike i teorije grafova, te se za rješavanje odgovarajućih grafovskih problema koriste postojeće ili razvijaju nove tehnike iz te oblasti.

U slučaju kada se radi o grafovima velikih dimenzija, često korišten pristup u rješavanju problema je particionisanje (podjela) grafa na dva manja dijela, koji u nekoj mjeri nasljeđuju osobine polaznog grafa. Dalje se problem rekursivno rješava na manjim grafovima, za koje algoritmi pronalaze rješenja u kraćem vremenu i uz korištenje manje resursa uopšte. Ako pretpostavimo da je polazni graf povezan, prirodno je zahtijevati da i podgrafovi takođe budu povezani. U slučajevima kada je riječ o težinskom, povezanom grafu, prirodan zahtjev je da podgrafovi takođe budu povezani, a da sume težina u podgrafovima budu što bliže jedna drugoj.

Problem opisan u prethodnom pasusu se zove problem pronalaženja maksimalno balansirane povezane particije (eng. Maximally Balanced Connected Partition Problem - MBCP). Ovaj problem je NP težak [1], tako da ne



Slika 1. Graf i rješenje MBCP problema

postoji algoritam koji pronalazi rješenje u polinomijalnom vremenu. Ova činjenica motiviše naučnike da razvijaju razne heurističke metode za rješavanje ovog problema.

### 1.1. Matematička definicija problema

Neka je  $G = (V, E)$ ,  $V = \{1, 2, \dots, n\}$  povezan graf i  $|E| = m$ . Neka su  $w_i$  težine na čvorovima. Za proizvoljan  $V' \subset V$  uvedimo oznaku  $w(V')$  za sumu težina svih čvorova iz  $V'$ , tj.  $w(V') = \sum_{i \in V'} w_i$ . Problem maksimalno balansirane povezane particije u grafu (MBCP) je određivanje particije skupa čvorova  $V$  u dva disjunktne, neprazna skupa,  $V_1$  i  $V_2$ , takva da su podgrafovi grafa  $G$  indukovani sa  $V_1$  i  $V_2$  povezani, a sume težina čvorova iz  $V_1$  i  $V_2$  su po vrijednosti što bliže jedna drugoj, tj. razlika između suma težina je najmanja moguća. Formalno, MBCP je pronalaženje particije  $(V_1, V_2)$ , takve da su podgrafovi od  $G$  indukovani sa  $V_1$  i  $V_2$  povezani, a vrijednost  $obj(V_1, V_2) = |w(V_1) - w(V_2)|$  je minimizovana.

**Primjer 1.** Posmatrajmo graf na slici 1. Nazivi čvorova su 1, 2, ..., 6, dok su težine date u zagradama, odmah uz naziv čvora. Ukupna suma težina je 30. U idealnom slučaju, particija bi trebala da sadrži dvije komponente sa jednakim zbirovima težina (po 15). U tom slučaju, čvor 5, sa težinom 10 bi trebao biti u istoj komponenti sa čvorom 1, ili sa čvorovima 3 i 6, ali u oba ova slučaja komponente  $\{1, 5\}$  i  $\{3, 5, 6\}$  nisu povezane. Stoga, optimalno rješenje problema ne može biti nula, već najmanje 2, jer je ukupna suma svih težina parna. Jedno optimalno rješenje je  $(V_1, V_2) = (\{3, 4, 5\}, \{1, 2, 6\})$ , a  $obj(V_1, V_2) = |16 - 14| = 2$ . Drugo optimalno rješenje  $(V'_1, V'_2) = (\{1, 2, 3, 6\}, \{4, 5\})$ , takođe sa vrijednošću funkcije cilja  $obj(V'_1, V'_2) = |16 - 14| = 2$ .

### 1.2. Prethodni rezultati

Prvi značajniji teorijski rezultat u analizi ovog problema je prikazan u [1]. Kao što je već napomenuto, u tom radu je dokazano da je problem NP težak i predložen je jedan jednostavan heuristički "pohlepni" algoritam polinomijalne složenosti za rješavanje datog problema.

Uspješna implementacija genetskog algoritma (GA) za rješavanje ovog problema je data u [2]. Autori u propisanom GA koriste binarnu reprezentaciju hromozoma, koja ukazuje na to kojoj komponenti je dati čvor dodijeljen. U slučaju pojave nepovezanih komponenti (nekorektnih jedinki), primjenjuje se kaznena funkcija. Pouzdanost i efikasnost ove metode je testirana na grafovima koji sadrže do 300 čvorova i 2000 grana.

Uopštenje ovog problema je problem određivanja  $q, q > 1$  balansiranih povezanih particija u grafu (eng. Balanced Partition Problem - BCPq): za dati cio broj  $q$  i povezan težinski graf  $G$ , potrebno je odrediti  $q$  particija  $V_1, \dots, V_q$ , takvih da je svaki podgraf indukovani odgovarajućom particijom povezan, a razlika između najmanje i najveće sume težina je što manja moguća. To zapravo znači da su particije "izbalansirane" po težinama. U [3], dat je algoritam 2-aproksimacije za  $q = 3$  i  $q = 4$ . Takođe, u [3] je prikazan i dokaz NP složenosti za dati problem. Autori takođe razmatraju i varijantu problema kada  $q$  nije fiksirano, pokazujući da problem nema aproksimirajući algoritam sa odnosom manjim od  $6/5$ , osim ako ne važi da je  $P = NP$ .

Sličan problem ovom problemu je tzv.  $(l, u)$  particionisanje, gdje je cilj podijeliti graf  $G$  u povezane komponente, ali tako da ukupna težina u svakoj komponenti bude najmanje  $l$ , a najviše  $u$ . Skorija istraživanja ovog problema su prikazana u [4, 5]. Na primjer, u [4], autori razmatraju tri problema pronalaska  $(l, u)$  - particije



za dati graf: problem minimalne i problem maksimalne particije (pronalazak  $(l, u)$ -particije sa minimalnim (respektivno maksimalnim) brojem komponenti), te problem pronalaska  $p$  - particija (pronalaženje  $(l, u)$ -particije sa fiksiranim brojem od  $p$  komponenti). Iako je  $(l, u)$  particionisanje NP težak problem u opštem slučaju, za neke specijalne grafove, rješiv je u polinomijalnom vremenu. Na primjer, u [4], autori pokazuju da su za "serijsko - paralelne grafove" problemi pronalaska minimalne i maksimalne particije rješivi u  $O(u^4n)$ , dok je problem pronalaska  $p$ - particije rješiv u  $O(p^2u^4n)$  vremenu. Kratka analiza pokazuje da je određivanje postojanja  $(l, u)$  particije specijalan slučaj BCPq problema: Ako je pronađena particija za BCPq problem, tako da je funkcija cilja za datu particiju manja od  $\delta$ , tada  $(l, u)$  particija postoji za izabrane  $l = t\_sum/q - \delta$  i  $u = t\_sum/q + \delta$ , gdje je  $t\_sum$  suma svih težina.

Kao što vidimo, MBCP i BCPq pripadaju široj klasi problema koji se odnose na particionisanje grafa i kao takvi imaju veliku direktnu i indirektnu primjenu u raznim oblastima inženjerstva, menadžmenta, rješavanju nekih socijalnih problema, kao i problema u obrazovanju. Direktna primjena MBCP problema pronalazi se kod upravljanja i rutiranja u velikim bežičnim mrežama. Neka je data mreža od  $N$  klastera. Da bi se pojednostavilo upravljanje mrežom, ideja je da se tako velika mreža podijeli u dvije podmreže, kojima bi se upravljalo nezavisno jednom od druge. Mreža se modelira kao neusmjeren povezan graf,  $G(H, A)$ , gdje je  $H$  skup čvorova (glava klastera)  $H = \{CH_i : i = 1..N\}$ , a  $A$  skup neusmjerenih veza  $(CH_i, CH_j)$ , gdje su  $CH_i$  i  $CH_j$  dvije glave klastera. Cilj je napraviti particiju grafa  $G$  u povezane balansirane particije i ovaj problem upravo odgovara problemu MBCP. U [6], autori su prilagodili pristup prezentovan u [1] i iskoristili ga da podijele mrežu u dvije manje, povezane podmreže.

U obradi slika, particionisanje se može iskoristiti u situacijama kada dolazi do pogoršanja kvaliteta slike usljed konvertovanja iz jednog oblika u drugi. U situacijama kada nema informacija o procesu degradacije, jedini način da se poboljša kvalitet je da se poveća kontrast i smanje oštećenja pogodnim izmjenama na nivoima sive boje. Da bi se napravio graf koji odgovara slici, određuje se skup čvorova koji odgovaraju nivoima sive boje, a težine čvorova se definišu kao broj pojavljivanja odgovarajuće nijanse u slici. Pronalaženje optimalne transformacije sivih nivoa se formuliše kao particionisanje odgovarajućeg grafa, tako da zbrovi težina čvorova po svakoj komponenti budu što ujednačeniji [7].

Pored primjene u inženjerstvu, MBCP i BCPq problemi imaju značajnu primjenu i u rješavanju nekih socijalnih problema. Na primjer, prilikom određivanja političkih područja, cilj je podijeliti čitavu oblast (mapu) na nekoliko regiona sa skoro podjednakim brojem glasača. Ako je  $G$  odgovarajući dualni graf mape  $M$ , svaki čvor  $v$  od  $G$  predstavlja jedan region, a vrijednost  $w(v)$  predstavlja broj glasača u datom regionu  $v$ . Dva čvora u grafu su susjedna, ako su susjedni odgovarajući regioni u  $M$ . Optimizacioni problem se svodi na pronalaženje particije grafa, tako da odgovarajući regioni imaju skoro podjednak broj glasača, a regioni unutar svake podoblasti su povezani. U [8] je za rješavanje ovakvog problema prikazana heuristika zasnovana na tabu pretraživanju. Pomenimo i primjenu BCPq problema za unapređenje označavanja oblasti policijskih patrola [9]. Najprije se za svaki dio nekog grada (ili šireg regiona) identifikuje nivo kriminala. Kao cilj se zadaje podjela grada na podoblasti, tako da policijske patrole djeluju po datim podoblastima, a da nivo kriminala po podoblastima bude ujednačen. Metod opisan u [9] se sastoji od dvije faze: inicijalnog particionisanja i dodatnog poboljšanja. U prvoj fazi se graf podijeli na dva podgrafa, dok se u drugoj fazi u cilju poboljšanja krajnjeg rezultata heuristički vrši razmjena čvorova između particija.

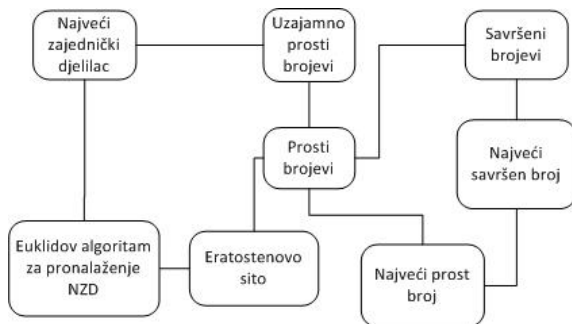
## 2. Primjena MBCP u obrazovanju

U obrazovanju, MBCP se koristi za određivanje rješenja nekih praktičnih organizacionih problema, uključujući upravljanje kursevima i studentskim grupama ili organizovanje istraživača u povezane grupe.

Materijal za neki kurs se obično dijeli u lekcije i svakoj lekciji se može dodijeliti težina. Uspostavljanje veza između lekcija može se ostvariti po nekoliko kriterijuma, kao što su sličnost, analogija, uopštenje ili uslovljenost. Podjela lekcija kursa u dvije cjeline se vrši uz logičan zahtjev da svaka cjelina bude povezana, a da težine te dvije cjeline budu u što većoj mjeri slične.

Kao drugi primjer, možemo navesti podjelu studenata u dvije manje grupe. "Povezanost" studenata se može definisati kao relacija "sposobnost da rade zajedno". Cilj bi mogao biti podijeliti studente u dvije manje grupe, imajući u vidu da grupe trebaju da budu balansirane prema sposobnostima studenata.

MBCP se može koristiti i za organizaciju mreže istraživača na nekom većem projektu. U ovom slučaju, istraživači se predstavljaju čvorovima u grafu, a težina čvorova može biti definisana kao "količina vremena koju istraživač planira da provede na projektu". Relacija "koautorstvo" je prirodna veza između istraživača i sličnog



Slika 2. Primjer povezanosti nekih lekcija

je značenja kao pomenuta veza "mogućnost zajedničkog rada". Tako se problem svodi na određivanje particije skupa svih istraživača, tako da ukupno vrijeme potrošeno na rad na projektu u obje grupe bude skoro podjednako.

### 2.1. Analiza upotrebe MBCP problema u kreiranju kursa

U ovom odjeljku ćemo prikazati kako se za dati kurs koji se sastoji od lekcija formira odgovarajući graf, na koji se mogu primijeniti tehnike za rješavanje MBCP problema.

Kurs ćemo predstaviti kao povezan graf kod koga čvorovi odgovaraju lekcijama. Dvije lekcije su povezane samo ako postoji direktna semantička ili metodološka veza. Preciznije, veze između lekcija se uspostavljaju u slučaju uopštenja, analogije, uslovljenosti ili obje lekcije obrađuju slične matematičke strukture.

**Primjer 2.** Primjer veza između lekcija iz oblasti Teorije brojeva je prikazan na slici 2. Na primjer, uspostavljena je semantička veza između lekcija Prosti brojevi i Savršeni brojevi, dok je između lekcija Prosti brojevi i Najveći prost broj uspostavljena veza uslovljenosti. Između lekcija Euklidov algoritam za pronalaženje NZD i Eratostenovo sito se uspostavlja veza sličnosti. Ostale relacije se identifikuju na analogan način.

Treba napomenuti da je određivanje veza između lekcija fleksibilno i u priličnoj mjeri može da zavisi od nastavnikovog mišljenja i subjektivne procjene.

Određivanje težine lekcija takođe može biti predmet subjektivne procjene nastavnika i zavisi od nastavnikove procjene o potrebnoj količini gradiva, nivoa prezentacije lekcije, ciljeva kursa, studentskih sposobnosti i slično. U svakom slučaju, mogu se uspostaviti neki objektivni kriterijumi za određivanje težine lekcije. Da bi se "izmjerila" težina lekcije, potrebno je uvesti faktore koji utiču na težinu:

- Faktor I: obim lekcije;
- Faktor II: nivo tipičnih studentskih sposobnosti i znanja prije početka slušanja lekcije;
- Faktor III: važnost lekcije za naredne lekcije;
- Faktor IV: važnost lekcije za matematiku uopšte.

Jasno je da Faktor I utiče na težinu lekcije, jer su lekcije koje sadrže više definicija i teorema, generalno teže nego kraće lekcije. Drugi faktor se povezuje sa nivoom potrebnog studentovog znanja i sposobnosti koje su potrebne da bi lekcija bila lakše razumljiva. Posljednja dva faktora ukazuju na nivo detaljnosti do kog se lekcija obrađuje, u smislu da lekcije koje su preduslovi za naredne lekcije ili generalno matematičko znanje trebaju biti obrađivane detaljnije. Da bi se odredila težina svake lekcije, uvodimo sljedeću notaciju: Neka koeficijenti  $p_1, p_2, p_3$  i  $p_4$  označavaju uticaj faktora (I-IV) i neka je  $L$  lekcija. Ako sa  $f_1(L), f_2(L), f_3(L)$  i  $f_4(L)$  označimo vrijednosti dodijeljene svakom faktoru (I-IV) za datu lekciju  $L$ , tada se težina lekcije  $L$  računa pomoću formule

$$w(L) = 10 \sum_{i=1}^4 p_i f_i(L) \quad (1)$$

Vrijednosti  $p_i, i = 1..4$  se biraju iz intervala  $[0, 1]$  i važi da je ukupna suma

$$\sum_{i=1}^4 p_i = 1. \quad (2)$$

**Tabela 1.** Predložene težine za lekcije iz primjera 3

R.b.	Ime lekcije	$f_1$	$f_2$	$f_3$	$f_4$	$w(L)$
1	Najveći zajednički djelilac	5	1	6	10	49
2	Uzajamno prosti brojevi	3	4	2	2	29.5
3	Prosti brojevi	10	4	10	10	85
4	Savršeni brojevi	5	2	3	8	44
5	Euklidov algoritam za pronalaženje NZD-a	2	1	2	5	22
6	Eratostenovo sito	2	1	2	5	22
7	Najveći prost broj	2	5	2	6	33.5
8	Najveći savršen broj	2	5	2	6	33.5

Način izbora vrijednosti koeficijenta  $p_i$ ,  $i = 1..4$  može da varira od tipa kursa, namjene kursa (na primjer da li je kurs uvodni kurs za neke naredne ili završni kurs u nekoj oblasti), nivoa procijenjenih studentskih sposobnosti. Na primjer, ako je riječ o nekom polaznom kursu, koji treba da bude osnova za naredne kurseve, prirodno je da se postavi veće vrijednosti za koeficijente  $p_1$  i  $p_3$ . Ako je riječ o kursu koji je opšte namjene i vezan je za opšte matematičko znanje, onda se parametru  $p_4$  može dodijeliti veća vrijednost.

**Primjer 3.** Neka su koeficijenti  $p_i$ ,  $i = 1..4$  postavljeni na vrijednosti:  $p_1 = 0.45$ ,  $p_2 = 0.25$ ,  $p_3 = 0.15$  i  $p_4 = 0.15$ . To znači da se pri određivanju težine lekcije obimu lekcija daje najveća važnost, dok se izborom nižih vrijednosti za koeficijente  $p_3$  i  $p_4$  definiše manji uticaj lekcija na kasnije lekcije i opšte matematičko znanje.

Poželjno je vrijednosti težina svih lekcija ograničiti u neki unaprijed zadati interval. Ako želimo da za lekciju  $L$  vrijednost težine  $w(L)$ , koja se računa pomoću formule (1), pripada intervalu  $[0, 100]$ , i s obzirom na (2), vrijednost sva četiri faktora treba skalirati u interval  $[0, 10]$ .

**Primjer 4.** Posmatrajmo neke lekcije koje čine dio kursa iz Teorije brojeva, prikazane na slici 2. Za svaku lekciju  $L$  određujemo vrijednosti  $f_i(L)$ ,  $i = 1..4$ . Težina za prikazane lekcije su predložene u tabeli 1, a vrijednosti za  $p_i$ ,  $i = 1..4$  su iste kao u primjeru 3.

Kao što vidimo iz tabele 1, lekciji Prosti brojevi su dodijeljene maksimalne vrijednosti za faktore  $f_1$ ,  $f_3$  i  $f_4$ . Time se određuje da je lekciji Prosti brojevi, zbog obima lekcije i njene primjene kako u daljim lekcijama, tako i u matematičkom znanju uopšte, dodijeljena veća ukupna težina, koja iznosi 85. Lekcije koje se odnose na algoritme za pronalaženje NZD-a (Euklidov algoritam), odnosno prostih brojeva (Eratostenovo sito) ne zahtijevaju visoko predznanje, a po obimu su relativno male. Ovim lekcijama je dat malo veći značaj za poznavanje matematike uopšte ( $f_4 = 5$ ), jer se može smatrati da je poznavanje Euklidovog algoritma i principa rada Eratostenovog sita čak i stvar opšte kulture. Sa druge strane, faktoru  $f_4$  je dodijeljen manji uticaj ( $p_4 = 0.15$ ) na cjelokupnu težinu, tako da sveukupno gledano, ove dvije lekcije nemaju veliku ukupnu težinu i ona za obje lekcije iznosi po 22.

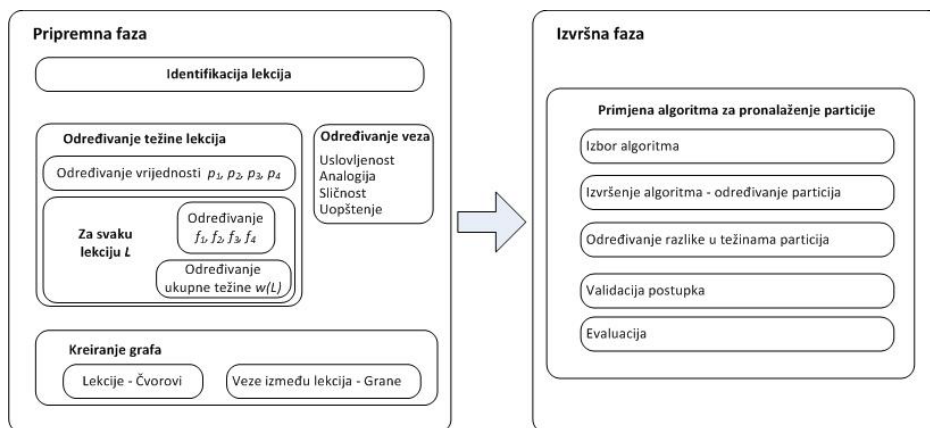
Lekcije koje se odnose na određivanje najvećeg prostog broja (naravno, pošto je skup prostih brojeva beskonačan, misli se na određivanje najvećeg prostog broja koji se može tačno odrediti) i najvećeg poznatog savršenog broja po obimu nisu velike, ali podrazumijevaju poznavanje tehnika teorije brojeva, kao i poznavanje tehnika koje se odnose na rad sa velikim brojevima na računaru. Dalje, ove dvije lekcije nisu od velikog značaja za nastupajuće lekcije, dok im se pridaje srednji značaj za matematičko znanje uopšte.

U ovom primjeru ne analiziraju se sve lekcije iz date oblasti Teorije brojeva. Za detaljniju analizu lekcija iz ove oblasti, kao i njihovih veza, preporučuje se [10–12].

## 2.2. Određivanje particije datog skupa lekcija

Određivanje particije datog skupa lekcija, prema opisanom kriterijumu se dalje vrši odgovarajućim tehnikama vezanim za metode rješavanja MBCP problema. Čitav proces organizacije kursa po ovom modelu se izvršava po šemi prikazanoj na slici 3.

U pripremnoj fazi, vrši se identifikacija lekcija, kao i određivanje težine za svaku lekciju, preko odgovarajućih faktora  $f_1 - f_4$ . Pored toga, određuju se i veze na osnovu navedenih kriterijuma. Nakon toga se na osnovu kursa kreira odgovarajuća matematička struktura - graf, tako što se lekcije predstavljaju čvorovima, a veze između lekcija granama. U drugoj fazi, prvo se bira algoritam za rješavanje MBCP problema, koji se koristi za određivanje particije. S obzirom da je problem  $NP$  težak, egzaktne metode ne mogu rješavati probleme većih dimenzija. Eksperimenti pokazuju da se direktne egzaktne metode kao što je totalna enumeracija, bez dodatnih poboljšanja i ubrzanja mogu koristiti na grafove do 40 čvorova (odnosno kurseve do 40 lekcija). Za grafove većih dimenzija koriste se heurističke metode. Metode za rješavanje MBCP problema su prikazane u sljedećem odjeljku. Nakon



Slika 3. Postupak za određivanje podjele kursa

izvršenog algoritma, vrši se validacija dobijenih rezultata (ispitivanje da li je dobijena particija odgovarajuća), kao i evaluacija čitavog postupka.

### 3. Metode za rješavanje MBCP problema

Kao što je već rečeno, MBCP je NP težak problem tako da nijedan deterministički algoritam ne rješava opšti slučaj problema u polinomijalnom vremenu. Stoga se egzaktne metode mogu koristiti samo za male grafove, jer vrijeme izvršenja ovakvih algoritama eksponencijalno raste sa dimenzijom problema. Za veće instance (grafove), jedini način da se dostignu kvalitetna rješenja je primjena neke od heurističkih metoda.

U ovom odjeljku detaljnije ćemo opisati jednu egzaktnu metodu (metodu totalne enumeracije) i jednu heurističku metodu (pohlepni algoritam).

#### 3.1. Metoda totalne enumeracije

Metoda totalne enumeracije je zasnovana na ispitivanju svih mogućih rješenja. Time se garantuje da će, po završetku algoritma, biti pronađeno optimalno rješenje. Sa druge strane, s obzirom na vremenska i memorijska ograničenja, metoda totalne enumeracije je teško primjenljiva na mnoge realne probleme. Stoga se ona uglavnom koristi za rješavanje problema manjih dimenzija ili, uz dodatna prilagođavanja, za verifikovanje rezultata dobijenih nekim drugim metodama.

Za dati graf  $G = (V, E)$ , svako potencijalno rješenje MBCP problema se može predstaviti binarnim nizom  $x$  dužine  $|V|$ . Elementi niza odgovaraju čvorovima, određujući kojoj komponenti odgovarajući čvor pripada:  $i \in V_1$  ako  $x_i = 1$  i  $i \in V_2$  ako  $x_i = 0$ . Očigledno je da svakoj particiji (potencijalnom rješenju koje daje algoritam totalne enumeracije) odgovara jedan niz dužine  $|V|$ , sa elementima iz  $\{0, 1\}$ . Kako je ukupan broj ovakvih nizova  $2^{|V|}$ , algoritam totalne enumeracije mora da analizira ukupno  $2^{|V|}$  potencijalnih rješenja. Ovim vidimo da se ovakav algoritam može koristiti samo za male grafove.

**Primjer 5.** Pretpostavimo da je ulazni graf kao onaj iz Primjera 1. S obzirom da graf ima 6 čvorova, algoritam totalne enumeracije treba da analizira ukupno  $2^6 = 64$  binarnih nizova dužine 6. Svaki niz odgovara jednom potencijalnom rješenju. Na primjer, niz 100010 odgovara particiji  $(\{1, 5\}, \{2, 3, 4, 6\})$ , ali ova podjela indukuje nepovezane komponente, te se ona zbog toga odbacuje. Optimalnom rješenju  $(\{3, 4, 5\}, \{1, 2, 6\})$  prikazanom na slici 1 odgovara niz 110001.

#### 3.2. "Pohlepni" algoritam za rješavanje MBCP problema

"Pohlepni" (eng. greedy) algoritam je heuristička metoda za rješavanje problema koja je zasnovana na principu izbora najboljeg lokalnog rješenja u svakom koraku, sa nadom da će se tako pronaći i optimalno rješenje. Iako u nekim slučajevima ovaj pristup zaista i dovodi do optimalnog rješenja (na primjer Kruskal-ov algoritam za pronalaženje minimalnog pokrivajućeg stabla [13]), u opštem slučaju algoritam ne može da pronađe optimalno rješenje, a u mnogim slučajevima je rješenje pronađeno pohlepnim algoritmom znatno lošije od optimalnog. Sa druge strane, ova metoda može biti korisna za pronalaženje polaznog dopustivog rješenja, čiji se kvalitet može dalje poboljšavati drugim tehnikama.

U slučaju MBCP problema, prikazaćemo pohlepni algoritam zasnovan na "balansiranju blokova" (eng. block balance algorithm) iz [1].

Neka je  $G = (V, E)$  dati graf. Algoritam počinje sa povezanim particijama  $(V_1, V_2)$ , takvim da  $V_1$  sadrži samo jedan čvor, i to onaj maksimalne težine. Tokom iterativnog procesa, algoritam traži čvor  $u$  iz  $V_2$ , takav da obje particije  $V_1 \cup \{u\}$  i  $V_2 \setminus \{u\}$  ostanu povezane.

Među svim takvim čvorovima, bira se onaj sa minimalnom težinom. Algoritam završava sa radom kada dodavanje novog čvora u  $V_1$  više ne poboljšava particiju.

Detaljniji opis pohlepnog algoritma, kao i ilustracija njegovog rada je prikazana u [1, 12].

#### 4. Zaključak

U ovom radu je prezentovan problem pronalaženja maksimalno balansirane povezane particije u grafu i njezova primjena na rješavanje nekih organizacionih problema u nastavi. MBCP pripada širokoj klasi problema koji se bave pronalaženjima particija u grafu i kao takav ima veliku primjenu u rješavanju problema u raznim oblastima inženjerstva, menadžmenta i nekim društvenim pitanjima. U obrazovanju, ovaj problem se može direktno iskoristiti u organizovanju lekcija kursa u dvije povezane cjeline koje su ujednačene po težini.

U radu je opisana metodologija određivanja težina lekcija i način povezivanja lekcija, čime se omogućava da se čitav kurs predstavi kao matematička struktura - graf, na koji se primjenjuju tehnike za rješavanje MBCP problema. S obzirom na činjenicu da je problem NP težak, razvijaju se razne heurističke metode koje se koriste za rješavanje problema veće dimenzije, dok se za manje grafove može koristiti algoritam totalne enumeracije.

Ovo istraživanje se može proširiti u nekoliko smjerova. Bilo bi interesantno primijeniti ovaj metod organizacije na kurseve iz raznih matematičkih i drugih oblasti, te ispitati kvalitet predloženog pristupa nekom evaluacionom metodom. Takođe, ovaj rad može da motiviše istraživače za razvoj novih tehnika za rješavanje opisanog problema kombinatorne optimizacije.

#### Literatura

- [1] **J. Chlebikova**, Approximating the Maximally Balanced Connected Partition Problem in graphs, *Information Processing Letters*, 1996, 60, 225–230.
- [2] **B. Djurić, J. Kratica, D. Tosić, V. Filipović**, Solving the Maximally Balanced Connected Partition Problem in Graphs by Using Genetic Algorithm, *Computers and Informatics*, 2008, 27(3), 341–354.
- [3] **F. Chataigner, L.R.B. Salgado and Y. Wakabayashi**, Approximation and inapproximability results on balanced connected partitions of graphs, *Discrete Mathematics & Theoretical Computer Science*, 2007, 9, 177–192.
- [4] **T. Ito, X. Zhou, T. Nishizeki**, Partitioning a graph of bounded tree-width to connected subgraphs of almost uniform size, *Journal of Discrete Algorithms*, 2006, 4, 142–154.
- [5] **T. Ito, T. Uno, X. Zhou, T. Nishizeki**, Partitioning a Weighted Tree to Subtrees of Almost Uniform Size, *Lecture Notes in Computer Science*, 2008, 5369, 196–207.
- [6] **I. Slama, B. Jouaber, D. Zeghlache**, Topology Control and Routing in Large Scale Wireless Sensor Networks, *Wireless Sensor Network*, 2010, 2, 584–598.
- [7] **M. Lucertini, Y. Perl, B. Simeone**, Most uniform path partitioning and its use in image processing, *Discrete Applied Mathematics*, 1993, 42(2-3), 227–256.
- [8] **B. Bozkaya, E. Erkuť, G. Laporte**, A tabu search heuristic and adaptive memory procedure for political districting, *European Journal of Operational Research*, 2003, 144(1), 12–26.
- [9] **T. Assunção, V. Furtado**, A Heuristic Method for Balanced Graph Partitioning: An Application for the Demarcation of Preventive Police Patrol Areas, in: *H. Geffner, R. Prada, I. M. Alexandre, N. David (Eds.), Advances in Artificial Intelligence - Iberamia 2008, LNCS 5290/2008 Springer Berlin/Heidelberg*, 2008, pp. 62–72.
- [10] **V. Mićić, Z. Kadelburg**, Uvod u teoriju brojeva, drugo dopunjeno izdanje, *Društvo matematičara Srbije, Beograd*, 1989.
- [11] **Hardy, G. H., Wright, E. M.**, An Introduction to the Theory of Numbers (fifth ed.), *USA: Oxford University Press*, 1979.
- [12] **Matić, D., Božić, M.**, Maximally Balanced Connected Partition Problem in Graphs: Application in Education, *The Teaching of Mathematics*, 2012, 15(2), 121–132.
- [13] **Kruskal, J.**, On the Shortest Spanning Subtree of a Graph and the Traveling Salesman Problem, *Proceedings of the American Mathematical Society* 7(1), 1956, pp. 48–50

Received



# Област „Мерење” у настави математике у основној школи

Петар Огривовић

Образовни систем „Руђер Бошковић”, Београд, petogriz@yahoo.com

**Апстракт.** Анализиран је задатак „Разговор преко интернета” (ПИСА, 2003. година) кроз резултате тестирања 176 ученика Основне школе и Гимназије „Руђер Бошковић” у Београду. Показало се да део овог задатка, намењеног ученицима који имају петнаест година, успешно решавају и млађи основци. С тим у вези, разматрана је област „Мерење”, која постоји у Образовним стандардима за крај обавезног образовања, али је нема у наставним програмима за старије разреде основне школе. Дато је и неколико примера и предлога како се мерење може детаљније обрадити кроз друге наставне области.

**Кључне речи.** Мерење, образовни стандарди, ПИСА, настава математике

## 1. Увод

Међународна упоредна истраживања ТИМСС<sup>1</sup> и ПИСА<sup>2</sup> показала су, почетком двадесет првог века, да је образовни систем у Републици Србији недовољно функционалан. Наши ученици на овим тестирањима имају нижа постигнућа у односу на своје вршњаке из других европских земаља, посебно када је реч о употреби стечених знања у новим ситуацијама. Због тога је, током претходне деценије, започета реформа школства, у оквиру које је, између осталог, усвојен нови Закон о основама система образовања и васпитања, начињене су измене у наставним програмима и плановима, уведени су завршни испити, а Национални просветни савет усвојио је, 19. маја 2009. године, Образовне стандарде за крај обавезног образовања.

## 2. ПИСА

Међународни програм процене ученичких постигнућа ПИСА утврђује да ли су ученици који завршавају период општег образовања, узраста 15 година, усвојили знања и вештине неопходне за наставак школовања, професионални развој и одговорно учествовање у цивилном друштву. Постигнуће се процењује у области математичке, читалачке и природно-научне писмености [2]. Програм ПИСА инициран је са циљем да се систематски прати квалитет и праведност образовања у земљама учесницама. ПИСА не испитује у којој мери ученици могу да репродукују оно што су научили у школама, већ у којој су мери оспособљени да разумеју и користе дате информације приликом решавања релевантних проблема из свакодневног живота [1].

ПИСА је, у овом тренутку, највеће међународно истраживање у области образовања. Реализује се у организацији ОЕЦД<sup>3</sup> од 1997. године. Тестирања ученика организују се сваке треће године, почев од 2000. У истраживању 2009. године учествовале су 74 земље, од тога 34 чланице ОЕЦД и 40 партнерских земаља [2].

Одлуком Министарства просвете и спорта, Србија учествује у програму ПИСА од 2001. године, а наши ученици су до сада тестирани четири пута – 2003, 2006, 2009. и 2012. године.

<sup>1</sup> Trends in International Mathematics and Science Study (TIMSS)

<sup>2</sup> Programme for International Student Assessment (PISA)

<sup>3</sup> Organisation for Economic Co-operation and Development (OECD)

У свим земљама у којима се спроводи ПИСА ученици решавају исте задатке. Тестирање се обавља током априла и маја, а за решавање задатака ученици имају на располагању два сата. Математички садржаји на које се ослањају проблеми у задацима смештени су у четири тематске области: простор и облик, трансформације и релације, бројеви и мере, неизвесност.

Скала математичке писмености конструисана је тако да је просечан скор 500. Скала је подељена на шест нивоа постигнућа, а сматра се да су функционално писмени они ученици чија су постигнућа на другом и вишим нивоима.

На скали математичке писмености ученици у Србији су 2009. године у просеку постигли 442 поена, док је 2006. године тај просек био мањи за 7 поена. Иако постоји напредак у односу на прва тестирања, забрињава податак да око 40% ученика из Србије није достигло ниво функционалне писмености из математике, као и да ђаци у нашој земљи бележе за 60 до 70 поена нижа постигнућа из математике него што је просек у земљама ОЕИЦД. Један од разлога за то треба тражити и у чињеници да су на ПИСА тестирањима заступљени задаци који се тичу обраде података и мерења. Те две теме, до недавно, нису биле (довољно) присутне у настави математике у Србији.

### **3. Мерење – образовни стандарди и наставни програми**

Наставни садржаји у оквиру предмета математика нису претрпели веће измене у најновијој реформи образовног система. Поједине лекције су преименоване, неколико наставних јединица је изостављено, а изванредан број часова посвећен је новој теми названој обрада података. Са друге стране, у оквиру новоуведених образовних стандарда за математику издвојено је пет области - „Бројеви и операције са њима”, „Алгебра и функције”, „Геометрија”, „Обрада података” и „Мерење”.

Образовни стандарди из области „Мерење” тичу се коришћења одговарајућих јединица за мерење дужине, површине, запремине, масе, времена и угла, као и различитих апоена новца. Следе претварање јединица из већих у мање и обрнуто, употреба приближних вредности, процена и заокругљивање датих података, као и рачунање са њима. Реч је, дакле, о садржајима који се углавном обрађују у разредној настави, а делом и на часовима физике, док се на часовима математике од петог до осмог разреда основне школе мерење појављује у оквиру других садржаја, и то најчешће геометријских.

Од постојећих 66 образовних стандарда из математике, у области мерење дато их је 9, док их је из геометрије 18, што није у складу са заступљеношћу ове две области у настави. Наиме, геометрија заузима више од трећине часова математике од петог до осмог разреда основне школе, док за мерење, као посебну тему, није предвиђен ниједан час.

### **4. Пример ПИСА задатка из области мерење**

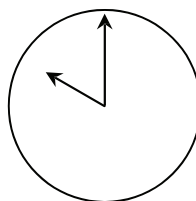
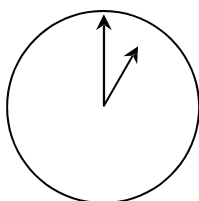
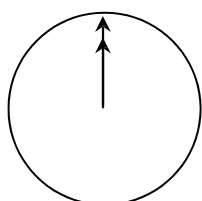
У оквиру припрема за ПИСА тестирање у Србији 2012. године Институт за психологију Филозофског факултета Универзитета у Београду и Министарство просвете и науке иницирали су објављивање серије текстова са ПИСА задацима у дневном листу „Политика”. У једном од тих текстова анализиран је задатак „Разговор преко интернета”, који припада области мерење. Задаци попут овог ретко се могу наћи у збиркама чија је употреба одобрена у основним школама у Србији.

#### **4.1. Задатак: Разговор преко интернета**

Марк из Сиднеја у Аустралији и Ханс из Берлина у Немачкој међусобно комуницирају користећи „chat” на интернету. Да би могли да разговарају морају да се прикључе на интернет у истом тренутку.



Тражећи одговарајуће време за „chat“, Марк је консултовао карту часовних зона и нашао је следеће:



Гринич 24 сата (поноћ)    Берлин 1 сат после поноћи    Сиднеј 10 сати ујутру

Питање 1: Када је у Сиднеју 19 сати, колико је сати у Берлину?

Питање 2: Марк и Ханс не могу да разговарају између 9 сати ујутру и 16:30 сати поподне по њиховим локалним временима, зато што морају да иду у школу. Исто тако, неће моћи да разговарају између 23:00 увече и 7:00 ујутру по њиховим локалним временима, зато што тада спавају.

Када Марк и Ханс могу да разговарају? Упиши локално време у табелу.

Место	Време
Сиднеј	
Берлин	

#### 4.2. Тестирање ученика

У циљу провере сложености задатка „Разговор преко интернета“, као и стратегија за његово решавање, тестирана је група од 176 ученика две београдске школе - 94 ученика Основне школе „Руђер Бошковић“ (четврти, шести и осми разред) и 82 ученика Гимназије „Руђер Бошковић“ (први и други разред).

Ученици осмог разреда основне школе и првог разреда гимназије изабрани су за ово тестирање јер су ПИСА тестови осмишљени управо за њихов узраст (15 година). У тестираној групи су, циљано, били и млађи и старији ученици. Четврти разред основне школе (10 година) изабран је зато што је у питању последња година када ученици, у оквиру разредне наставе, обрађују мерење времена на часовима математике. Шести разред основне школе (12 година) и други разред гимназије (16 година) тестирани су зато што су ученици у претходном разреду (петом, односно првом), обрађивали временске зоне у оквиру предмета географија. Овакав избор направљен је на основу претпоставки да део задатка могу да реше и млађи основци, као и да су старији ученици успешнији у решавању.

У Табели 1. приказани су резултати тестирања.

Табела 1. Процент тачно решених задатака

	Основна школа		Гимназија		
	1. задатак	2. задатак		1. задатак	2. задатак
4. разред	43%	0%	1. разред	79%	36%
6. разред	49%	0%	2. разред	75%	37%
8. разред	68%	36%			

#### 4.3. Како ученици решавају задатак

Ученици овај задатак решавају тако што прво одреде временску разлику између Берлина и Сиднеја (9 сати), а затим од 19 часова, колико је у Сиднеју, одузимају добијени број. Тако долазе до одговора да је у Берлину 10.00 сати ујутру.

Иако питање делује једноставно, оно одговара трећем нивоу математичке писмености. Статистика показује да је у земљама ОЕЦД на њега тачно одговорило 54% тестираних ученика.

У тестираној групи остварени су бољи резултати у узрасту од 15 година. Као што је било и очекивано, први део задатка умели су у великом проценту да реше и млађи ученици. Многима од њих, међутим, није било јасно да ли број 9 треба да одузму од 19 или је потребно сабрати те бројеве.

Оно што је додатно отежало решавање задатке јесте чињеница да је потребно израчунати времена у два града (Берлин и Сиднеј), а у уводном делу задатка, такозваном стимулусу, приказана су три часовника. Време по Гриничу овде служи само као „ометач” (тзв. „дистрактор”) и задатак се може решити без тог податка. У задацима на ПИСА тестирањима често се у тексту појављују сувишни подаци, а од ученика се очекује да прво издвоје битне информације, а тек потом приступе израчунавањима. Са друге стране, у збиркама из математике у Србији преовлађују задаци у којима је дато тачно онолико података колико је неопходно да се дође до решења.

Друго питање у задатку (Када Марк и Ханс могу да разговарају?) је сложеније, припада петом нивоу математичке писмености, а ОЕЦД просек је 29% тачних одговора, док читавих 19,2% ученика није ни покушало да га реши.

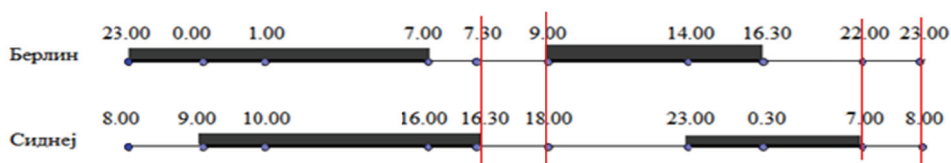
Оно што отежава решавање овог дела задатке је чињеница да није довољно само разумети разлику у временским зонама, већ је потребно и да се одреде решења која задовољавају већи број услова. Млађе ученике збунио је појам „по њиховим локалним временима”, па су један исти одговор записивали и у првом и у другом реду табеле, што је нетачно. Неки од ученика су наслутили да постоји више од једног дела дана када Марк и Ханс могу да разговарају, али нису знали како да одреде све термине. У тестираној групи било је 66 ученика четвртог и шестог разреда, а само је двоје успело да тачно реши овај задатак.

Ситуација је значајно боља у групи ученика осмог разреда и међу гимназијалцима. Чак 36% њих дошло је до тачног одговора. Закључак је да су у седмом и осмом разреду основне школе стекли знања која им олакшавају решавање оваквих проблема.

Једна стратегија за решавање овог дела задатке била је исписивање низа података (табеле) о времену за Сиднеј и за Берлин, након чега је следило прецртавање интервала који не задовољавају услове задатка.

Друга идеја била је да се на нацртаним часовницима осенче, односно затамне делови дана када Марк и Ханс не могу да разговарају. Неосенчени делови представљају решење.

Слична замисао појављује се и у трећој опцији решења (цртеж 1), где су ученици отишли и корак даље у визуелизацији проблема. Они нису цртали часовнике, већ две бројевне праве - на једној су означавали временске интервале за Сиднеј, а на другој за Берлин. Након тога одређивали су пресек. Овакав начин записивања интервала подсећа на решавање линеарних неједначина, које се детаљно обрађују у осмом разреду основне школе.



Цртеж 1. Пример решења

Одговор је:

Место	Време
Сиднеј	од 16.30 до 18.00 или од 7.00 до 8.00
Берлин	од 7.30 до 9.00 или од 22.00 до 23.00

## 5. Мерење на часу математике

Тестиралим ученицима је, након решавања, постављено и питање: „Из ког школског предмета је задатак 'Разговор преко интернета'?" Више од половине њих навело је географију (односно познавање природе и друштва у 4. разреду основне школе), а у неколико одговора поменуте су и физика, логика и тек потом математика. Испитаници, дакле, ређе везују садржај задатка са математиком, што потврђује и претходно разматрање о мерењу као области која није јасно издвојена у настави овог предмета у старијим разредима.

Чак и када се у збиркама из математике појаве задаци који се тичу мерења, полази се од претпоставки да су ученици усвојили појмове и технике у оквиру неког другог предмета, или да их знају из млађих разреда. Ређе се на часовима математике нешто заиста и мери. Уместо да од ученика очекујемо да умеју да гледају на сат, разумеју концепт временских зона, користе мерне јединице, схватају како се мере дужине, површине и запремине, то бисмо, кроз различите садржаје, могли да их научимо или да им помогнемо да утврде оно што већ знају.

Најједноставнији примери тичу се мерења дужина дужи и обима фигура. Ученицима можемо задати да одреде обим неке фигуре која је нацртана на папиру. На оваквом задатку можемо, већ у петом разреду, видети да ли и колико схватају појам дужине. Исти захтев може бити интересантан и за старије ученике, који су навикли да им у геометријским задацима буду унапред задати подаци о дужинама, а да непознате одређују рачунским путем.

У теми „Угао” у петом разреду мерење се појављује када се први пут користи угломер. Наставак теме углавном води ка рачунском одређивању збира и разлике углава. Ови садржаји се могу проширити задатком мерења упадног угла сунчевих зрака. За решавање тог проблема потребно је изаћи у школско двориште, употребити угломер (за таблу), парче канапа и сенку коју прави неки објекат. Довољно је канапом повезати тачку на објекту и одговарајућу тачку на слици (сенци), а затим угломером измерити под којим углом падају сунчеви зраци.

За увођење децималних бројева могу се користити различити дигитални мерачи (за температуру, масу), а слични примери могу се употребити и за увођење негативних величина.

Мотивација за увођење ирационалног броја  $\pi$  у седмом разреду у складу је са историјском потребом за тим бројем. Да би се показало да ова константа представља количник обима и пречника круга потребно је на часу, уз помоћ канапа и метра, премерити обиме и пречнике неколико кругова различитих полупречника (компакт-диск, шерпа, тањир, дно чаше и слично). Имајући у виду да је у питању однос који није представљен целим бројем, на овом часу би ученици могли да употребе и калкулатор за израчунавање поменутог количника. Записивање (на табли) табеле у којој су подаци о обимима, пречницима и њиховом количнику, ученике води ка закључку о приближној вредности броја  $\pi$ .

Ово су само неки од примера који илуструју чињеницу да постоји простор за „Мерење” у оквиру наставе математике.

## 6. Закључак

Потребно је посветити додатну пажњу мерењу на часовима математике, и то из више разлога. Оно је заступљено на међународним тестирањима на којима учествују ученици из Србије, постоји као засебна област у Образовним стандардима и појављује се у задацима на завршном испиту. Ипак, оно што је најважније за свакодневни рад са ученицима је чињеница да се кроз задатке из области „Мерење” остварују многи циљеви наставе математике у основној школи, попут, на пример, оспособљавања ученика за примену усвојених математичких знања у решавању задатака из животне праксе.

Тема „Мерење” пружа бројне могућности за остваривање корелација и у оквиру самог предмета (са геометријом, обрадом података), али и са другим наставним предметима (физика, хемија, географија, техничко и информатичко образовање).

Анализирани ПИСА задатак, и њему слични, помажу ученицима да уоче употребну вредност онога што уче, и тиме повећавају мотивацију за учење математике.

## Литература

[1] Драгица Павловић Бабић, Александар Бауцал. Математичка писменост: ПИСА 2003 и ПИСА 2006. *Министарство просвете Републике Србије, Завод за вредновање квалитета образовања и васпитања, Институт за психологију Филозофског факултета Универзитета у Београду, Београд, 2009.*

[2] Александар Бауцал, Драгица Павловић Бабић. Научи ме да мислим, научи ме да учим: ПИСА 2009 у Србији; први резултати. *Институт за психологију Филозофског универзитета у Београду, Центар за примењену психологију, Београд, 2010.*

[3] уредник Гордана Чапрић. Образовни стандарди за крај обавезног образовања. *Министарство просвете Републике Србије, Завод за вредновање квалитета образовања и васпитања, Београд, 2009.*

[4] Christian Monseur, Ariane Baye, Dominique Lafontaine, Valérie Quittre. PISA test format assessment and the local independence assumption. *University of Liège, Belgium, 2011.*

[5] Gerry Shiel, Rachel Perkins, Seán Close, Elizabeth Oldham. PISA mathematics: a teacher's guide. *Department of Education and Science, Dublin, 2007.*

## PROBLEM SOPSTVENIH VREDNOSTI SA NELOKALNIM ROBIN-DIRIHLEOVIM GRANIČNIM USLOVIMA

Sandra Hodžić

Matematički fakultet  
sandra@matf.bg.ac.rs

**Abstract.** Šturm-Liuvilove jednačine se pojavljuju u problemima primenjene matematike. Na primer, one predstavljaju vibraciona stanja različitih sistema, kao što su vibracije žice ili energetske sopstvene funkcije kvantno-mehaničkih oscilatora. To su granični problemi običnih diferencijalnih jednačina, koji se još nazivaju i problemima sopstvenih vrednosti. Ovde je izložen i rešen problem sa tzv. nelokalnim Robin-Dirihleovim graničnim uslovima sa primerima koji ilustruju prethodna teorijska razmatranja.

**Keywords:** Šturm-Liuvilova jednačina, granični problem

## Problem sopstvenih vrednosti sa nelokalnim Robin-Dirihleovim graničnim uslovima

Razmotrimo sledeći granični problem:

$$(1) \quad -u''(x) = \lambda u(x), \quad 0 \leq x \leq 1$$

$$(2) \quad \begin{aligned} u'(0) &= \sigma_0 u(0) - \rho_0 u(1) \\ u'(1) &= -\sigma_1 u(1) + \rho_1 u(0) \end{aligned}$$

Uslovi (2) nazivaju se nelokalni Robin-Dirihleovi granični uslovi. U njima se pojavljuju vrednosti nepoznate funkcije i njenih izvoda u obe granične tačke. Primitimo odmah da se za  $\rho_0 = \rho_1 = 0$  problem svodi na Robinov, a ako je još i  $\sigma_0 = \sigma_1 = 0$ , onda se radi o Nojmanovom graničnom problemu. Tražimo pozitivne sopstvene vrednosti. Onda je  $\lambda = \mu^2$  i rešenje je oblika  $u = \sin(\mu x + \gamma)$ . Na dalje pretpostavljamo da je  $\gamma \neq k\pi/2$ . Kada  $u$  i  $u'$  zamenimo u (2), dobićemo sledeće dve jednačine:

$$\begin{aligned} \mu \cos \gamma &= \sigma_0 \sin \gamma - \rho_0 \sin(\mu + \gamma) \\ \mu \cos(\mu + \gamma) &= -\sigma_1 \sin(\mu + \gamma) + \rho_1 \sin \gamma. \end{aligned}$$

Primenjujući poznate trigonometrijske identitete za sinus i kosinus zbira uglova dobijamo iz prve jednačine

$$\mu \cos \gamma = \sigma_0 \sin \gamma - \rho_0 \sin \mu \cos \gamma - \rho_0 \cos \mu \sin \gamma,$$

a iz druge

$$\mu \cos \mu \cos \gamma - \mu \sin \mu \sin \gamma = -\sigma_1 \sin \mu \cos \gamma - \sigma_1 \cos \mu \sin \gamma + \rho_1 \sin \gamma.$$

Iz prve dalje, deleći je sa  $\cos \gamma$ , dobijamo:

$$\begin{aligned} \mu &= \sigma_0 \operatorname{tg} \gamma - \rho_0 \sin \mu - \rho_0 \cos \mu \operatorname{tg} \gamma, \\ (\sigma_0 - \rho_0 \cos \gamma) \operatorname{tg} \gamma &= \mu + \rho_0 \sin \mu, \end{aligned}$$

$$(3) \quad \operatorname{tg} \gamma = \frac{\mu + \rho_0 \sin \mu}{\sigma_0 - \rho_0 \cos \mu}.$$

Ako i drugu jednačinu podelimo sa  $\cos \gamma$  imamo

$$\begin{aligned} \mu \cos \mu - \mu \sin \mu \operatorname{tg} \gamma + \sigma_1 \sin \mu + \sigma_1 \cos \mu \operatorname{tg} \gamma - \rho_1 \operatorname{tg} \gamma &= 0, \\ \mu \cos \mu + \sigma_1 \sin \mu + (\sigma_1 \cos \mu - \mu \sin \mu - \rho_1) \frac{\mu + \rho_0 \sin \mu}{\sigma_0 - \rho_0 \cos \mu} &= 0. \end{aligned}$$

Množeći poslednju jednakost sa  $\sigma_0 - \rho_0 \cos \mu$ , dobijamo

$$\begin{aligned} \sigma_0 \mu \cos \mu + \sigma_0 \sigma_1 \sin \mu - \rho_0 \mu \cos^2 \mu - \sigma_1 \rho_0 \sin \mu \cos \mu + \sigma_1 \mu \cos \mu + \\ + \sigma_1 \rho_0 \sin \mu \cos \mu - \mu^2 \sin \mu - \mu \rho_0 \sin^2 \mu - \rho_1 \mu - \rho_0 \rho_1 \sin \mu &= 0, \end{aligned}$$

odnosno,

$$(\sigma_0 + \sigma_1) \mu \cos \mu - \rho_0 \mu (\cos^2 \mu + \sin^2 \mu) - \rho_1 \mu + (\sigma_0 \sigma_1 - \rho_0 \rho_1 - \mu^2) \sin \mu = 0,$$

$$(\sigma_0 + \sigma_1)\mu \cos \mu - \mu(\rho_0 + \rho_1) + (\sigma_0\sigma_1 - \rho_0\rho_1 - \mu^2) \sin \mu = 0,$$

$$(4) \quad (\sigma_0 + \sigma_1) \operatorname{ctg} \mu - \frac{\rho_0 + \rho_1}{\sin \mu} = \mu - \frac{\sigma_0\sigma_1 - \rho_0\rho_1}{\mu}.$$

Na taj način  $\mu$  dobijamo rešavanjem jednačine (4), posle čega  $\gamma$  dobijamo iz (3).

Koristeći programski paket MATLAB rešili smo jednačinu (4). Funkciju sa leve strane jednakosti (4) označimo sa  $f_1(\mu)$ , a sa desne strane sa  $f_2(\mu)$ . Rešenja  $\mu$  se nalaze u preseku grafika funkcija  $f_1(\mu)$  i  $f_2(\mu)$ . Program dozvoljava unos parametara  $\sigma_0, \sigma_1, \rho_0$  i  $\rho_1$ , a kao rezultat nam vraća grafike funkcija  $f_1(\mu)$  i  $f_2(\mu)$  sa istaknutim presečnim tačkama. Vrednosti  $\mu$  za koje važi  $f_1(\mu) = f_2(\mu)$  određene su numeričkim metodama. Kvadrati tih dobijenih vrednosti jesu upravo tražene sopstvene vrednosti. Rešenja tražimo na  $[0, 4\pi] \times [-20, 20] \subset \mathbb{R}^2$  kome pripadaju grafici ovih dveju funkcija. Najpre su sa tačnošću  $\varepsilon$ , tj. sa uslovom  $|f_1(\mu) - f_2(\mu)| < \varepsilon$  lokalizovane presečne tačke i iscrtni grafici. Tim postupkom su već nađene apscise  $\mu$  presečnih tačaka. Međutim, tačnost je poboljšana tako što su nađene tačke lokalnih minimuma funkcije  $|f_1(\mu) - f_2(\mu)|$ .

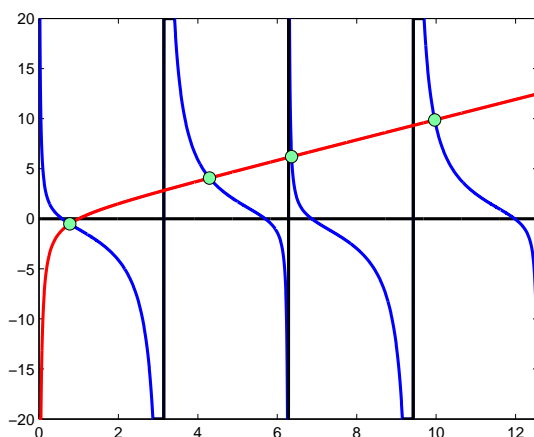
Ako je  $\sigma_0 + \sigma_1 = \rho_0 + \rho_1 = C$ , onda je  $f_1(\mu) = C(\operatorname{ctg} \mu - \frac{1}{\sin \mu})$ . Funkcija u zagradi je pozitivna na intervalima  $(k\pi, (k+1)\pi)$ ,  $k = 1, 3, 5, \dots$ , tj.  $\operatorname{ctg} \mu \geq \frac{1}{\sin \mu}$  na takvim intervalima. Uočavamo da postoje dva slučaja:

$$1) \sigma_0 + \sigma_1 \geq \rho_0 + \rho_1,$$

$$2) \sigma_0 + \sigma_1 < \rho_0 + \rho_1.$$

U prvom slučaju funkcija  $\operatorname{ctg} \mu$ , “dominira” nad funkcijom  $\frac{1}{\sin \mu}$ , a u drugom slučaju obrnuto. U drugom slučaju se može desiti da je presek grafika funkcija  $f_1(\mu)$  i  $f_2(\mu)$  prazan (na intervalu  $[0, 4\pi]$  koji smo uzeli za domen tih funkcija pri pisanju programa), a to znači da ne postoje rešenja  $\mu \in [0, 4\pi]$ , a samim tim ni odgovarajuće sopstvene vrednosti posmatranog graničnog zadatka. Što se tiče funkcije  $f_2(\mu)$ , to je, u opštem slučaju, kriva oblika  $\mu - C\frac{1}{\mu}$ ,  $C = \text{const}$ , ali za  $\sigma_i = \rho_j = 0$ ,  $i, j \in \{0, 1\}$ , grafik  $f_2$  predstavlja pravu  $\mu$ . Navedimo nekoliko primera.

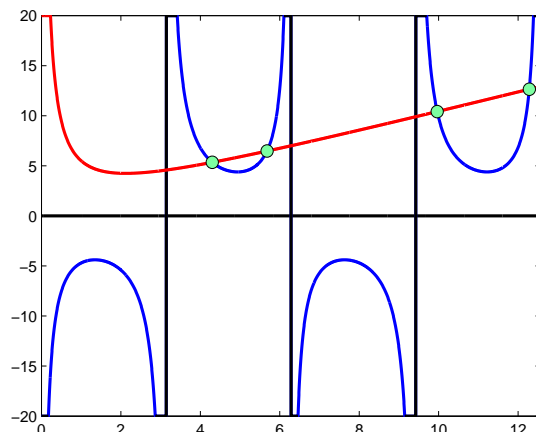
**Primer 1:**  $\sigma_0 = 1, \sigma_1 = 2, \rho_0 = 0.5, \rho_1 = 2$



Slika 1. *Slučaj 1)*

Prve četiri sopstvene vrednosti su: 0.6015, 18.4466, 40.4719, 99.2923

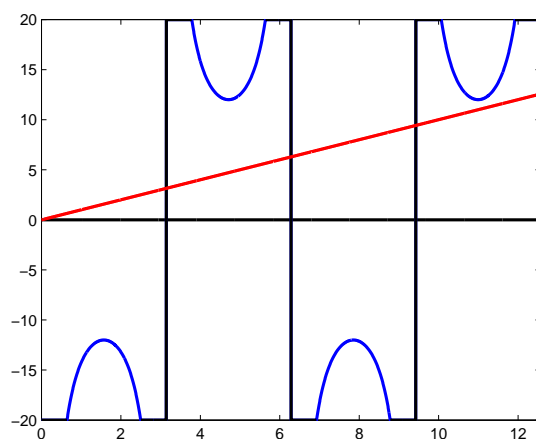
**Primer 2:**  $\sigma_0 = 0, \sigma_1 = 1, \rho_0 = 1.5, \rho_1 = 3$



Slika 2. *Slučaj 2)*

Prve četiri sopstvene vrednosti su: 18.4924, 32.2486, 99.2932, 150.8627

**Primer 3:**  $\sigma_0 = 0, \sigma_1 = 0, \rho_0 = 0, \rho_1 = 12$



Slika 3.

Za ovakav izbor parametara ne postoji rešenje jednačine (4) u intervalu  $[0, 4\pi]$ , pa nemamo ni sopstvene vrednosti  $0 < \lambda < (4\pi)^2$ .

Do sada su tražene samo pozitivne sopstvene vrednosti jer je  $\lambda = \mu^2$ . Proverimo da li nula može biti sopstvena vrednost problema (1), (2). Iz (1) se vidi da je za  $\lambda = 0$  funkcija  $u$  oblika  $u = ax + b$ , gde su  $a$  i  $b$  konstante. Stavljajući  $u$  u uslove (2) dobijamo

$$u'(0) = a = \sigma_0 b - \rho_0(a + b),$$



$$u'(1) = a = -\sigma_1(a + b) + \rho_1 b,$$

odakle je

$$(5) \quad a(\sigma_1 - \rho_0) + b(\sigma_0 - \rho_0 + \sigma_1 - \rho_1) = 0.$$

Dakle,  $\lambda = 0$  jeste sopstvena vrednost ukoliko je ispunjen uslov (5).

Najzad, ako je  $\lambda = -\mu^2 < 0$ , opšte rešenje jednačine ima oblik

$$u = C_1 \sinh(\mu x) + C_2 \cosh(\mu x) = D_1 \sinh(\mu x + \gamma),$$

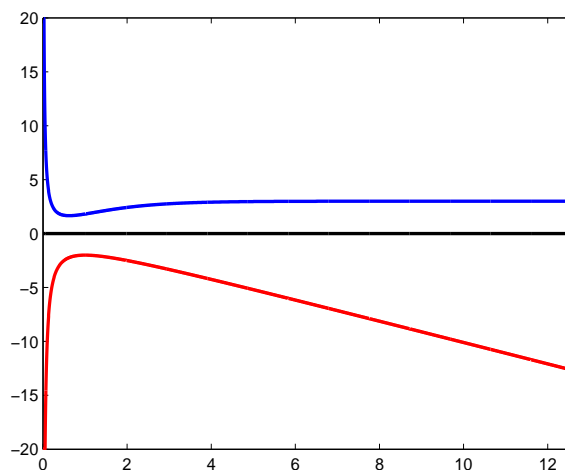
pa je sopstvena funkcija oblika  $u = \sinh(\mu x + \gamma)$ . Nakon sličnog izvođenja sa hiperboličkim trigonometrijskim funkcijama dobijamo sledeće:

$$(6) \quad \tanh \gamma = \frac{\mu + \rho_0 \sinh \mu}{\sigma_0 - \rho_0 \cosh \mu},$$

$$(7) \quad (\sigma_0 + \sigma_1) \cosh \mu - \frac{\rho_0 + \rho_1}{\sinh \mu} = -\mu - \frac{\sigma_0 \sigma_1 - \rho_0 \rho_1}{\mu}.$$

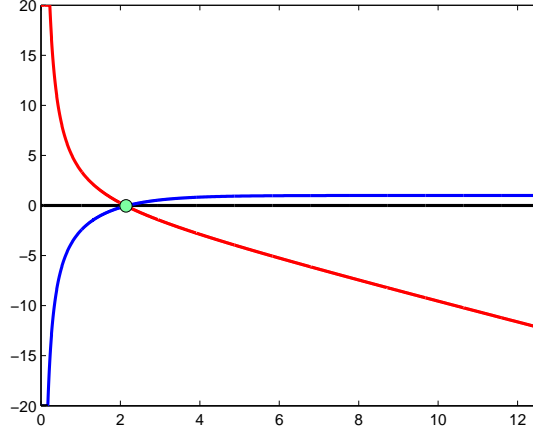
Ostaje još da u postojećem programu umesto jednačine (4) stavimo jednačinu (7) i testiramo program na sledećim primerima.

**Primer 4:** Za parametre kao iz primera 1, tj.  $\sigma_0 = 1, \sigma_1 = 2, \rho_0 = 0.5, \rho_1 = 2$ , nemamo negativne sopstvene vrednosti. Presek funkcija sa leve i desne strane jednačine (7) je prazan.



Slika 4.

**Primer 5:** Za parametre kao iz primera 2, tj.  $\sigma_0 = 0, \sigma_1 = 1, \rho_0 = 1.5, \rho_1 = 3$ , dobijamo jednu negativnu sopstvenu vrednost. Apscisa presečne tačke je 2.1422 pa je tražena sopstvena vrednost  $-(2.1422)^2 = -4.5892$



Slika 5.

Ovaj granični zadatak specifičan je po tome što operator  $L$ , koji odgovara zadatku (1), (2), u opštem slučaju, nije ni samokonjugovan ni nenegativan. U Dirihleovom, Nojmanovom i Robinovom graničnom zadatku ova dva uslova su bila ispunjena, što je garantovalo da su sve sopstvene vrednosti realne i nenegativne. Ispitajmo pod kojim uslovima će operator  $L$  ispunjavati ove uslove.

$$\begin{aligned}
 (Lu, v) &= \int_0^1 -u''v \, dx = -u'v \Big|_0^1 + \int_0^1 u'v' \, dx = \\
 &= -u'(1)v(1) + u'(0)v(0) + \int_0^1 u'v' \, dx = \\
 &= [\sigma_1 u(1) - \rho_1 u(0)]v(1) + [\sigma_0 u(0) - \rho_0 u(1)]v(0) + \int_0^1 u'v' \, dx = \\
 &= \int_0^1 u'v' \, dx + \sigma_0 u(0)v(0) + \sigma_1 u(1)v(1) - \rho_0 u(1)v(0) - \rho_1 u(0)v(1).
 \end{aligned}$$

Slično,

$$(u, Lv) = (Lv, u) = \int_0^1 u'v' \, dx + \sigma_0 u(0)v(0) + \sigma_1 u(1)v(1) - \rho_0 u(0)v(1) - \rho_1 u(1)v(0).$$

Oдавде sledi da je

$$(Lu, v) - (u, Lv) = (\rho_1 - \rho_0)[u(1)v(0) - u(0)v(1)],$$

pa zaključujemo da je operator samokonjugovan za  $\rho_1 = \rho_0$ . Dalje, za  $v = u$  imamo

$$(Lu, u) = \int_0^1 (u')^2 \, dx + \sigma_0 u^2(0) + \sigma_1 u^2(1) - (\rho_0 + \rho_1)u(0)u(1).$$

Kako je integral nenegativne funkcije nenegativan, poslednji izraz biće tim pre nenegativan ako je i

$$\sigma_0 u^2(0) + \sigma_1 u^2(1) - (\rho_0 + \rho_1)u(0)u(1) \geq 0,$$

odakle je,

$$(\rho_0 + \rho_1)^2 - 4\sigma_0\sigma_1 \leq 0.$$

**Napomena:** U vezi sa ovim, primetimo da je u primeru 4 zaista ispunjen uslov za pozitivnu definitnost  $((0.5 + 2)^2 - 4 \cdot 1 \cdot 2 \leq 0)$ , pa ne postoje negativne sopstvene vrednosti.

## Literatura

- [1] А.В. Гулин, Н.И. Ионкин, В.А. Морозова:  
*Устойчивость нелокальных разностных схем, Издательство ЛКИ, Москва 2008.*
- [2] Boško Jovanović, Desanka Radunović:  
*Numerička analiza. Matematički fakultet, Beograd 2003.*
- [3] Julka Knežević-Miljanović, Svetlana Janković, Jelena Manojlović, Vladimir Jovanović:  
*Parcijalne diferencijalne jednačine. Univerzitetska štampa, Beograd 2000.*
- [4] Aleksandra M. Delić, Boško S. Jovanović, Zorica D. Milovanović:  
*About a Transmission Eigenvalue Problem in Disjoint Domains. Comput. Methods Appl. Math. 11, No 4 (2011) (407-417)*
- [5] Radoje Šćepanović, Julka Knežević-Miljanović, Ljubomir Protić:  
*Diferencijalne jednačine. Matematički fakultet, Beograd 2005.*
- [6] Miloš Arsenović, Milutin Dostanić, Danko Jocić:  
*Teorija mere, funkcionalna analiza, teorija operatora. Matematički fakultet, Beograd 1998.*



# DVOMREŽNA METODA KONAČNIH ELEMENATA

Vanja Nikolić

Univerzitet u Beogradu, Fakultet organizacionih nauka  
Jove Ilića 154, 11 000 Beograd, Srbija  
e-mail: nikolicv@fon.bg.ac.rs

**Apstrakt.** Dvomrežna metoda konačnih elemenata zasniva se na uvođenju dva prostora konačnih elemenata različitih dimenzija kojima odgovaraju finija i grublja triangulacija datog domena. Ovu metodu diskretizacije prvi put je primenio Xu u [5] za rešavanje nelinearnih i nesimetričnih problema eliptičkog tipa. Primenjujući dvomrežni algoritam polazni komplikovan problem (nerazdvojen sistem parcijalnih diferencijalnih jednačina, nelinearan problem, nesimetričan problem, problem sopstvenih vrednosti, itd.) rešavamo samo na grubljoj mreži, dok na finijoj mreži rešavamo znatno jednostavniji problem (razdvojen sistem parcijalnih jednačina, linearan problem, simetričan problem, Puasonovu jednačinu, itd.) pri čemu uz odgovarajući izbor grublje mreže rešenje zadržava asimptotski optimalnu tačnost. U radu je razmatrana primena metode dve mreže na problem rešavanja nerazdvojenog sistema linearnih parcijalnih diferencijalnih jednačina (kao modelni zadatak posmatran je granični problem Šredingerovog tipa) i na problem sopstvenih vrednosti za linearne parcijalne diferencijalne operatore. U oba slučaja izloženi su odgovarajući dvomrežni algoritmi i date ocene greške. Numeričkim eksperimentima demonstrirana je efikasnost dvomrežne metode.

**Ključne reči:** metoda konačnih elemenata, dve mreže.

## 1. Uvod

U radu su prikazane tehnike diskretizacije metodom konačnih elemenata koje se baziraju na dva prostora konačnih elemenata. Dvomrežnu metodu diskretizacije prvi put je primenio Xu u [5] za rešavanje nelinearnih i nesimetričnih problema eliptičkog tipa. Kasnije je ona primenjivana na brojnim problemima, na primer na nelinearne jednačine parabolikog tipa u [1], na Navijer-Stoksove jednačine u [4], na evolucione jednačine u [3], itd.

Osnovna ideja metode jeste da komplikovan problem (nerazdvojen sistem parcijalnih diferencijalnih jednačina, problem sopstvenih vrednosti, nelinearan problem itd.) rešimo na grubljoj mreži sa korakom  $H$ , a da jednostavniji problem (Puasonova jednačina, razdvojen sistem parcijalnih diferencijalnih jednačina, linearan problem itd.) kao korekciju rešimo na finijoj mreži sa korakom  $h \ll H$ . Pokazuje se da se uz odgovarajući izbor grublje mreže postiže asimptotski optimalna tačnost.

U radu je razmatrana primena ove metode na dva problema: problem razdvajanja sistema parcijalnih jednačina na primeru graničnog problema Šredingerovog tipa i Dirihleov problem sopstvenih vrednosti. Numeričkim primerima ilustrovana je efikasnost metode.

## 2. Granični problem Šredingerovog tipa

Razmotrimo sledeći granični problem Šredingerovog tipa:

$$-\Delta\psi(x) + V(x)\psi(x) = f(x), \quad x \in \Omega, \quad (1)$$

$$\psi(x) = 0, \quad x \in \partial\Omega. \quad (2)$$

gde je  $\Omega \subset \mathbb{R}^2$  poligonalan, konveksan skup. Funkcija potencijala  $V$ , kao i funkcije  $\psi$  i  $f$  su kompleksne, pa se mogu prikazati kao

$$V(x) = V_1(x) + iV_2(x),$$

$$f(x) = f_1(x) + if_2(x),$$

$$\psi(x) = \psi_1(x) + i\psi_2(x),$$

gde su  $V_1, V_2, f_1$  i  $f_2$  realne funkcije. Postavljeni problem je ekvivalentan sledećem:

$$-\Delta\psi_1(x) + V_1(x)\psi_1(x) - V_2(x)\psi_2(x) = f_1(x), \quad x \in \Omega \quad (3)$$

$$-\Delta\psi_2(x) + V_1(x)\psi_2(x) + V_2(x)\psi_1(x) = f_2(x), \quad x \in \Omega \quad (4)$$

$$\psi_1(x) = 0, \quad \psi_2(x) = 0, \quad x \in \partial\Omega. \quad (5)$$

Uvedimo sledeće vektorske funkcije:

$$\boldsymbol{\psi}(x) = (\psi_1(x), \psi_2(x)),$$

$$\mathbf{V}(x) = (V_1(x), V_2(x)),$$

$$\mathbf{f}(x) = (f_1(x), f_2(x)).$$

Za proizvoljnu vektorsku funkciju  $\boldsymbol{\omega}(x) = (\omega_1(x), \omega_2(x))$  neka je

$$\|\boldsymbol{\omega}\|_{H^1 \times H^1} = \sqrt{\|\omega_1\|_{H^1(\Omega)}^2 + \|\omega_2\|_{H^1(\Omega)}^2}.$$

Uvedimo oznaku  $\leq_c$ , pri čemu  $x \leq_c y$  znači  $x \leq Cy$  za neku pozitivnu konstantu  $C$ , nezavisnu od parametra diskretizacije.

Ako uvedemo oznake

$$l(\boldsymbol{\omega}) = \int_{\Omega} f_1\omega_1 dx + \int_{\Omega} f_2\omega_2 dx,$$

$$a_1(\boldsymbol{\psi}, \boldsymbol{\omega}) = \int_{\Omega} \nabla\psi_1 \cdot \nabla\omega_1 dx + \int_{\Omega} \nabla\psi_2 \cdot \nabla\omega_2 dx,$$

$$R(\boldsymbol{\psi}, \boldsymbol{\omega}) = \int_{\Omega} (V_1\psi_1\omega_1 - V_2\psi_2\omega_1 + V_1\psi_2\omega_2 + V_2\psi_1\omega_2) dx,$$

$$a(\boldsymbol{\psi}, \boldsymbol{\omega}) = a_1(\boldsymbol{\psi}, \boldsymbol{\omega}) + R(\boldsymbol{\psi}, \boldsymbol{\omega}),$$

slaba forma postavljenog problema glasi:

$$\text{Naći } \boldsymbol{\psi} \in H_0^1(\Omega) \times H_0^1(\Omega) \text{ tako da je } a(\boldsymbol{\psi}, \boldsymbol{\omega}) = l(\boldsymbol{\omega}), \quad \forall \boldsymbol{\omega} \in H_0^1(\Omega) \times H_0^1(\Omega). \quad (6)$$

Važi sledeća teorema (videti Teoremu 1 u [2]):

**Teorema 1.** Neka  $\mathbf{f} \in L_2(\Omega) \times L_2(\Omega)$ ,  $\mathbf{V} \in L_{\infty}(\Omega) \times L_{\infty}(\Omega)$ ,  $V_1(x) \geq 0$ ,  $x \in \Omega$ . Tada problem (6) ima jedinstveno rešenje  $\boldsymbol{\psi} \in H^2(\Omega) \times H^2(\Omega)$ , pri čemu važi:

$$\|\boldsymbol{\psi}\|_{H^2 \times H^2} \leq_c \|\mathbf{f}\|_{L_2 \times L_2}.$$

Aproksimirajmo polazni problem metodom konačnih elemenata. Neka je  $T^h = \{K\}$  uniformna triangulacija domena  $\Omega$  i  $S^h \subset H_0^1(\Omega)$  prostor konačnih elemenata

$$S^h = \{\omega_h \in H_0^1(\Omega) \mid \omega_h|_K \in P_K^1, \forall K \in T^h\},$$

gde je  $P_K^1$  prostor polinoma stepena ne većeg od 1.

Odabirom konačnodimenzionalnog potprostora dolazimo do aproksimativnog varijacionog zadatka:

$$\text{Naći } \boldsymbol{\psi}_h \in S^h \times S^h \text{ tako da je } a(\boldsymbol{\psi}_h, \boldsymbol{\omega}_h) = l(\boldsymbol{\omega}_h), \quad \forall \boldsymbol{\omega}_h \in S^h \times S^h.$$

Naredna teorema (videti Teoremu 2 u [2]) daje ocenu greške aproksimacije, tj. ocenu razlike  $\boldsymbol{\psi} - \boldsymbol{\psi}_h$  u različitim normama.

**Teorema 2.** Neka  $\mathbf{f} \in L_2(\Omega) \times L_2(\Omega)$ ,  $\mathbf{V} \in L_{\infty}(\Omega) \times L_{\infty}(\Omega)$ ,  $V_1(x) \geq 0$ ,  $x \in \Omega$ . Tada važi:

$$\|\boldsymbol{\psi} - \boldsymbol{\psi}_h\|_{L_2 \times L_2} \leq_c h^2 \|\boldsymbol{\psi}\|_{H^2 \times H^2},$$

$$\|\boldsymbol{\psi} - \boldsymbol{\psi}_h\|_{H^1 \times H^1} \leq_c h \|\boldsymbol{\psi}\|_{H^2 \times H^2}.$$

## 2.1. Dvomrežna diskretizacija

Dvomrežna diskretizacija se zasniva na uvođenju dva prostora konačnih elemenata različitih dimenzija kojima odgovaraju finija i grublja triangulacija domena.

Uvedimo, dakle, novi prostor konačnih elemenata  $S^H (\subset S^h \subset H_0^1(\Omega))$ , definisan na grubljoj triangulaciji  $T^H$  domena  $\Omega$  sa korakom  $H > h$ . Polazni problem ćemo sada rešavati samo na grubljoj mreži, dok ćemo na finijoj mreži rešavati značajno lakši problem.

### ALGORITAM 1

**Korak 1.** Naći  $\psi_H \in S^H \times S^H$  tako da je  $a(\psi_H, \omega_H) = l(\omega_H)$ ,  $\forall \omega_H \in S^H \times S^H$ .

**Korak 2.** Naći  $\psi_h^* \in S^h \times S^h$  tako da je  $a_1(\psi_h^*, \omega_h) = l(\omega_h) - R(\psi_H, \omega_h)$ ,  $\forall \omega_h \in S^h \times S^h$ .

Primetimo da je sistem parcijalnih jednačina koji se javlja u drugom koraku algoritma razdvojen, uključuje dve Puasonove jednačine, a nerazdvojeni sistem rešavamo samo na grubljoj mreži u prvom koraku.

Pokazuje se da rešenje  $\psi_h^*$  dobijeno ovim algoritmom postiže asimptotski optimalnu tačnost (videti Teoremu 3 u [2]):

**Teorema 3.** Neka  $f \in L_2(\Omega) \times L_2(\Omega)$ ,  $V \in L_\infty(\Omega) \times L_\infty(\Omega)$ ,  $V_1(x) \geq 0$ ,  $x \in \Omega$ . Tada važi:

$$\begin{aligned} \|\psi_h - \psi_h^*\|_{H^1 \times H^1} &\leq_c H^2, \\ \|\psi - \psi_h^*\|_{H^1 \times H^1} &\leq_c h + H^2. \end{aligned}$$

Prema prethodnoj teoremi dovoljno je uzeti  $H = \sqrt{h}$  da bi  $\psi_h^*$  bilo iste tačnosti kao i  $\psi_h$  u  $H^1 \times H^1$  normi. Time je dimenzija prostora  $S^H$  značajno manja od dimenzije prostora  $S^h$ , pa najveći deo posla u Algoritmu 1 čini rešavanje dve razdvojene Puasonove jednačine u drugom koraku.

## 2.2. Numerički primer 1

Testirajmo algoritam 1 na sledećem primeru:

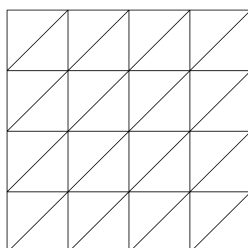
$$-\Delta\psi(\mathbf{x}) + V(\mathbf{x})\psi(\mathbf{x}) = f(\mathbf{x}), \quad \mathbf{x} \in \Omega, \quad (7)$$

$$\psi(\mathbf{x}) = 0, \quad \mathbf{x} \in \partial\Omega, \quad (8)$$

pri čemu je  $\Omega = (0, 1) \times (0, 1)$ ,  $V(\mathbf{x}) = 1 + i$ .

Funkcija  $f(\mathbf{x}) = ((\pi^2 + \frac{3}{2}) + (-2\pi^2 - \frac{1}{2}) \cdot i) \sin(\pi x) \sin(\pi y)$  je izabrana tako da je  $\psi(\mathbf{x}) = (0.5 - i) \sin(\pi x) \sin(\pi y)$  tačno rešenje problema.

Neka je  $T^h = \{K\}$  triangulacija domena dobijena njegovom podelom na manje kvadrate ivice  $h$  i potom podelom svakog kvadrata dijagonalom na dva trougla (slika (1)).

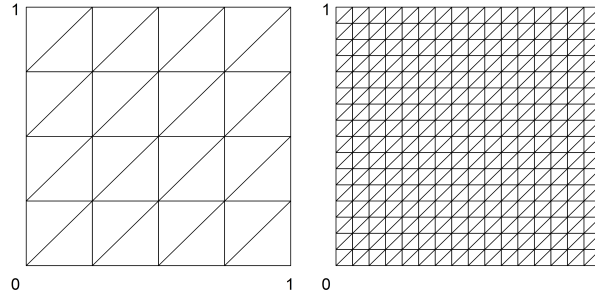


Slika 1. Triangulacija domena.

Uvedimo odgovarajući prostor konačnih elemenata na sledeći način:

$$S^h(\Omega) = \{\omega \in H_0^1(\Omega) : \omega|_K \text{ je linearna, } \forall K \in T^h\}.$$

Primenimo prvo metodu konačnih elemenata koristeći mrežu sa korakom  $h = \frac{1}{2^i}$ , ( $i = 3, 4, 5, 6$ ).



**Slika 2.** Dve mreže sa koracima  $H = \frac{1}{4}$  i  $h = \frac{1}{16}$ .

Dobijeni rezultati prikazani su u tabeli (1). Sva izračunavanja vršena su korišćenjem programa MATLAB 7. Vidimo da je

$$\begin{aligned} \|\psi - \psi_h\|_{L_2 \times L_2} &\approx O(h^2), \\ \|\psi - \psi_h\|_{H^1 \times H^1} &\approx O(h), \end{aligned}$$

što je u skladu sa teorijskim rezultatima iz teoreme (2).

Primenimo sada dvomrežni algoritam koristeći mreže sa korakom  $h = \frac{1}{2^i}$ , ( $i = 4, 6, 8$ ) i odgovarajuće grublje mreže sa korakom  $H = \sqrt{h}$  (slika (2)). Dobijeni rezultati prikazani su u tabeli (2).

**Tabela 1.** Metoda konačnih elemenata

$h$	$\ \psi - \psi_h\ _{L_2 \times L_2}$	$\ \psi - \psi_h\ _{H^1 \times H^1}$
$\frac{1}{8}$	7.6794D-02	9.3073D-01
$\frac{1}{16}$	2.4704D-02	3.9428D-01
$\frac{1}{32}$	7.8157D-03	9.2366D-02
$\frac{1}{64}$	9.5429D-04	7.8007D-02

**Tabela 2.** Dvomrežna metoda

$(H, h)$	$\ \psi - \psi_h^*\ _{H^1 \times H^1}$
$(\frac{1}{4}, \frac{1}{16})$	1.9553D-01
$(\frac{1}{8}, \frac{1}{64})$	9.6071D-02
$(\frac{1}{16}, \frac{1}{256})$	2.4717D-02

Vidimo da je

$$\|\psi - \psi_h^*\|_{H^1 \times H^1} \approx O(H^2),$$

što je u skladu sa teorijskim rezultatima iz teoreme (3).



### 3. Problem sopstvenih vrednosti

Posmatrajmo sledeći Dirihleov granični problem:

$$Lu \equiv - \sum_{i,j=1}^n \frac{\partial}{\partial x_i} (a_{ij} \frac{\partial u}{\partial x_j}) = \lambda u, \quad (9)$$

$$u(x) = 0, \quad x \in \partial\Omega, \quad (10)$$

u ograničenoj, konveksnoj i poligonalnoj oblasti  $\Omega \subset \mathbb{R}^n$ . Pretpostavićemo da su ispunjeni uslovi:

$$\begin{aligned} a_{ij}(x) &\in L_\infty(\Omega), \\ a_{ij}(x) &= a_{ji}(x) \in \mathbb{R}, \text{ s.s. u } \Omega, \\ \sum_{i,j=1}^n a_{ij}(x) \xi_i \xi_j &\geq c_0 \sum_{i,j=1}^n |\xi_i|^2, \quad \forall \xi, \text{ s.s. u } \Omega, \quad c_0 > 0. \end{aligned}$$

**Definicija 1.** Broj  $\lambda$  za koji postoji netrivialno rešenje graničnog problema (9), (10) naziva se sopstvenom vrednošću zadatka (9), (10). Odgovarajuće rešenje  $u(x)$  naziva se sopstvenom funkcijom problema (9), (10).

Ako uvedemo oznake:

$$\begin{aligned} a(u, \omega) &= \int_{\Omega} \sum_{i,j=1}^n a_{ij}(x) \frac{\partial u}{\partial x_j} \frac{\partial \omega}{\partial x_i} dx, \\ b(u, \omega) &= \int_{\Omega} u(x) \omega(x) dx, \end{aligned}$$

varijacioni zadatak glasi:

Naći  $u \in H_0^1(\Omega)$ ,  $u \neq 0$ ,  $\lambda \in \mathbb{R}$  tako da je

$$a(u, \omega) = \lambda b(u, \omega), \quad \forall \omega \in H_0^1(\Omega). \quad (11)$$

Poznato je da problem (11) ima prebrojivo mnogo sopstvenih vrednosti koje su sve pozitivne

$$0 < \lambda_1 \leq \lambda_2 \leq \lambda_3 \leq \dots$$

Za odgovarajuće sopstvene funkcije  $u_1, u_2, u_3, \dots$  možemo pretpostaviti da je

$$a(u_i, u_j) = \lambda_j b(u_i, u_j) = \delta_{ij}.$$

Lako se proverava da je forma  $a(\cdot, \cdot)$  bilinearna, simetrična, ograničena i koercivna. U daljem radu koristićemo energetska normu

$$\|\cdot\|_a = \sqrt{a(\cdot, \cdot)}$$

koja je ekvivalentna normi  $\|\cdot\|_{H^1(\Omega)}$  na  $H_0^1(\Omega)$ .

Neka je  $T^h = \{K\}$  uniformna triangulacija domena  $\Omega$ . Uvedimo kao i ranije prostor konačnih elemenata:

$$S^h(\Omega) = \{\omega \in H_0^1(\Omega) : \omega|_K \in P_K^1, \forall K \in T^h\},$$

gde je  $P_K^1$  prostor polinoma stepena ne većeg od 1. Aproksimacija problema konačnim elementima glasi: Naći  $(\lambda_h, u_h)$ ,  $u_h \in S^h(\Omega)$ ,  $u_h \neq 0$  tako da je

$$a(u_h, \omega) = \lambda_h b(u_h, \omega), \quad \forall \omega \in S^h(\Omega). \quad (12)$$

Ova diskretna aproksimacija problema ima konačno mnogo sopstvenih vrednosti:

$$0 < \lambda_{1,h} \leq \lambda_{2,h} \leq \lambda_{3,h} \leq \dots \leq \lambda_{n_h,h}, \quad n_h = \dim S^h$$

i odgovarajućih sopstvenih funkcija  $u_{1,h}, u_{2,h}, u_{3,h}, \dots, u_{n_h,h}$ , pri čemu važi

$$a(u_{i,h}, u_{j,h}) = \lambda_{j,h} b(u_{i,h}, u_{j,h}) = \delta_{ij}.$$

Sopstvene vrednosti problema (11) zadovoljavaju min-max princip (videti [7]), pa odatle direktno sledi da je

$$\lambda_i \leq \lambda_{i,h}, \quad i = 1, 2, \dots, n_h.$$

Naredna teorema daje ocenu greške ovakve aproksimacije (videti Stav 3.1 u [6]):

**Teorema 4.** Za svako rešenje  $u_{i,h}$ ,  $i = 1, 2, \dots, n$ , problema (11) postoji sopstvena funkcija  $u_i$  problema (12) koja odgovara  $\lambda_i$  takva da je  $\|u_i\|_a = 1$  i

$$\|u_i - u_{i,h}\|_a \leq c h.$$

Za odgovarajuće sopstvene vrednosti važi

$$\lambda_{i,h} - \lambda_i \leq C_i h^2,$$

gde je  $C_i$  pozitivna konstanta koja ne zavisi od koraka  $h$ .

### 3.1. Dvomrežna diskretizacija

Razmotrimo sada dvomrežnu diskretizaciju za problem sopstvenih vrednosti. Njom se rešavanje problema sopstvenih vrednosti na finijoj mreži svodi na rešavanje problema sopstvenih vrednosti na znatno grubljoj mreži i rešavanje linearnog algebarskog sistema na finijoj mreži.

Neka su  $S^H(\Omega)$  i  $S^h(\Omega)$ ,  $S^H(\Omega) \subset S^h(\Omega) \subset H_0^1(\Omega)$ , dva prostora konačnih elemenata kojima odgovaraju redom grublja mreža  $T^H$  i finija  $T^h$ .

#### ALGORITAM 2

**Korak 1.** Naći  $\lambda_{i,H} \in \mathbb{R}$  i  $u_{i,H} \in S^H(\Omega)$  ( $i = 1, 2, \dots, n_H$ ) tako da je  $\|u_{i,H}\|_a = 1$  i

$$a(u_{i,H}, \omega) = \lambda_{i,H} b(u_{i,H}, \omega), \quad \forall \omega \in S^H(\Omega).$$

**Korak 2.** Naći  $u_{i,h} \in S^h(\Omega)$  ( $i = 1, 2, \dots, n_H$ ) tako da je

$$a(u_{i,h}, \omega) = \lambda_{i,H} b(u_{i,H}, \omega), \quad \forall \omega \in S^h(\Omega).$$

**Korak 3.**

$$\lambda_{i,h} = \frac{a(u_{i,h}, u_{i,h})}{b(u_{i,h}, u_{i,h})}, \quad i = 1, 2, \dots, n_H.$$

Vidimo da se problem sopstvenih vrednosti rešava samo u prvom koraku algoritma na grubljoj mreži. U drugom koraku algoritma, na finijoj mreži, rešavamo jednostavan linearan problem i na kraju u trećem koraku računamo Rayleigh-ev količnik. Sledeća teorema daje ocenu greške:

**Teorema 5.** Pretpostavimo da su  $\lambda_{i,h}$  i  $u_{i,h}$  ( $i = 1, 2, \dots, n_H$ ) dobijeni pomoću algoritma 2. Ako je  $S^H \subset S^h$ , tada je

$$\begin{aligned} \|u_i - u_{i,h}\|_a &\leq c H^2 + h, \\ |\lambda_{i,h} - \lambda_i| &\leq c H^4 + h^2. \end{aligned}$$

*Dokaz.* Direktna posledica Teoreme 4.3 i Stava 3.1 u [6]. □

Odavde zaključujemo da je dovoljno uzeti  $H = \sqrt{h}$  da bismo dobili optimalnu grešku. To znači da je dimenzija prostora  $S^H$  značajno manja od dimenzije prostora  $S^h$  i da najveći deo posla u Algoritmu 2 predstavlja rešavanje linearnog eliptičkog problema u drugom koraku.

### 3.2. Numerički primer 2

Ilustriramo izloženi algoritam na sledećem problemu sopstvenih vrednosti:

$$-\Delta u = \lambda u, \quad x \in \Omega, \tag{13}$$

$$u(x) = 0, \quad x \in \partial\Omega, \tag{14}$$

pri čemu je  $\Omega = (0, 1) \times (0, 1)$ .

Sopstvene vrednosti ovog problema su

$$\lambda = (i^2 + j^2)\pi^2,$$

a odgovarajuće sopstvene funkcije

$$u(x) = \sin(i\pi(x-1)) \sin(j\pi(y-1)), \quad i, j = 1, 2, \dots$$

Uvedimo kao i ranije triangulaciju domena  $T^h(\Omega) = \{K\}$  (slika (1)) i prostor konačnih elemenata:

$$S^h(\Omega) = \{\omega \in H_0^1(\Omega) : \omega|_K \text{ je linearna, } \forall K \in T^h\}.$$

Označimo sa  $(\lambda_h, u_h)$  prvu sopstvenu vrednost i odgovarajuću sopstvenu funkciju problema (13), (14) koje su dobijene metodom konačnih elemenata.

Dobijeni podaci dati su u tabeli (3).

Prema teoremi (4) važi:

$$\|\nabla(u - u_h)\|_{L_2} \approx O(h).$$

Vidimo da su rezultati numeričkog eksperimenta prikazani u tabeli (3) saglasni sa teorijskim rezultatima.

Primenimo sada dvomrežni algoritam koristeći mreže sa korakom  $h = \frac{1}{2^i}$ , ( $i = 4, 6, 8$ ) i odgovarajuće grublje mreže sa korakom  $H = \sqrt{h}$ . Označimo sa  $(\lambda^h, u^h)$  prvu sopstvenu vrednost i sopstvenu funkciju problema (13), (14) koje su dobijene korišćenjem dvomrežne metode. Dobijeni rezultati dati su tabeli (4).

**Tabela 3.** Metoda konačnih elemenata

$h$	$\ u - u_h\ _{L_2}$	$\ \nabla(u - u_h)\ _{L_2}$
$\frac{1}{4}$	2.9592D-02	3.6582D-01
$\frac{1}{8}$	7.8655D-03	1.0903D-01
$\frac{1}{16}$	2.0180D-03	2.9097D-02
$\frac{1}{32}$	5.1958D-04	7.4391D-02
$\frac{1}{64}$	1.4283D-04	1.8733D-02
$\frac{1}{128}$	4.9541D-05	8.6940D-03
$\frac{1}{256}$	2.7492D-05	1.1741D-03

**Tabela 4.** Dvomrežna metoda

$(H, h)$	$\ \nabla(u - u^h)\ _{L_2}$
$(\frac{1}{4}, \frac{1}{16})$	6.4821D-02
$(\frac{1}{8}, \frac{1}{64})$	2.9608D-02
$(\frac{1}{16}, \frac{1}{256})$	6.2749D-03

Iz teoreme (5) imamo da je

$$\|\nabla(u - u^h)\|_{L_2} \approx O(H^2).$$

Rezultati prikazani u tabeli (4) saglasni su sa navedenom procenom.

#### Literatura

- [1] **C. N. Dawson, Wheeler, M. F.** Two-grid methods for mixed finite element approximations of nonlinear parabolic equations. *Contemp. Math.* 180, 1994, 191-203.
- [2] **J. Jin, S. Shu, J. Xu.** A two-grid discretization method for decoupling systems of partial differential equations. *Math. Comp.* 75, 2006, 1617-1626.
- [3] **M. Marion, J. Xu.** Error estimates on new nonlinear Galerkin method based on two-grid finite elements. *SIAM J. Numer. Anal.* 32 15, 1995, 1170-1184.
- [4] **T. Utnes.** Two-grid finite element formulations of the incompressible Navier-Stokes equations. *Comm. Numer. Methods Engrg.* 34 15, 1997, 675-684.
- [5] **J. Xu.** A novel two-grid method for semilinear equations. *SIAM J. Sci. Comput.* 15, 1994, 17-25.
- [6] **J. Xu, A. Zhou.** A two-grid discretization scheme for eigenvalue problems. *Math. Comp.* 70(233), 2001, 231-237.
- [7] **Chatelin, F.** Spectral Approximations of Linear Operators. *Academic Press, New York*, 1983.

Received

# RAZVOJ NOVIH MODELA PREDVIĐANJA RETENCIONOG PONAŠANJA U TEČNOJ HROMATOGRAFIJI HIDROFILNIH INTERAKCIJA

**Aleksandar Đenić**

Matematički fakultet, Univerzitet u Beogradu  
e-mail: djenic@matf.bg.ac.rs

**Miroslav Marić**

Matematički fakultet, Univerzitet u Beogradu  
e-mail: maricm@matf.bg.ac.rs

**Tijana Rakić**

Farmaceutski fakultet, Univerzitet u Beogradu  
e-mail: tijana.rakic@pharmacy.bg.ac.rs

**Zorica Stanimirović**

Matematički fakultet, Univerzitet u Beogradu  
e-mail: zoricast@matf.bg.ac.rs

**Biljana Janičić Stojanović**

Farmaceutski fakultet, Univerzitet u Beogradu  
e-mail: jancic.stojanovic@pharmacy.bg.ac.rs

**Apstrakt.** U ovom radu predstavljen je razvoj i optimizacija metode u tečnoj hromatografiji hidrofilnih interakcija (HILIC, eng. *hydrophilic interaction liquid chromatography*) primenom nove tehnike matematičkog modelovanja i nove funkcije hromatografskog odgovora. HILIC je nova separaciona tehnika čij je retencioni mehanizam složen i nedovoljno proučen. Uobičajene tehnike modelovanja hromatografskog odgovora u reverzno-faznom hromatografskom sistemu ne mogu se uvek primeniti na HILIC sistem, pogotovo ako je opseg ispitivanih faktora širok. U ovom istraživanju predložena je nova tehnika modelovanja hromatografskih odgovora putem Njutnovog interpolacionog polinoma sa podeljenim razlikama. Kao funkcija cilja odabrana je složena funkcija hromatografskog odgovora koja je modelovana indirektno, a direktno su modelovani retencioni faktori analiziranih supstanci. Predloženi pristup omogućio je identifikaciju optimalnih uslova razdvajanja. Verifikacija predložene tehnike pokazala je visoko slaganje eksperimentalno i teorijski dobijenih hromatograma. Pokazano je da se Njutnov interpolacioni polinom sa podeljenim razlikama može uspešno koristiti u rešavanju optimizacionih problema u tečnoj hromatografiji hidrofilnih interakcija.

**Ključne reči:** tečna hromatografija, HILIC, interpolacija, podeljene razlike.

## 1. Uvod

Tečna hromatografija hidrofilnih interakcija (HILIC, eng. *Hydrophilic Interaction Liquid Chromatography*) je nova separaciona tehnika koja se zasniva na upotrebi polarnih stacionarnih faza i mobilne faze sa niskim udelom vodene i visokim udelom organske faze [1, 2]. HILIC je pokazao velike prednosti u analizi slabo baznih supstanci u odnosu na ostale tipove hromatografije jer omogućava zadovoljavajuće zadržavanje ovih supstanci na koloni kao i dobar oblik pikova. Kako su većina lekovitih supstanci prisutnih na tržištu slabe baze, interesovanje farmaceutske industrije za razvoj metoda u HILIC sistemu je veliko.

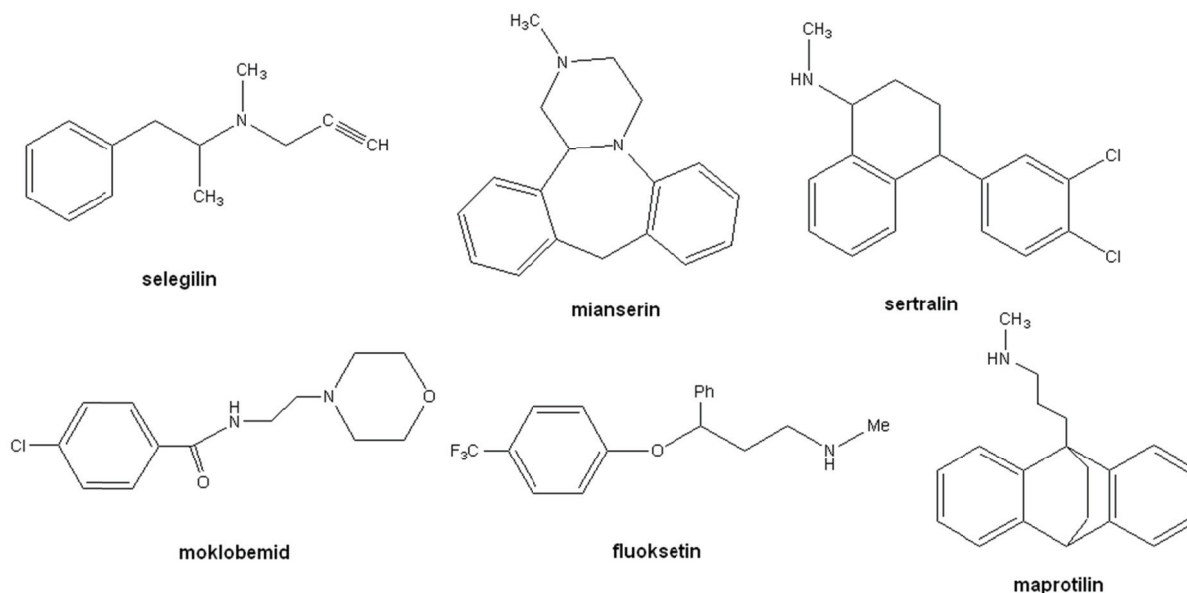
Najefikasniji način za razvoj i optimizaciju hromatografskih metoda predstavlja primena metodologije eksperimentalnog dizajna koja uključuje:

- 1) identifikaciju faktora koji značajno utiču na sistem (*input*)
- 2) definisanje eksperimentalnog plana
- 3) odabir odgovora koji će se pratiti (*output*)
- 4) kreiranje matematičkog modela koji uspostavlja vezu između *input*-a i *output*-a
- 5) definisanje funkcije hromatografskog odgovora
- 6) identifikacija globalnih optimalnih uslova razdvajanja

Ključni koraci u ovoj metodologiji su odabir odgovarajućeg matematičkog modelovanja i adekvatne funkcije hromatografskog odgovora. Hromatografski odgovori najčešće se modeluju primenom kvadratnih funkcija jer se smatra da u uskom intervalu variranja faktora ova funkcija može adekvatno da opiše sistem. Međutim, za razliku od ostalih tipova hromatografije, separacioni mehanizam u HILIC-u je izuzetno složen i nedovoljno proučen pa aproksimacija kvadratnom funkcijom ne uspeva na odgovarajući način da opiše retenciono ponašanje supstanci. Stoga, odabir

odgovarajućeg matematičkog modela za ovaj sistem predstavlja veliki izazov. Nakon uspostavljanja matematičke veze između inputa i outputa, potrebno je definisati funkciju cilja t.j. funkciju hromatografskog odgovora (CRF, eng. *Chromatographic Response Function*). U literaturi do sada postoji veliki broj predloženih CRF, ali one nisu univerzalno primenljive na svaki hromatografski sistem [3-5].

Cilj ovog rada bio je razvoj i optimizacija metode tačne hromatografije hidrofilnih interakcija primenom novih modela retencionog predviđanja i nove funkcije hromatografskog odgovora. Kao eksperimentalna model smeša primenjena je smeša šest antidepresiva (selegilin, mianserin, sertralin, moklobemid, fluoksetin i maprotilin) čija je hemijska struktura prikazana na slici 1.



**Slika 1.** Hemijska struktura analiziranih supstanci

## 2. Eksperimentalni uslovi

### 2.1. Reagensi

Svi upotrebljeni reagensi bili su analitičke čistoće. Mobilna faza i rastvarači pripremljeni su od acetonitrila (*Lab Scan*, Ireland), amonijum-acetata (*J. T. Backer*, The Netherlands), glacijalne sirćetne kiseline (*Zorka Pharma*, Srbija) i vode HPLC stepena čistoće.

### 2.2. Rastvori analiziranih supstanci

Eksperimentalna model smeša pripremljena je rastvaranjem supstanci u smeši acetonitril – vodena faza (40 mM amonijum-acetat, pH 4.0 podešen glacijalnom sirćetnom kiselinom) 90:10 v/v tako da se dobije koncentracija  $400 \mu\text{g mL}^{-1}$  za fluoksetin i maprotilin,  $100 \mu\text{g mL}^{-1}$  za mianserin,  $50 \mu\text{g mL}^{-1}$  za moklobemid,  $600 \mu\text{g mL}^{-1}$  za sertralin i  $1 \text{ mg mL}^{-1}$  za selegilin.

### 2.3. Mobilna faza

Variranja sastava mobilne faze definisana su eksperimentalnim planom datim u Tabeli 1.

### 2.3. Hromatografski uslovi

Eksperimenti su izvedeni na tačnom hromatografu *Finnigan Surveyor Thermo Scientific* koji se sastoji od HPLC pumpe, Autosampler Plus i UV/VIS Plus detektora. Podaci su sakupljeni u softveru ChromQuest. Kao stacionarna faza upotrebljena je BETASIL Silica-100 (100 mm x 4.6 mm, 5  $\mu\text{m}$ ) kolona. Protok mobilne faze bio je  $1 \text{ mL min}^{-1}$ . Temperatura kolone bila je  $30 \text{ }^\circ\text{C}$ , a talasna dužina detekcije 254 nm.

## 2.4. Softver

Za kreiranje matrice eksperimentalnog plana upotrebljena je STATISTICA 7. Za svrhe računanja parametara matematičkog modela i proveru ponašanja različitih uslova razdvajanja razvijena je posebna aplikacija.

## 3. Rezultati i diskusija

### 3.1. Identifikacija faktora koji značajno utiču na sistem

Preliminarnim istraživanjima ustanovljeno je da su faktori sa najvećim uticajem na retenciono ponašanje analiziranih antidepresiva sadržaj acetonitrila u mobilnoj fazi, pH vrednost vodene faze i koncentracija amonijum-acetata u vodenoj fazi. Uticaj ova tri faktora dalje je ispitivan u intervalima: 86 % do 94 % za sadržaj acetonitrila u mobilnoj fazi, 3,0 do 6,0 pH jedinica za pH vrednost vodene faze i 20 do 60 mM<sup>-1</sup> za koncentraciju amonijum-acetata u vodenoj fazi. Ostali hromatografski parametri zadržani su na konstantnom nivou.

### 3.2. Definisanje eksperimentalnog plana

Eksperimentalna matrica definisana je 3<sup>3</sup> punim faktorskim dizajnom gde su odabrana 3 faktora ispitana na 3 nivoa u okviru 27 eksperimenata. Nivoi faktora su kodirani i šema eksperimenata prikazana je u tabeli 1.

**Tabela 1.** 3<sup>3</sup> eksperimentalna matrica i dobijeni hromatografski odgovori

Run	x	y	z	k <sub>2</sub>	k <sub>3</sub>	k <sub>4</sub>	k <sub>5</sub>	k <sub>6</sub>
1	-1*	-1	-1	2,02	2,02	2,09	1,89	2,11
2	-1	-1	0	1,61	1,59	1,66	1,49	1,69
3	-1	-1	1	1,36	1,28	1,37	1,24	1,41
4	-1	0	-1	0,62	2,13	2,54	2,54	2,86
5	-1	0	0	0,46	1,41	1,63	1,62	1,86
6	-1	0	1	0,44	1,28	1,46	1,42	1,65
7	-1	1	-1	0,36	2,09	2,94	3,51	4,13
8	-1	1	0	0,21	1,29	1,79	2,02	2,34
9	-1	1	1	0,14	1,01	1,39	1,59	1,9
10	0	-1	-1	3,44	4,41	4,63	4,35	4,88
11	0	-1	0	2,95	3,62	3,76	3,54	4,03
12	0	-1	1	2,51	3,02	3,09	2,93	3,36
13	0	0	-1	0,82	3,87	5,25	6,19	7,31
14	0	0	0	0,78	3,15	3,97	4,41	5,04
15	0	0	1	0,47	2,2	2,74	3,17	3,77
16	0	1	-1	0,64	3,52	5,51	7,81	9,69
17	0	1	0	0,42	2,3	3,52	4,8	5,92
18	0	1	1	0,33	1,9	2,87	3,96	4,94
19	1	-1	-1	5,09	12,31	13,34	13,74	16,07
20	1	-1	0	3,86	8,81	8,95	9,53	11,28
21	1	-1	1	3,47	7,26	7,29	7,77	9,27
22	1	0	-1	1,59	7,97	13,99	23,44	31,42
23	1	0	0	1,8	6,27	9,32	14,63	19,57
24	1	0	1	0,99	5,14	7,86	12,78	16,93
25	1	1	-1	1,37	6,71	12,99	25,44	37,42
26	1	1	0	1,04	4,99	9,12	17,04	24,21
27	1	1	1	0,95	4,54	8,09	15,36	22,8

x – sadržaj actonitrila u mobilnoj fazi (%), y - pH vrednost vodene faze, z – koncentracija amonijum-acetata u vodenoj fazi (mM); k<sub>2</sub> – k<sub>6</sub> – retencioni faktori mianserina, sertralina, moklobemida, fluoksetina i mapotilina, redom.

\* nivoi faktora dati su u kodiranim vrednostima

### 3.3. Odabir odgovora sistema

Postoji veliki broj odgovora sistema koji se mogu odabrati kao output. Odgovori poput retencionog vremena i retencionog faktora govore nam samo o položaju pojedinačnih supstanci pri određenim eksperimentalnim uslovima, dok nam odgovori poput faktora rezolucije i faktora selektivnosti daju korisnije podatke o hromatogramu jer ukazuju na kvalitet razdvajanja. Međutim, direktno modelovanje kompleksnih hromatografskih odgovora poput faktora rezolucije ne daje dovoljno tačne modele, pre svega zbog mogućnosti inverzije pikova koja je česta kada se radi u širokom opsegu pH vrednosti mobilne faze. Direktno modelovanje hromatografskih funkcija je još problematičnije, jer je njihovo ponašanje obično kompleksno i uslovljeno istovremenom promenom više odgovora, pa direktni modeli imaju značajne greške [6, 7].

U ovom radu izvršeno je direktno modelovanje jednostavnih hromatografskih odgovora - retencionih faktora, a zatim su naknadno izračunate vrednosti ostalih hromatografskih parametri. Dobijeni rezultati za retencione faktore ispitivanih supstanci prikazani su u tabeli 1. Kako je sertralin pokazao neretenciono ponašanje, u dalje modelovanje su ušli retencioni faktori mianserina, sertralina, fluoksetina, moklobemida i maprotilina.

### 3.4. Kreiranje matematičkog modela

Eksperimentalno su dobijene vrednosti funkcija  $k_2, k_3, k_4, k_5, k_6$  u svakoj od tačaka  $(x, y, z) \in \mathbb{R}^3$ ,  $x \in \{-1, 0, 1\}$ ,  $y \in \{-1, 0, 1\}$ ,  $z \in \{-1, 0, 1\}$ . Na osnovu datih vrednosti, konstruisana je aproksimacija svake od funkcija  $k_2, k_3, k_4, k_5, k_6$  Njutnovim interpolacionim polinomom sa podeljenim razlikama:  $P(x, y, z) \approx L_3(x, y, z)$ ,

gde je:

$$\begin{aligned} L_3(x, y, z) = & a_{000} + a_{100}(x - x_0) + a_{010}(y - y_0) + a_{001}(z - z_0) + a_{110}(x - x_0)(y - y_0) \\ & + a_{101}(x - x_0)(z - z_0) + a_{011}(y - y_0)(z - z_0) + a_{111}(x - x_0)(y - y_0)(z - z_0) \\ & + a_{200}(x - x_0)(x - x_1) + a_{020}(y - y_0)(y - y_1) + a_{002}(z - z_0)(z - z_1) \\ & + a_{210}(x - x_0)(x - x_1)(y - y_0) + a_{201}(x - x_0)(x - x_1)(z - z_0) \\ & + a_{120}(x - x_0)(y - y_0)(y - y_1) + a_{021}(y - y_0)(y - y_1)(z - z_0) \\ & + a_{102}(x - x_0)(z - z_0)(z - z_1) + a_{012}(y - y_0)(z - z_0)(z - z_1), \end{aligned}$$

Pri tome je

$$x_0 = y_0 = z_0 = -1 \text{ i}$$

$$x_1 = y_1 = z_1 = 0.$$

Vrednost  $a_{ijk}$  jednaka je odgovarajućoj podeljenoj razlici funkcije tri promenljive, u oznaci  $f[x_0, x_1, \dots, x_i; y_0, y_1, \dots, y_j; z_0, z_1, \dots, z_k]$ .

Podeljene razlike funkcije 3 promenljive definišemo na sledeći način:

- (1) Vrednost podeljene razlike nultog reda po sve tri promenljive je  $f[x_0; y_0; z_0] = f(x_0, y_0, z_0)$ .
- (2) Vrednost podeljene razlike višeg reda računa se preko podeljenih razlika nižeg reda.



Na primer:

$$f[\mathbf{x}_0, \mathbf{x}_1; y_0, y_1, \dots, y_j; z_0, z_1, \dots, z_k] = \frac{f[\mathbf{x}_1; y_0, y_1, \dots, y_j; z_0, z_1, \dots, z_k] - f[\mathbf{x}_0; y_0, y_1, \dots, y_j; z_0, z_1, \dots, z_k]}{\mathbf{x}_1 - \mathbf{x}_0}$$

$$f[\mathbf{x}_0, \mathbf{x}_1, \mathbf{x}_2; y_0, \dots, y_j; z_0, \dots, z_k] = \frac{f[\mathbf{x}_1, \mathbf{x}_2; y_0, \dots, y_j; z_0, \dots, z_k] - f[\mathbf{x}_0, \mathbf{x}_1; y_0, \dots, y_j; z_0, \dots, z_k]}{\mathbf{x}_2 - \mathbf{x}_0}$$

Na sličan način se računaju podeljene razlike višeg reda svođenjem na podeljene razlike nižeg reda u odnosu na proenljiive y i z.

### 3.5. Definisanje funkcije hromatografskog odgovora

Za procenu kvaliteta dobijenih hromatograma analizirane smeše dizajnirana je nova funkciju hromatografskog odgovora  $CRF_k$  koja ima sledeći oblik:

$$CRF_k = \left[ a \cdot \left( \alpha_{opt} - \frac{\sum_{i=1}^N \frac{k_{i+1}}{k_i}}{N-1} \right) + 1 \right] \cdot \left( \frac{t_f}{t_{opt}} \right)^b$$

Za  $k_{i+1}/k_i > \alpha_{opt}$ ,  $k_{i+1}/k_i = \alpha_{opt}$  i za  $t_f > t_{opt}$ ,  $t_f = t_{opt}$

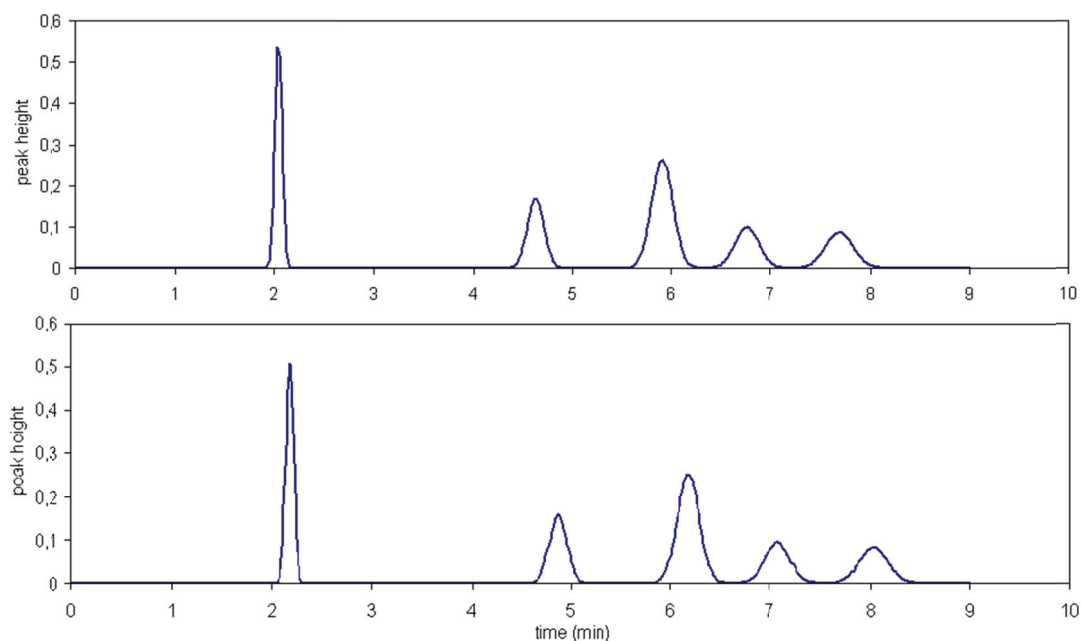
Funkcija se sastoji od člana koji pricenjuje razdvajanje i člana koji procenjuje ukupno vreme trajanja analize. Prednost ovakve funkcije je dobar balans između članova separacije i vremena koji se postiže koeficijentima  $a$  i  $b$  (u ovom slučaju podešeni su na 5 i 1, redom). Kvalitet razdvajanja ocenjuje se kao funkcija retencionih faktora jer smo upravo njih modelovali. Maskiranje loše razdvojenih pikova visokim vrednostima faktora selektivnosti dobro razdvojenih izbegnuto je nivelisanjem svih fakotra selektivnosti većih od odabrane optimalne vrednosti na optimalnu vrednost (u ovom slučaju  $\alpha_{opt}$  bilo je 1,2). Slično, vreme trajanja analiza koje su bila kraće od optimalne vrednosti nivelisano je na optimalnu kako analize sa izrazito kratkim vremenom ne bi maskirale loše razdvajanje (u ovom primeru  $t_{opt}$  bilo je 8 minuta). Funkcija je dizajnirana tako da poboljšanje kvaliteta hromatograma prati smanjenje njene numeričke vrednosti. Najbolja moguća vrednost funkcije koja karakteriše optimalan hromatogram je 1.

### 3.6. Identifikacija globalnih optimalnih uslova razdvajanja

Za identifikaciju globalnih optimalnih uslova primenjena je metoda pretraživanja čvorova mreže (*grid point search*). Intervali ispitivanih faktora su diskretizovani. Gustina mreže definisana je inkrementima 0,25 za faktore x i z i 0,2 za faktor y (u kodiranim vrednostima). Stoga je variranje acetonitrila ispitivano u koracima od 1 %, pH vrednosti vodene faze u koracima od 0,3 pH jedinice i koncentracije amonijum-acetata u vodenom fazi u koracima od 5 mM. Ukupan broj tako dobijenih čvorova iznosio je 891.

Vrednost funkcije hromatografskog odgovora izračunata je u svim tačkama i identifikovana je tačka sa najboljom (najmanjom) vrednošću  $CRF_k$  od 1,04 za koju su vrednosti x, y, z iznosile -1, 1 i -1 redom. Na taj način identifikovan je sledeći optimalni sastav mobilne faze: acetonitril: vodena faza (20 mM amonijum-acetat, pH 6,0 podešen glacijskom sirćetnom kiselinom) 86:14 v/v. Dobijeni uslovi eksperimentalno su verifikovani. Teorijski i eksperimentalno dobijeni hromatogram prikazani su na slici 2.

Dobijeno visoko slaganje između hromatograma potvrdilo je efikasnost nove predložene tehnike matematičkog modelovanja i odabrane funkcije hromatografskog odgovora.



**Slika 2.** Teorijski i eksperimentalno dobijeni hromatogram analizirane smeše, redosled eluiranja: mianserin, sertralin, moklobemid, fuoksetin, maprotilin

#### 4. Zaključak

Ovaj rad predstavlja novi pristup optimizaciji razdvajanja farmaceutski aktivnih supstanci u tečnoj hromatografiji hidrofилnih interakcija. Njutnov interpolacioni polinom sa podeljenim razlikama primenjen je za modelovanje retencionog ponašanja analiziranih supstanci, a nova funkcija hromatografskog odgovora u kombinaciji sa optimizacijom pretraživanjem čvorova mreže omogućila je identifikaciju globalnih optimalnih uslova razdvajanja. Visoko slaganje između teorijskih i eksperimentalno dobijenih hromatograma potvrdilo je da se predloženi pristup može koristiti u analizi sistema tečne hromatografije hidrofилnih interakcija.

#### Literatura

- [1] **P. Hemstrom, K. Irgum.** Hydrophilic interaction chromatography, *Journal of Separation Science*, 2006, 29, 1784 –1821.
- [2] **Y. Hsieh.** Potential of HILIC–MS in quantitative bioanalysis of drugs and drug metabolites, *Journal of Separation Science*, 2008, 31, 1481 – 1491.
- [3] **J.C. Berridge.** Unattended optimization of reversed-phase high-performance liquid chromatographic separations using the modified simplex algorithm, *Journal of Chromatography*, 1982, 244, 1 – 14.
- [4] **V.M. Morris, J.G. Hughes, P.J. Marriott.** Examination of a new chromatographic function, based on an exponential resolution term, for use in optimization strategies: application to capillary gas chromatography separation of phenols, *Journal of Chromatography A*, 1996, 755, 235 – 243.
- [5] **B. Jancic–Stojanovic, T. Rakic, N. Kostic, A. Vemic, A. Malenovic, D. Ivanovic, M. Medenica.** Advancement in optimization tactic achieved by newly developed chromatographic response function: application to LC separation of raloxifene and its impurities, *Talanta*, 85, 2011, 1453 – 1460.
- [6] **W. Dewe, R.D. Marini, P. Chiap, Ph. Hubert, J. Crommen, B. Boulanger.** Development of response models for optimizing HPLC methods, *Chemometrics and Intelligent Laboratory Systems*, 2004, 74, 263 – 268.
- [7] **P. Lebrun, B. Govaerts, B. Debrus, A. Ceccato, G. Caliato, P. Hubert, B. Boulanger.** Development of a new predictive modeling technique to find with confidence equivalence zone and design space of chromatographic analytical methods, *Chemometrics and Intelligent Laboratory Systems*, 2008, 91, 4 – 16.

# FUNKCIJE HROMATOGRAFSKOG ODGOVORA U REŠAVANJU OPTIMIZACIONIH PROBLEMA U TEČNOJ HROMATOGRAFIJI HIDROFILNIH INTERAKCIJA

**Marina Pekić**

*Farmaceutski fakultet, Univerzitet u Beogradu,  
pikollo88@yahoo.com*

**Tijana Rakić**

*Katedra za analitiku lekova, Farmaceutski fakultet,  
Univerzitet u Beogradu, tijana.rakic@pharmacy.bg.ac.rs*

**Sanja Ribić**

*Farmaceutski fakultet, Univerzitet u Beogradu,  
sanja\_r\_88@yahoo.com*

**Biljana Jančić Stojanović**

*Katedra za analitiku lekova, Farmaceutski fakultet,  
Univerzitet u Beogradu,  
jancic.stojanovic@pharmacy.bg.ac.rs*

## Rezime

Funkcije hromatografskog odgovora predstavljaju matematičko rešenje koje omogućava objektivnu procenu kvaliteta hromatograma i dopušta istovremenu optimizaciju više različitih hromatografskih ciljeva. Cilj ovog rada je procena efikasnosti četiri različite funkcije hromatografskog odgovora na simuliranim hromatogramima i eksperimentalno dobijenim hromatogramima nakon analize beta agonista i blokatora tečnom hromatografijom hidrofilnih interakcija (HILIC – eng. *Hydrophilic Interaction Liquid Chromatography*). Pet simuliranih hromatograma kreirano je u *Microsoft Excel* programu. Osamnaest eksperimentalno dobijenih hromatograma generisano je nakon analize pet beta agonista i blokatora pod različitim uslovima u HILIC sistemu (stacionarna faza: Betasil Silica-100 (100 mm x 4,6 mm, 5  $\mu$ m veličine čestica), mobilna faza: acetonitril: vodeni rastvor amonijum-acetata, čiji je pH podešen glacijalnom sirćetnom kiselinom). Sastav mobilne faze variran je prema planu centralnog kompozicionog dizajna. Četiri funkcije hromatografskog odgovora primenjene su u proceni simuliranih hromatograma. Analizirani su različiti separacioni parametri uključeni u funkcije i njihova sposobnost da istovremeno procenjuju kvalitet razdvajanja svih prisutnih pikova. Ispitani su pristupi proceni ukupne dužine trajanja hromatografske analize. Merena je sposobnost funkcija da postave adekvatan balans između procene kvaliteta razdvajanja i ukupne dužine trajanja analize. Funkcije su testirane na eksperimentalno dobijenim hromatogramima. Primećene su značajne razlike u vrednovanju kvaliteta hromatograma od strane različitih funkcija. Pokazano je da funkcije hromatografskog odgovora predstavljaju pouzdan i objektivan pristup u optimizaciji razdvajanja supstanci u HILIC sistemu. Proučene su prednosti i mane različitih funkcija i definisane su mere opreza pri odabiru funkcije izbora za dati optimizacioni problem.

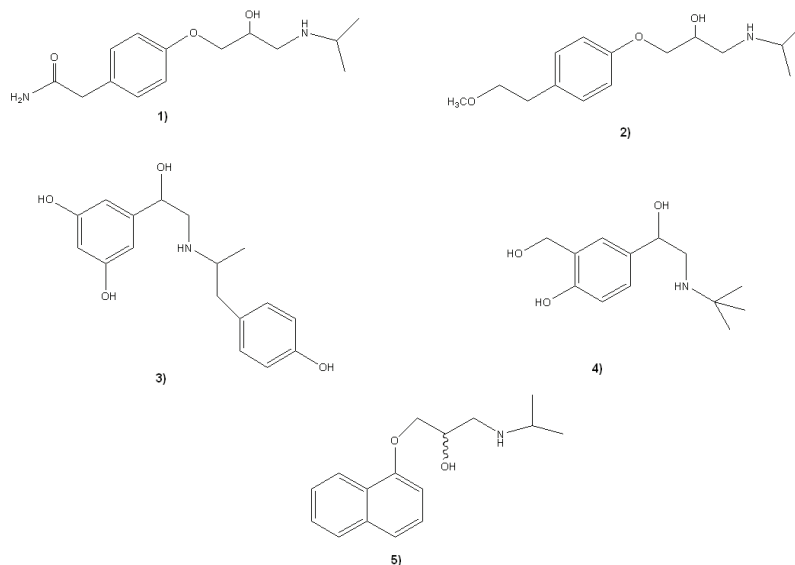
## Ključne reči

Funkcije hromatografskog odgovora, HILIC, beta agonisti i blokatori.

## Uvod

Razvoj i optimizacija metode tačne hromatografije najčešće podrazumeva istovremeno postizanje više ciljeva: najbolje moguće razdvajanje analiziranih supstanci, maksimalno razdvajanje kritičnog para pikova, minimalno vreme trajanja analize, zadovoljavajući oblik pikova ispitivanih supstanci, maksimalnu robusnost postignutog optimuma... Kada je analizirana smeša složena, nemoguće je postići sve ciljeve istovremeno. Zato se postavlja problem rangiranja hromatograma koji su jedan deo postavljenih kriterijuma zadovoljili, a drugi nisu. Funkcije hromatografskog odgovora (CRF, eng. *Chromatographic Response Function*) predstavljaju matematičko rešenje koje omogućava objektivnu procenu kvaliteta hromatograma [1–4]. CRF omogućavaju transformaciju različitih parametara kvaliteta hromatograma u jednu numeričku vrednost na osnovu koje se može izvršiti rangiranje hromatograma. U literaturi postoji veliki broj hromatografskih funkcija, ali nijedna od njih nije univerzalno primenljiva. Cilj ovog rada bio je da se proceni efikasnost četiri različite funkcije hromatografskog odgovora. U ispitivanje su uključene *Berridge*-ova funkcija hromatografskog odgovora ( $B_{CFR}$ ) [1], *Morris*-ova hromatografska eksponencijalna funkcija (CEF) [2], *Duarte*-ova funkcija hromatografskog odgovora ( $D_{CRF}$ ) [3] i nova funkcija hromatografskog

odgovora ( $N_{CRF}$ ) koju su dizajnirali autori ovog rada [4]. Funkcije su testirane prvo na seriji simuliranih hromatograma gde su razmatrane prednosti i mane u matematičkoj formulaciji svake od njih. Zatim su funkcije ispitane na eksperimentalno dobijenim hromatogramima nakon analize smeše agonista i antagonista  $\beta$ -receptora koja je uključila: atenolol, metoprolol, propranolol, fenoterol i salbutamol (Slika 1).



**Slika 1.** Hemijska struktura analiziranih supstanci: 1) atenolol, 2) metoprolol, 3) fenoterol, 4) salbutamol i 5) propranolol

Supstance su analizirane u tečnoj hromatografiji hidrofilnih interakcija (HILIC, eng. *Hydrophilic Interaction Liquid Chromatography*) jer je to relativno nova tehnika i pretragom literature zaključeno je da su funkcije hromatografskog odgovora do sada malo ispitivane u ovom sistemu [5, 6].

## Eksperimentalni uslovi

### Reagensi

Svi upotrebljeni reagensi bili su analitičke čistoće. Mobilna faza i rastvarači pripremljeni su od acetonitrila (*Lab Scan, Ireland*), amonijum-acetata (*Riedel-de Haen, Seelze, Germany*), glacijalne sirćetne kiseline (*Zorka Pharma, Srbija*) i vode HPLC stepena čistoće.

### Rastvori analiziranih supstanci

Eksperimentalna model smeša pripremljena je rastvaranjem supstanci u smeši acetonitril – vodena faza (40 mM amonijum-acetat, pH 4,5 podešen glacijalnom sirćetnom kiselinom) 90:10 v/v tako da se dobiju koncentracije  $50 \mu\text{g mL}^{-1}$  za atenolol, metoprolol, fenoterol i salbutamol i  $20 \mu\text{g mL}^{-1}$  za propranolol.

### Mobilna faza

Sastav mobilne faze je variran u opsegu koji je dat u Tabeli 2.

### Hromatografski uslovi

Eksperimenti su izvedeni na tečnom hromatografu *Waters Breeze System*. Kao stacionarna faza upotrebljena je Betasil Silica-100 (100 mm x 4,6 mm, 5  $\mu\text{m}$ ) kolona. Protok mobilne faze bio je  $1,0 \text{ mL min}^{-1}$ . Temperatura kolone bile je  $30 \text{ }^\circ\text{C}$ , a talasna dužina detekcije 254 nm.

## Softver

Za kreiranje matrice eksperimentalnog plana upotrebljen je *Design Expert*.

## Rezultati i diskusija

Funkcije hromatografskog odgovora koje su analizirane u ovom radu bile su:

1. Beridžova funkcija hromatografskog odgovora ( $B_{CRF}$ ) [1] čija je formula:

$$B_{CRF} = \sum_{i=1}^L R_i + L^{w_1} - w_2 |T_A - T_L| - w_3 (T_1 - T_0)$$

gde je  $R_i$  faktor rezolucije između susednih pikova,  $L$  ukupan broj pikova,  $T_A$  maksimalna prihvatljiva dužina hromatografske analize,  $T_L$  retenciono vreme poslednjeg pika,  $T_1$  retenciono vreme prvog pika,  $T_0$  minimalno prihvatljivo retenciono vreme prvog pika i  $w_1$ ,  $w_2$  i  $w_3$  su težinske konstante koje određuje analitičar (u ovom radu  $w_1 = w_2 = w_3 = 1$ ,  $T_A = 10$  min,  $T_1 = 3$  min).

2. Morisova hromatografska eksponencijalna funkcija (CEF) [2] čija je formula:

$$CEF = \left[ \left( \sum_{i=1}^{n-1} (1 - e^{a(R_{opt} - R_i)})^2 \right) + 1 \right] \left[ 1 + \frac{t_f}{t_{max}} \right]$$

gde je  $n$  broj očekivanih pikova,  $R_{opt}$  prihvatljiv faktor rezolucije,  $R_i$  eksperimentalno dobijen faktor rezolucije,  $t_f$  retenciono vreme poslednjeg pika,  $t_{max}$  maksimalna prihvatljiva dužina hromatografske analize i  $a$  je konstanta koju definiše analitičar. U ovom radu definisane su konstante:  $a = 3$ ,  $R_{opt} = 1,5$ ,  $t_{max} = 10$  min.

3. Duarteova funkcija hromatografskog odgovora ( $D_{CRF}$ ) [3] čija je formula:

$$D_{CRF} = \sum_{i=1}^{N-1} \theta_{s,i} + N - ((t_{R,L} - t_0) / t_{R,L})$$

gde je  $N$  je broj pikova,  $t_{R,L}$  retenciono vreme poslednjeg pika,  $t_0$  je retenciono vreme pika mobilne faze,  $\theta_{s,i}$  je kriterijum razdvajanja po Karlu definisan na sledeći način:

gde su  $H_s$  i  $H_l$  visine pikova,  $H_v$  visina doline između njih,  $t_{R,s}$  i  $t_{R,l}$  retencionna vremena pikova, a  $t_{R,v}$  retenciono vreme doline.

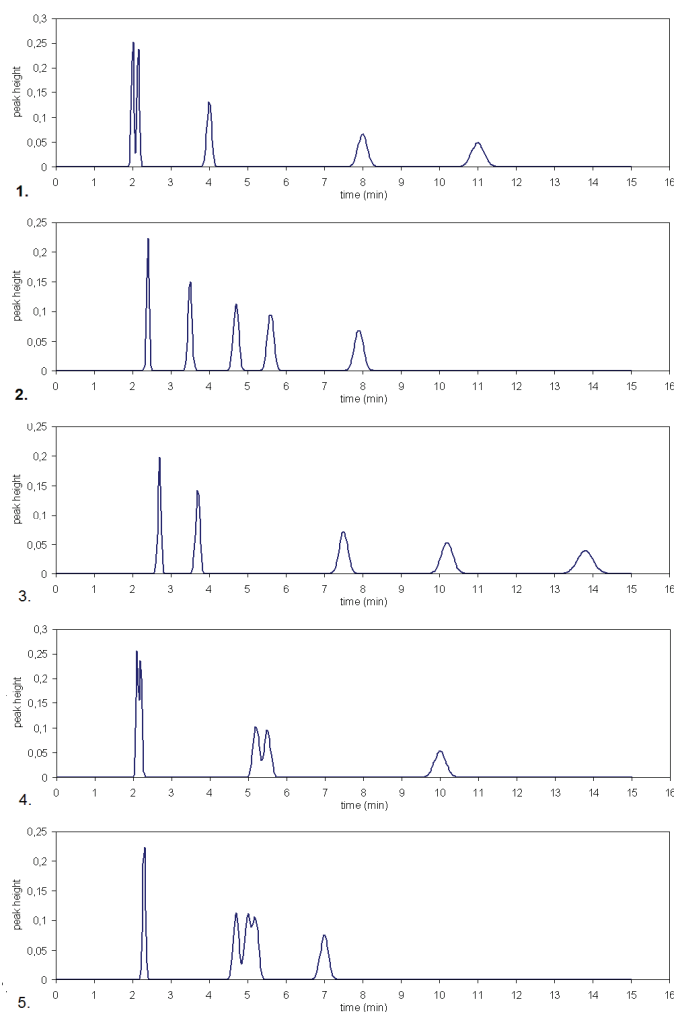
$$\theta_{s,i} = 1 - \left( \frac{(H_v \times |t_{R,l} - t_{R,s}|)}{(|t_{R,v} - t_{R,s}| \times (H_l - H_s) + H_s \times |t_{R,l} - t_{R,s}|)} \right)$$

4. Nova funkcija hromatografskog odgovora ( $N_{CRF}$ ) [4] čija je formula:

$$N_{CRF} = \left( a \left( 1 - \frac{\sum_{i=1}^{N-1} \theta_{s,i}}{N-1} \right) + 1 \right) \left( 1 + \left( \frac{t_f}{t_{opt}} \right)^b \right)$$

gde je  $N$  broj očekivanih pikova,  $\theta_{s,i}$  kriterijum razdvajanja,  $t_f$  retenciono vreme poslednjeg pika,  $t_{opt}$  maksimalna prihvatljiva dužina hromatografske analize, a  $a$  i  $b$  su konstante (u ovom radu  $a = 5$ ,  $b = 1$ ).

U prvoj fazi istraživanja procena efikasnosti ove četiri funkcije vršena je na simuliranim hromatogramima prikazanim na slici 2.



**Slika 2.** Simulirani hromatogrami

Važni hromatografski parametri i vrednosti analiziranih funkcija za datih pet hromatograma prikazani su u tabeli 1.

**Tabela 1.** Hromatografski parametri i dobijene vrednosti analiziranih funkcija

Hromatogram	$\theta_{1/2}$	$\theta_{2/3}$	$\theta_{3/4}$	$\theta_{4/5}$	$Rs_{1/2}$	$Rs_{2/3}$	$Rs_{3/4}$	$Rs_{4/5}$	$t_1$	$t_f$	$B_{CRF}$	$D_{CRF}$	CEF	$N_{CRF}$
1	0,89	1,00	1,00	1,00	1,26	10,39	11,24	5,28	2,0	11,0	31,18	7,92	10,62	2,40
2	1,00	1,00	1,00	1,00	6,39	4,92	2,94	5,72	2,4	7,9	22,27	8,05	8,90	1,79
3	1,00	1,00	1,00	1,00	5,36	11,50	5,14	5,06	2,7	13,8	27,95	8,03	11,90	2,38
4	0,36	1,00	0,61	1,00	0,79	13,61	0,96	9,83	2,1	10,0	29,29	6,02	152,14	4,56
5	1,00	0,76	0,15	1,00	11,80	1,07	0,67	4,93	2,3	7,0	19,70	6,97	218,86	4,02

$\theta_{s,l}$  – kriterijum razdvajanja;  $t_1$  – retenciono vreme prvog pika;  $t_f$  – retenciono vreme poslednjeg pika;  $Rs_{s,l}$  – faktor rezolucije;  $B_{CRF}$  – Beridžova funkcija hromatografskog odgovora [1];  $D_{CRF}$  – Duarteova funkcija hromatografskog odgovora [3]; CEF –Morisova hromatografska eksponencijalna funkcija [2];  $N_{CRF}$  – nova funkcija hromatografskog odgovora [4]

Funkcije  $B_{CRF}$  i  $D_{CRF}$  postižu maksimum, a funkcije CEF i  $N_{CRF}$  minimum kako se hromatogram približava optimumu. Analizim rezultata u tabeli 1 vidimo da je rangiranje hromatograma koje su predložile različite funkcije bitno drugačije. Može se primetiti da je Beridžova funkcija hromatografskog odgovora rangirala simulirane hromatograme redosledom 1, 4, 3, 2, 5. Dakle,  $B_{CRF}$  je hromatogram broj jedan identifikovala kao najbolji iako su prvi i drugi pik loše razdvojeni. Nedostatak ove funkcije ogleda se u činjenici da ona kvalitet razdvajanja procenjuje kao zbir svih rezolucija, pa samim tim rezolucija loše razdvojenih pikova biva maskirana vrednostima rezolucije dobro razdvojenih pikova. Upravo na primeru ovog hromatograma može se videti da je loše razdvajanje prva dva pika maskirano dobrim razdvajanjem trećeg i četvrtog pika.

Za razliku od  $B_{CRF}$ , CEF je date hromatograme rangirala na sledeći način: 2, 1, 3, 4, 5. U formulaciji ove funkcije može se uočiti da je samo prvi član koji procenjuje rezoluciju predstavljen eksponencijalno. Stoga je uticaj dužine trajanja hromatografske analize na ukupnu vrednost funkcije veoma mali.

Duarteova funkcija hromatografskog odgovora predložila je sledeće rangiranje hromatograma: 2, 3, 1, 5, 4. Ovakav redosled je u dobroj korelaciji sa stvarnim kvalitetom hromatograma. Za razliku od  $B_{CRF}$  i CEF,  $D_{CRF}$  kao kriterijum razdvajanja koristi  $\theta$ , a ne faktor rezolucije. Ovakav pristup smanjuje mogućnost maskiranja loše razdvojenih pikova dobro razdvojenim i omogućava procenu separacije i u slučaju pikova koji nemaju Gausov oblik. Nedostatak ove funkcije je činjenica da dužina trajanja hromatografske analize ima mali uticaj na numeričku vrednost funkcije. Ako pogledamo simulirane hromatograme, možemo videti da hromatogram broj dva ima numeričku vrednost 8,5, a hromatogram broj tri 8,3, pa se može zaključiti da je kvalitet ov dva hromatograma približno isti. Međutim, ako se razmotri dužina trajanja analize, može se videti da kod hromatograma broj dva iznosi oko osam minuta, dok kod hromatograma broj tri iznosi čak četrnaest minuta.

Poslednja funkcija čija je efikasnost procenjivana je nova funkcija hromatografskog. Analizom rezultata možemo uočiti da je ova funkcija rangirala hromatograme na sledeći način: 2, 3, 1, 5, 4. Rangiranje je identično kao i ono predloženo sa funkcijom  $D_{CRF}$ . Ipak,  $N_{CRF}$  pokazuje veću fleksibilnost u pogledu biranja ravnoteže između rezolucije i dužine trajanja hromatografske analize zahvaljujući težinskim konstantama  $a$  i  $b$ .

U drugoj fazi istraživanja, funkcije su testirane na eksperimentalno dobijenim podacima iz analize agonista i antagonista  $\beta$  – receptora u HILIC sistemu. Eksperimenti su izvedeni koristeći metodologiju ekpserimentalnog dizajna. Varirana su tri faktora vezana za sastav mobilne faze – udeo acetonitrila, pH vrednost vodene faze i molaritet amonijum-acetata u vodenoj fazi. Intervali ispitivanih faktora prikazani su u tabeli 2.

**Tabela 2.** Ispitivani faktori i njihovi nivoi

Ispitivni faktori	Merne jedinice	Donji nivo (-1)	Gornji nivo (+1)	Nominalni nivo
Udeo ACN (A)	%	80	90	85
pH vodene faze (B)		3,5	5,5	4,5
Molaritet pufera u vodenoj fazi (C)	mM	20	60	40

Eksperimentalni plan generisan je centralnim kompozicionim dizajnom koji se sastoji od punog faktorskog dizajna, zvezda dizajna i ponavljanja u centralnoj tački. Broj eksperimenata u centralnom kompozicionom dizajnu je:

$$2^n + 2n + \text{ponavljanje u centralnoj tački}$$

gde n predstavlja broj faktora koji utiču na sistem. Izvedeno je 4 eksperimenta u centralnoj tački, pa je ukupan broj eksperimenata bio 18.

Za 18 dobijenih hromatograma procenjeni su važni hromatografski parametri i izračunate su vrednosti ispitivanih funkcija. Dobijeni rezultati prikazani su u tabeli 3.

**Tabela 3.** Hromatografski parametri i dobijene vrednosti analiziranih funkcija

Run	$\theta_{1/2}$	$\theta_{2/3}$	$\theta_{3/4}$	$\theta_{4/5}$	$Rs_{1/2}$	$Rs_{2/3}$	$Rs_{3/4}$	$Rs_{4/5}$	$t_1$	$t_f$	$B_{CRF}$	$D_{CRF}$	CEF	$N_{CRF}$
1	0	1	1	1	0,17	1,24	1,02	2,03	2,9	4,9	3,25	6,31	4141,07	3,35
2	1	1	1	1	1,05	1,82	1,65	3,54	6,8	17,6	1,6	8,09	28,82	2,76
3	1	1	1	1	0,98	1,46	1,59	2,18	3,2	8,4	9,4	8,18	29,5	1,84
4	1	1	1	1	3,47	2,51	2,63	4,18	7,1	28,9	-5,2	8,05	18,81	3,89
5	0,09	1	1	1	0,17	0,98	0,97	1,47	2,3	3,9	0,75	6,48	3816,19	2,96
6	1	1	1	1	0,57	1,45	1,51	2,94	4	9,3	9,81	8,16	450,94	1,93
7	0,94	1	1	1	0,42	1,03	1,63	1,71	2,4	5,3	4,53	8,22	947,88	1,65
8	1	1	1	1	2,32	2,31	3,19	4,25	5,3	22,5	2,34	8,07	15,11	3,25
9	0,18	1	1	1	0,17	1,04	0,91	1,6	2,3	3,9	1,91	7,57	3826,35	2,81
10	1	1	1	1	1,72	2,12	2,38	3,17	14	59,2	45,84	8,03	26,33	6,92
11	0,78	1	1	1	0,34	1,08	1,02	2,18	3	5,3	4,96	8,06	1572,19	1,95
12	1	1	1	1	1,69	2,08	1,87	2,82	3,7	11	11,75	8,13	6,91	2,1
13	0,9	1	1	1	0,38	1,37	1,25	2,45	6,5	13,1	3,79	8,01	1785,93	2,59
14	0,64	1	1	1	0,26	1,09	1,29	1,84	2,7	5,41	4,58	7,92	2446,13	2,23
15	1	1	1	1	0,81	1,61	1,71	2,64	3,4	7,6	8,74	8,2	330,53	1,76
16	1	1	1	1	0,59	1,47	1,42	2,3	3,4	7,5	7,9	8,2	368,02	1,75
17	1	1	1	1	0,59	1,29	1,38	2,15	3,3	7,2	7,29	8,21	353,07	1,72
18	0,98	1	1	1	0,63	1,08	1,11	2,49	3,5	8,1	7,95	8,17	311,09	1,85

\*  $\theta_{s,i}$ ;  $t_1$ ;  $t_f$ ;  $Rs_{s,i}$ ;  $B_{CRF}$   $D_{CRF}$ ; CEF i  $N_{CRF}$  imaju isto značenje kao u tabeli 1

Slično kao i kod analize simuliranih hromatograma, četiri ispitivane funkcije predložile su potpuno drugačije rangiranje eksperimentalnih hromatograma. Samim tim, i pretraga eksperimentalnog prostora i identifikacija optimuma biće potpuno drugačiji. To znači da se posebna pažnja mora obratiti na odabir odgovarajuće funkcije hromatografskog odgovora kako bi se izbegla identifikacija lažnih optimuma.

Analizom tabele 3 može se primetiti da su funkcije  $B_{CRF}$  i CEF dale prednost hromatogramima sa velikim vrednostima faktora rezolucije. Poredeći hromatograme pod brojevima 6 i 12 možemo videti da je u oba slučaja postignuto razdvajanje na baznoj liniji, a da hromatogram 6 ima kraću dužinu trajanja analize. Ipak  $B_{CRF}$  i CEF karakterišu hromatogram 12 kao bolji. Sa druge strane, funkcije  $D_{CRF}$  i  $N_{CRF}$  adekvatno su rangirale hromatograme identifikujući one sa razdvajanjem na baznoj liniji i kratkom dužinom trajanja analize kao najbolje. Takođe, može se primetiti veća osetljivost funkcije  $N_{CRF}$  na produženje ukupnog vremena, ako se uporede hromatogrami 7 i 17.



## Zaključak

Ovaj rad prikazao je analizu četiri različite funkcije hromatografskog odgovora na simuliranim i eksperimentalno dobijenim hromatogramima u HILIC sistemu. Pokazano je da funkcije koje procenjuju razdvajanje faktrom  $\theta$  vrše bolju evaluaciju separacije u odnosu na funkcije koje procenjuju razdvajanje faktorom rezolucije. Uočeno je da se mora voditi računa o balansu između članova koji procenjuju separaciju i članova koji procenjuju ukupnu dužinu trajanja analize. Takođe je pokazano da se odabiru funkcije za odgovarajući separacioni problem mora pristupiti sa velikim oprezom kako bi se izbegla identifikacija lažnih optimuma.

## Literatura

- [1] **J.C. Berridge.** Unattended optimization of reversed-phase high-performance liquid chromatographic separations using the modified simplex algorithm, *Journal of Chromatography*, 1982, 244, 1 – 14.
- [2] **V.M. Morris, J.G. Hughes, P.J. Marriott.** Examination of a new chromatographic function, based on an exponential resolution term, for use in optimization strategies: application to capillary gas chromatography separation of phenols, *Journal of Chromatography A*, 1996, 755, 235 – 243.
- [3] **R.M.B.O. Duarte, A.C. Duarte.** A new chromatographic response function for use in size-exclusion chromatography optimization strategies: Application to complex organic mixtures. *Journal of Chromatography A*, 2010, 1217, 7556 – 7563.
- [4] **B. Jancić–Stojanovic, T. Rakic, N. Kostic, A. Vemic, A. Malenovic, D. Ivanovic, M. Medenica.** Advancement in optimization tactic achieved by newly developed chromatographic response function: application to LC separation of raloxifene and its impurities, *Talanta*, 85, 2011, 1453 – 1460.
- [5] **P. Hemstrom, K. Irgum.** Hydrophilic interaction chromatography, *Journal of Separation Science*, 2006, 29, 1784 – 1821.
- [6] **Y. Hsieh.** Potential of HILIC–MS in quantitative bioanalysis of drugs and drug metabolites, *Journal of Separation Science*, 2008, 31, 1481 – 1491.



# HIBRIDNI GENETSKI ALGORITAM ZA JEDNOSTEPENI LOKACIJSKI PROBLEM NEOGRANIČENIH KAPACITETA

Stefan Mišković

*Matematički fakultet, Beograd  
stefan@matf.bg.ac.rs*

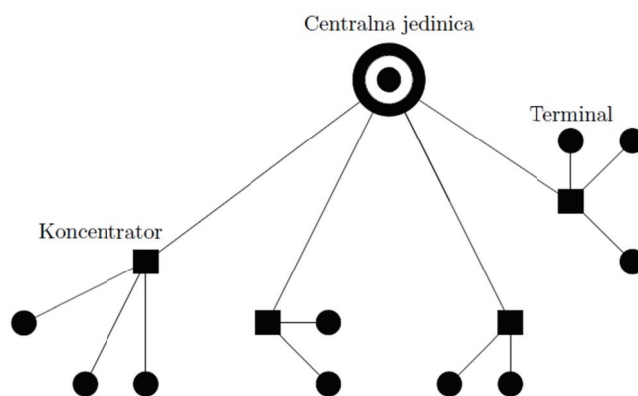
**Apstrakt.** U radu je prikazan hibridni genetski algoritam za jednostepeni lokacijski problem ograničenih kapaciteta. Za dati skup terminala i potencijalnih lokacija za koncentratore, potrebno je odrediti koji će koncentratori biti uspostavljeni tako da suma cena pridruživanja terminala koncentratorima i cena instalacije koncentratora bude najmanja moguća. Pritom je potrebno voditi računa da kapaciteti koncentratora odgovaraju potražnjama terminala. Hibridni genetski algoritam koristi elitističku strategiju, fino gradiranu turnirsku selekciju, uniformno ukrštanje i prostu mutaciju sa zaleđenim bitovima. Za izračunavanje funkcije cilja je korišćen CPLEX. U svakoj generaciji se primenjuju i dve pomoćne heuristike lokalne pretrage, inverzija i transpozicija. Algoritam je testiran na ORLIB instancama i za svaku test instancu je dostignut optimalan rezultat.

**Ključne reči.** Genetski algoritam, lokacijski problem, lokalna pretraga, hibridizacija.

## 1. Uvod

U literaturi se može pronaći veliki broj problema koji se tiču problema dizajna telekomunikacionih mreža. Pri dizajniranju tih mreža nepraktičnim se pokazalo direktno prenošenje podataka od terminala do centralne telekomunikacione jedinice. Zbog toga se često pri njihovom uspostavljanju ubacuju koncentratori koji imaju ulogu da sav protok podataka skupe od terminala, koji se zatim prosleđuje centralnoj jedinici. Često su ti koncentratori raspoređeni u više nivoa. Iako moderne telekomunikacione mreže imaju hijerarhijsku strukturu, u mnogim optimizacionim modelima dovoljno je razmatrati deo hijerarhijske strukture sa jednim nivoom. Detaljan pregled modelâ koji se tiču dizajna telekomunikacionih mreža se može pronaći u [6].

U ovom radu se razmatra jednostepeni lokacijski problem ograničenih kapaciteta (Capacitated Facility Location Problem – CFLP). Kod ovog problema se podaci prenose iz terminala preko koncentratora do centralne jedinice. Primer jednog takvog modela je prikazan na slici 1. Kod problema CFLP je dat skup terminala i skup potencijalnih lokacija za koncentratore. Pritom je potrebno odrediti na kojim lokacijama će koncentratori biti uspostavljeni. Pritom se svaki terminal pridružuje jednom ili više koncentratora i tada se svaki od tih koncentratora smatra uspostavljenim. Za svako pridruživanje u celosti je poznata njegova cena, kao i cena instalacije koncentratora. Za svaki terminal poznata je njegova potražnja, a za svaki koncentrator poznat je njegov kapacitet. Potrebno je minimizovati zbir cena pridruživanja terminala koncentratorima i cena instalacije uspostavljenih koncentratora tako da budu zadovoljene potražnje terminala i da se ne prekorači kapacitet nijednog uspostavljenog koncentratora. Problem je do sada rešavan standardnim metaheuristikama lokalne pretrage, tabu pretrage i genetskih algoritama ([1], [3], [4], [7], [8], itd.). Prvi put u literaturi je na ovom problemu primenjena hibridizacija genetskog algoritma sa heuristikama lokalne pretrage.



Slika 1. Primer modela hijerarhijske telekomunikacione mreže

## 2. Matematička formulacija

Neka su uvedene sledeće oznake:

- $I = \{1, 2, \dots, m\}$  – skup terminala,
- $J = \{1, 2, \dots, n\}$  – skup potencijalnih lokacija za koncentratore,
- $c_{ij}$  – cena pridruživanja celog terminala  $i$  koncentratoru  $j$ ,
- $f_j$  – cena uspostavljanja koncentratora  $j$ ,
- $d_i$  – potražnja terminala  $i$ ,
- $s_j$  – kapacitet koncentratora  $j$ .

Neka promenljive imaju sledeća značenja:

- $x_{ij}$  – deo terminala  $i$  koji se pridružuje koncentratoru  $j$ ,
- $y_j$  – binarna promenljiva koja je jednaka 1 ako je koncentrator  $j$  uspostavljen, a 0 inače.

Potrebno je izračunati

$$\min \sum_{i=1}^m \sum_{j=1}^n c_{ij} x_{ij} + \sum_{j=1}^n f_j y_j \quad (1)$$

uz sledeća ograničenja:

$$\sum_{j=1}^n x_{ij} = 1 \quad \forall i \in I, \quad (2)$$

$$\sum_{i=1}^m d_i x_{ij} \leq s_j y_j \quad \forall j \in J, \quad (3)$$

$$x_{ij} \leq y_j \quad \forall i \in I \quad j \in J, \quad (4)$$

$$0 \leq x_{ij} \leq 1 \quad i \in I \quad j \in J, \quad (5)$$

$$y_j \in \{0, 1\} \quad \forall j \in J. \quad (6)$$

Minimizacija funkcije cilja (1) se svodi na minimizaciju zbira cene pridruživanja terminala koncentratorima i cene instalacije koncentratora. Značenja ograničenja su sledeća:

- (2) svaki terminal je u celosti pridružen nekim koncentratorima;
- (3) ukoliko je koncentrator uspostavljen, potražnja njemu pridruženih terminala ne sme biti veća od njegovog kapaciteta, a ako nije uspostavljen, nijedan terminal mu ne sme biti pridružen;
- (4) terminali mogu biti pridruženi samo uspostavljenim koncentratorima;
- (5) promenljiva  $x_{ij}$  uzima vrednosti na segmentu  $[0, 1]$ .
- (6) promenljiva  $y_{ij}$  je binarnog karaktera.

Problem CPLP je NP-težak jer predstavlja uopštenje prostog lokacijskog problema (Uncapacitated Facility Location Problem – UFLP) za koji je u [9] pokazano da je NP-težak.

### 3. Hibridni genetski algoritam

Genetski algoritmi su meteheuristike koje se zasnivaju na konceptu prirodne evolucije. Kreira se skup jedinki koje čine jednu populaciju i svakoj jedinki se dodeli kôd određene dužine, nakon čega se operatori primenjuju na kôd, a ne na samu jedinku. U svakoj iteraciji genetskog algoritma se na jedinke primenjuju genetski operatori (selekcija, ukrštanje i mutacija) u cilju pospešivanja rešenja. Svaka jedinka ima svoju prilagođenost, a bolje jedinke imaju veću šansu da prežive i time daju bolje potomke, pri čemu se stvara nova populacija. Algoritam se izvršava sve dok nije ispunjen kriterijum završetka. Često se pri rešavanju složenih problema kombinatorne optimizacije dešava da genetski algoritmi ne mogu dati optimalne rezultate, zbog čega je potrebno takav algoritam hibridizovati sa dodatnim heuristikama u cilju pospešivanja rešenja. Takav je i slučaj pri rešavanju CFLP, gde je genetski algoritam hibridizovan sa heuristikom lokalnog pretraživanja.

#### 3.1 Način kodiranja i funkcija prilagođenosti

Za predstavljanje jedinki korišćeno je binarno kodiranje. Svaka jedinka je predstavljena binarnim stringom dužine  $n$  koji odgovara potencijalnim lokacijama za koncentratore. Odgovarajući bit ima vrednost 1 ukoliko je dati koncentrator uspostavljen, a inače ima vrednost 0. Na primer, genetski kôd 10101 (za  $n = 5$ ) označava da su uspostavljeni prvi, treći i peti koncentrator.

Funkcija prilagođenosti se poklapa sa funkcijom cilja. Za njeno izračunavanje je korišćen CPLEX. Pri fiksiranim koncentratorima razmatrani problem postaje polinomijalan, zbog čega je u kratkom vremenskom roku CPLEX je u mogućnosti da vrati rešenje. Za CPLEX je korišćena sledeća matematička formulacija problema (korišćene oznake i uslovi odgovaraju oznakama iz (1) – (6), s tim što se može bez umanjena opštosti pretpostaviti da je skup  $J$  skup uspostavljenih koncentratora): potrebno je izvršiti minimizaciju funkcije

$$\sum_{i=1}^m \sum_{j=1}^n c_{ij} x_{ij} + \sum_{j=1}^n f_j \quad (7)$$

uz sledeća ograničenja:

$$\sum_{j=1}^n x_{ij} = 1 \quad \forall i \in I, \quad (8)$$

$$\sum_{i=1}^m d_i x_{ij} \leq s_j \quad \forall j \in J, \quad (9)$$

$$0 \leq x_{ij} \leq 1 \quad \forall i \in I \quad \forall j \in J. \quad (10)$$

### 3.2 Genetski operatori i ostali aspekti

U genetskom algoritmu je primenjena elitistička strategija biranja jedinki za narednu generaciju. To znači da se 2/3 jedinki prenosi direktno. Na preostalu trećinu se primenjuju genetski operatori selekcije, ukrštanja i mutacije. Korišćena je fino gradirana turnirska selekcija, koja predstavlja jednu varijantu turnirske selekcije, gde je umesto unapred zadate celobrojne poznata željena realno vrednosna veličina turnira  $F$ , a veličine turnirâ se biraju tako da je njihova aritmetička sredina što približnija  $F$  [5]. U ovoj implementaciji je uzeto  $F = 5,6$ , a veličine turnirâ su birane tako da se razlikuju za 1 i iznose  $\lfloor F \rfloor$  i  $\lceil F \rceil$ . Korišćeno je uniformno ukrštanje sa verovatnoćom 0,85 i prosta mutacija sa zaleđenim bitovima. Verovatnoća mutacije je  $0,4/n$ , dok njena verovatnoća na zaleđenim bitovima iznosi  $2,5/n$ .

Veličina populacije iznosi 30 jedinki. U narednu generaciju se uvek direktno prenosi 20 najprilagođenijih jedinki, a na ostalih 10 se primenjuju genetski operatori. Početna populacija se generiše tako što se za svaku jedinku generišu bitovi koji odgovaraju koncentratorima. Svaki uzima vrednost 1 sa verovatnoćom 0,35. Kriterijum zaustavljanja se odnosi na maksimalan broj iteracija i maksimalan broj ponavljanja jedinki. U ovoj implementaciji ta dva parametra iznose 5000 i 2000, respektivno. U algoritmu se mogu naći i nekorektne jedinke, koje se ne koriguju.

### 3.3 Pomoćne heuristike

Genetski algoritam je hibridizovan sa dve heuristike lokalnog pretraživanja. One se primenjuju na sve jedinke populacije dajući im podjednaku šansu da budu popravljenije (sem na nekorektne jedinke). One se u svakoj iteraciji izvršavaju nakon primene genetskih operatora. Prva od te dve heuristike u svakoj iteraciji vrši invertovanje proizvoljno izabranog bita. Ukoliko je nova prilagođenost jedinke bolja od prethodne, ona se ažurira. Inače, algoritam se prekida. Heuristika se izvršava sve dok ima poboljšanja. Princip primene druge heuristike lokalne pretrage, transpozicije, je isti. Jedina razlika se ogleda u tome što se u svakoj iteraciji umesto invertovanja jednog bita pokušava sa zamenom dva proizvoljno izabrana bita. Osnovna struktura hibridnog genetskog algoritma je prikazana na slici 2.

```
GenerisanjePocetnePopulacije()
while not KriterijumZaustavljanja
    FunkcijaPrilagodjenosti()
    Selekcija()
    Ukrstanje()
    Mutacija()
    LPInverzija()
    LPTranspozicija()
end
```

Slika 2. Osnovna struktura hibridnog genetskog algoritma za CFLP

## 4. Eksperimentalni rezultati

Izvršena su testiranja na standardnim ORLIB instancama (dimenzije do 1000) koje su dostupne na internetu na adresi <http://people.brunel.ac.uk/~mastjjb/jeb/orlib/capinfo.html>. Algoritam je na svakoj instanci dostigao optimalne rezultate. Rezultati izvršavanja su prikazani u tabelama 1 (za manje test instance) i 2 (za veće test instance). Implementacija je izvršena u programskom jeziku C pod Windows 7 operativnim sistemom. Korišćen je Intel Core i5-2430M procesor brzine 2,4GHz i RAM memorija veličine 8GB. Za svaku test instancu program je pokretan po 10 puta.

Nazivi kolona u tabelama 1 i 2 imaju sledeća značenja:

Naziv – naziv test instance;

$m$  – broj terminala za datu test instancu;

$n$  – broj koncentratora za datu test instancu;

Rešenje – poznati optimalan rezultat za datu instancu;

Rezultat – najbolje rešenje koje je dostigao genetski algoritam (ukoliko je algoritam dostigao optimalan rezultat iz kolone Rešenje, u odgovarajućem polju je naznačeno Opt);

$t[s]$  – srednje vreme u sekundama za koje algoritam dostiže najbolje rešenje;

$agap$  – srednje odstupanje od najboljeg rešenja koje je dostigao genetski algoritam (u procentima);

$\sigma$  – standardna devijacija (u procentima).

**Tabela 1.** Rezultati izvršavanja algoritma na ORLIB instancama manjih dimenzija

Naziv	$m$	$n$	Rešenje	Rezultat	$t[s]$	$agap$	$\sigma$
cap41	50	16	1.040.444,375	Opt	0,4	0,000	0,000
cap42	50	16	1.098.000,450	Opt	0,3	0,000	0,000
cap43	50	16	1.153.000,450	Opt	0,8	0,000	0,000
cap44	50	16	1.235.500,450	Opt	0,2	0,000	0,000
cap51	50	16	1.025.208,225	Opt	0,9	0,000	0,000
cap61	50	16	932.615,750	Opt	1,3	0,000	0,000
cap62	50	16	977.799,400	Opt	1,2	0,000	0,000
cap63	50	16	1.014.062,050	Opt	0,4	0,000	0,000
cap64	50	16	1.045.650,250	Opt	0,7	0,000	0,000
cap71	50	16	932.615,750	Opt	1,4	0,000	0,000
cap72	50	16	977.799,400	Opt	0,6	0,000	0,000
cap73	50	16	1.010.641,450	Opt	0,6	0,000	0,000
cap74	50	16	1.034.976,975	Opt	1,0	0,000	0,000
cap81	50	25	838.499,288	Opt	1,9	0,000	0,000
cap82	50	25	910.889,563	Opt	2,3	0,000	0,000
cap83	50	25	975.889,563	Opt	2,0	0,000	0,000
cap84	50	25	1.069.369,525	Opt	2,0	0,000	0,000
cap91	50	25	796.648,438	Opt	2,5	0,000	0,000
cap92	50	25	855.733,500	Opt	2,1	0,000	0,000
cap93	50	25	896.617,538	Opt	2,3	0,000	0,000
cap94	50	25	946.051,325	Opt	2,6	0,000	0,000
cap101	50	25	796.648,437	Opt	2,7	0,000	0,000
cap102	50	25	854.704,200	Opt	2,5	0,000	0,000
cap103	50	25	893.782,112	Opt	3,2	0,000	0,000
cap104	50	25	928.941,750	Opt	2,1	0,000	0,000
cap111	50	50	826.124,713	Opt	29,3	0,000	0,000
cap112	50	50	901.377,213	Opt	33,1	0,000	0,000
cap113	50	50	970.567,750	Opt	27,4	0,000	0,000
cap114	50	50	1.063.356,488	Opt	38,3	0,000	0,000
cap121	50	50	793.439,563	Opt	33,4	0,000	0,000
cap122	50	50	852.524,625	Opt	31,3	0,000	0,000
cap123	50	50	895.302,325	Opt	29,3	0,000	0,000
cap124	50	50	946.051,325	Opt	47,4	0,000	0,000
cap131	50	50	793.439,562	Opt	47,2	0,000	0,000
cap132	50	50	851.495,325	Opt	39,3	0,000	0,000
cap133	50	50	893.076,712	Opt	44,6	0,000	0,000
cap134	50	50	928.941,750	Opt	52,0	0,000	0,000

**Tabela 2.** Rezultati izvršavanja algoritma na ORLIB instancama većih dimenzija

Naziv	$m$	$n$	Rešenje	Rezultat	$t[s]$	$agap$	$\sigma$
capa1	1.000	100	19.240.822,449	Opt	2.323,5	0,000	0,000
capa2	1.000	100	18.438.046,543	Opt	2.412,2	0,000	0,000
capa3	1.000	100	17.765.201,949	Opt	1.986,7	0,000	0,000
capa4	1.000	100	17.160.439,012	Opt	2.523,6	0,000	0,000
capb1	1.000	100	13.656.379,578	Opt	2.743,3	0,000	0,000
capb2	1.000	100	13.361.927,449	Opt	2.133,4	0,000	0,000
capb3	1.000	100	13.198.556,434	Opt	2.622,0	0,000	0,000
capb4	1.000	100	13.082.516,496	Opt	2.033,2	0,000	0,000
capc1	1.000	100	11.646.596,974	Opt	3.147,3	0,026	0,042
capc2	1.000	100	11.570.340,289	Opt	3.122,2	0,000	0,000
capc3	1.000	100	11.518.743,744	Opt	2.846,4	0,000	0,000
capc4	1.000	100	11.505.767,394	Opt	3.423,9	0,066	0,094

## 6. Zaključak

U ovom radu je razmatran jednostepeni lokacijski problem ograničenih kapaciteta (CFLP). Prvi put je za njegovo rešavanje iskorišćen genetski algoritam hibridizovan sa metaheuristikama lokalne pretrage. Kako se običan genetski algoritam nije pokazao dobrim u primeni na CFLP, njegova hibridizacija sa dve metode lokalne pretrage se pokazala vrlo dobrom, budući da je, gde god su oni poznati, dala optimalne rezultate. Predložena metaheuristika se može primeniti i na srodne probleme kombinatorne optimizacije. Takođe, algoritam ostavlja mogućnost paralelizacije, što može dodatno doprineti njegovoj efikasnosti.

## Literatura

- [1] **J. E. Beasley.** An algorithm for solving large capacitated warehouse location problems. *European Journal of Operational Research* 33, 1988, pp. 314–325, 1988.
- [2] **B. L. Wildbore.** *Theoretical and computational analysis of the two stage capacitated plant location problem.* PhD thesis, Massey University, Palmerston North, New Zealand, 2008.
- [3] **G. Cornuejols, R. Sridharan, J. M. Thizy.** A comparison of heuristics and relaxations for the capacitated plant location problem. *European Journal of Operational Research* 50, pp. 280–297, 1991.
- [4] **W. Domschke, A. Drexl.** ADD-heuristics' starting procedures for capacitated plant location models. *European Journal of Operational Research* 21, pp. 47–53, 1985.
- [5] **V. Filipović.** Fine-grained tournament selection operator in genetic algorithms. *Computing and Informatics*, vol. 22, No. 2, pp. 143-162, 2003.
- [6] **E. Gourdin, M. Labb, H. Yaman.** *Telecommunication and location. Facility location: applications and theory.* Springer-Verlag, New York, pp. 274–305, 2002.
- [7] **S. K. Jacobsen.** Heuristics for the capacitated plant location model. *European Journal of Operational Research* 12, pp. 253–261, 1983.
- [8] **B. M. Khumawala.** An efficient heuristic procedure for the capacitated warehouse location problem. *Naval Research Logistics Quarterly* 21, pp. 609–623, 1974.
- [9] **J. Krarup, P. M. Pruzan.** The simple plant location problem: survey and synthesis, *European Journal of Operational Research*. vol. 12, pp. 36-81, 1983.



# „CYCLE-CANCELING ALGORITHM“ ZA REŠAVANJE PROBLEMA PROTOKA SA MINIMALNOM CENOM

Olivera Janković

**Rezime:** U praksi, problem protoka sa minimalnom cenom se često sreće u skoro svim granama industrije, jer za rešavanje najvećeg broja praktičnih problema, odlučujući značaj imaju troškovi neophodni za realizaciju posmatranih protoka. U radu je pokazano da je ovaj problem ekvivalentan transportnom problemu, a zatim je opisan „Cycle – canceling algorithm“ koji nalazi optimalno rešenje u pseudopolinomijalnom vremenu. Njegov rad je ilustrovan na kratkom primeru.

**Ključne reči:** „Cycle – canceling algorithm“, problem protoka sa minimalnom cenom, uslov optimalnosti, rezidualna mreža

## UVOD

Problem protoka sa minimalnom cenom se često sreće u skoro svim granama industrije, uključujući poljoprivredu, komunikacije, odbranu, energiju, medicinu, proizvodnju i transport. Primera radi, ako je reč o transportnoj mreži, onda je jasno da uz poštovanje svih ograničenja koja postoje u posmatranoj transportnoj mreži, odlučujući značaj za realizaciju transporta (protoka) imaju troškovi transporta i da se uvek zahteva njihova minimalizacija uz unapred data ograničenja. Štaviše, može se pokazati da je transportni problem specijalan slučaj problema protoka sa minimalnom cenom. Samim tim, svaki algoritam za rešavanje problema protoka sa minimalnom cenom, rešava i transportni problem. Opisaćemo „Cycle-canceling algorithm“ za nalaženje optimalnog rešenja u pseudopolinomijalnom vremenu. Za konstrukciju ovog algoritma je neophodno uvesti pojam rezidualne mreže i izvesti odgovarajući uslov optimalnosti.

## 1. Matematički model problema protoka sa minimalnom cenom

Neka je  $G = (N, A)$  usmerena mreža, gde je  $N$  skup od  $n$  čvorova, a  $A$  skup od  $m$  usmerenih grana. Promenljivu  $x_{ij}$  koja odgovara grani  $(i, j)$  sa početkom u čvoru  $i$  i krajem u čvoru  $j$  interpretiramo kao protok kroz tu granu, a odgovarajuće  $c_{ij}$  kao cenu tog protoka. Takođe svakoj grani  $(i, j) \in A$  pridružujemo kapacitet  $u_{ij}$  (koji označava maksimalni mogući protok na grani) i  $l_{ij}$  (koji označava minimalni neophodni protok na grani). Neka je  $b_i, i \in N$ , količina protoka koja prolazi kroz čvor  $i \in N$  (tj.  $b_i$  predstavlja ponudu ili potražnju). Čvor  $i$ , kome odgovara  $b_i > 0$ , zovemo izvor snage (ponude)  $b_i$ ; čvor  $i$ , kome odgovara  $b_i < 0$ , zovemo ponor snage (potražnje); čvorove koji odgovaraju nultim komponentama  $b_i$ , zovemo pretovarna mesta. Funkcija cilja predstavlja ukupnu cenu protoka kroz mrežu. [1,2]

Matematički model ovog zadatka je

$$(\min) \sum_{(i,j) \in A} c_{ij} x_{ij} \quad (1.a)$$

$$\sum_{(j,i) \in A} x_{ji} - \sum_{(i,j) \in A} x_{ij} = b_i \text{ za sve } i \in N \quad (1.b)$$

$$l_{ij} \leq x_{ij} \leq u_{ij} \text{ za sve } (i,j) \in A \quad (1.c)$$

Ograničenja (1.b) se nazivaju „mass balance“ ograničenja. Suma  $\sum_{(j,(i,j) \in A} x_{ij}$  predstavlja ukupan odliv iz čvora  $i$ , a  $\sum_{(i,(i,j) \in A} x_{ij}$  predstavlja ukupan priliv u čvor.

U većini slučajeva je  $l_{ij} = 0$ , pa ako se drugačije ne zahteva, smatraćemo da je  $0 \leq x_{ij} \leq u_{ij}$ .

Takođe, neka je  $C$  najveća količina troškova bilo koje grane, a  $U$  najveći obim bilo koje ponude/potražnje.

Transportni problem je specijalni slučaj ove klase problema za  $l_{ij} = 0$ ,  $u_{ij} = \infty$ . Mreža  $G$  je pritom bipartitni digraf sa  $m$  izvora i  $n$  ušća. Poznata je ponuda svakog izvora  $i$ , potražnja svakog ponora kao i cena transporta  $c_{ij}$  jedinične količine robe od izvora  $i$  do ponora  $j$ , za svako  $i=1,2,\dots,m$ ,  $j=1,2,\dots,n$ . Cilj je prevesti svu ponuđenu robu i zadovoljiti potražnje uz minimalne ukupne transportne troškove.

LP formulacija ovog problema je:

$$(\min) \sum_{(i,j) \in A} c_{ij} x_{ij}$$

$$\sum_{(j,(i,j) \in A} x_{ij} = b(i) \quad \text{za sve } i \in N_1$$

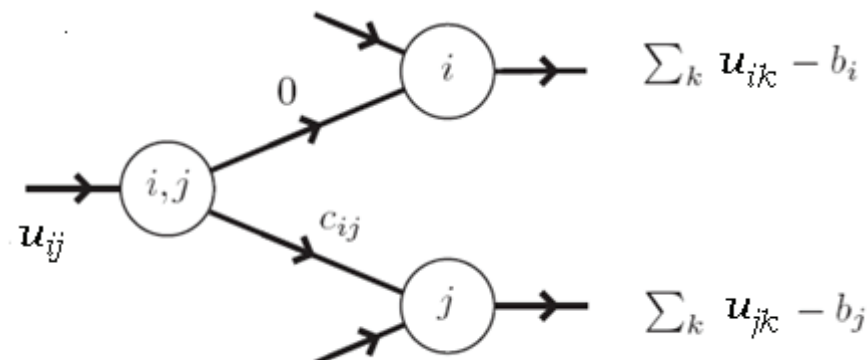
$$\sum_{(i,(i,j) \in A} x_{ij} = -b(j) \quad \text{za sve } j \in N_2$$

$$x_{ij} \geq 0 \quad \text{za sve } (i,j) \in A$$

gde je  $N_1$  skup izvor čvorova, a  $N_2$  skup ponor čvorova, tako da je  $N = N_1 \cup N_2$ . Skup grana je definisan sa  $A = \{(i,j) \mid i \in N_1, j \in N_2\}$ . Ne postoje gornje granice protoka.

**Lema 1** [3] Svaki problem protoka sa minimalnom cenom je ekvivalentan transportnom problemu.

Dokaz: Posmatrajmo problem protoka sa minimalnom cenom za koji su  $l_{ij} = 0$ ,  $u_{ij} < \infty$ , i podaci  $G = (N, A)$ ,  $c_{ij}$ ,  $b_i$  definisani kao ranije. Želimo da ovaj problem svedemo na transportni praveći bipartitni digraf snabdevača i potrošača. Za svaku granu  $(i,j) \in A$  konstruišemo čvor  $i,j$  koji predstavlja snabdevača sa ponudom  $u_{ij}$ . Potrošači će biti čvorovi  $i \in N$ , sa potražnjom  $\sum_{k:(i,k) \in A} u_{ik} - b_i$ . Zatim povežemo svakog snabdevača  $i,j$  sa potrošačima  $i$  i  $j$  granama neograničenog kapaciteta. I neka je cena transporta kroz grane definisana na sledeći način  $c_{i,i} = 0$  i  $c_{i,j} = c_{ij}$ .



Pokažimo da postoji „1-1“ preslikavanje između dopustivih protoka ova dva problema i da su cene protoka iste. Da bismo ovo pokazali, neka je  $f_{ij}$  protok od čvora  $i,j$  do  $j$ , a  $u_{ij} - f_{ij}$  protok od  $i,j$  do  $i$ . Očigledno je da razmatramo problem  $(\min) \sum_{(i,j) \in A} c_{ij} f_{ij}$ . Ukupna količina protoka kroz čvor  $i$  iznosi  $\sum_{j:(i,j) \in A} (u_{ij} - f_{ij}) + \sum_{j:(j,i) \in A} f_{ji}$ . Ovaj protok mora biti jednak  $\sum_{j:(i,j) \in A} u_{ij} - b_i$ .

Očigledno je i  $0 \leq f_{ij} \leq u_{ij}$ . Ovim smo dobili ograničenje za problem protoka sa minimalnom cenom, čime je lema dokazana.

## 2. „Cycle-canceling algorithm“

Ovaj algoritam je prvi predložio Klein, 1967. godine, pa se ponekad naziva i Klein-ov algoritam [4]. Oslanja se na koncept rezidualne mreže. Rezidualnu mrežu  $G(x)$  koja odgovara protoku  $x$ , definišemo na sledeći način: granu  $(i, j) \in A$  zamenjujemo sa dve grane,  $(i, j)$  i  $(j, i)$ . Grana  $(i, j)$  ima cenu  $c_{ij}$  i rezidualni kapacitet  $r_{ij} = u_{ij} - x_{ij}$ , a grana  $(j, i)$  ima cenu  $c_{ji} = -c_{ij}$  i rezidualni kapacitet  $r_{ji} = x_{ij}$ . Rezidualna mreža se sastoji samo od grana sa pozitivnim rezidualnim kapacitetom.

### Uslov optimalnosti negativnog ciklusa

**Teorema 1.** (Uslov optimalnosti negativnog ciklusa)

Dopustivo rešenje  $x^*$  je optimalno rešenje problema protoka sa minimalnom cenom ako i samo ako zadovoljava uslove optimalnosti negativnog ciklusa, odnosno, rezidualna mreža  $G(x^*)$  ne sadrži negativan (usmeren) ciklus.

Dokaz:

Pretpostavimo da je  $x$  neki dopustivi protok i  $G(x)$  sadrži negativan ciklus. Možemo da poboljšamo objektivnu vrednost funkcije širenjem pozitivnog protoka po ciklusu. Zato, ako je  $x^*$  optimalan protok,  $G(x^*)$  ne može sadržati negativan ciklus.

Sada pretpostavimo da je  $x^*$  dopustivi protok i  $G(x^*)$  ne sadrži negativan ciklus. Neka je  $x'$  optimalan protok i  $x^* \neq x'$ . Teorija dekompozicije protoka pokazuje da razlika vektora  $x' - x^*$  može da se razloži na  $m$  pojačanih ciklusa u odnosu na protok  $x^*$  i zbir cena protoka na ovim ciklusima je jednaka  $cx' - cx^*$ . Pošto su dužine svih ciklusa u  $G(x^*)$  nenegativne,  $cx' - cx^* \geq 0$ , tj.  $cx' \geq cx^*$ . Ali, pošto je  $x'$  optimalan protok,  $cx' \leq cx^*$ . Zato je  $cx' = cx^*$ , pa je  $x' = x^*$ , što je u suprotnosti sa pretpostavkom. Dakle, ako  $G(x^*)$  ne sadrži negativan ciklus,  $x^*$  mora biti optimalan. Teorema je dokazana.

Uslov optimalnosti negativnog ciklusa tvrdi da je dopustivi protok  $x$  optimalan ako i samo ako rezidualna mreža  $G(x)$  ne sadrži negativan ciklus. Ovaj uslov ukazuje na algoritamski pristup rešavanja problema protoka sa minimalnom cenom, nazvanog „cycle-canceling algorithm“. Algoritam najpre nalazi dopustivo rešenje  $x$  rešavanjem problema maksimalnog protoka (na primer preko Ford-Fulkersonove metode [5]), a zatim pokušava da nađe negativan ciklus u svakoj iteraciji i poboljšava protok na njemu. Algoritam se završava kada mreža ne sadrži negativan ciklus. Tada je, prema teoremi 1, protok optimalan. Sledi formalan opis ovog algoritma [7,8].

**algorithm** „cycle-canceling“ ;

**begin**

    ustpostaviti dopustivi protok  $x$  u mreži;

**while**  $G(x)$  sadrži negativan ciklus **do**

**begin**

koristiti neki algoritam za nalaženje negativnog ciklusa  $W$ ;

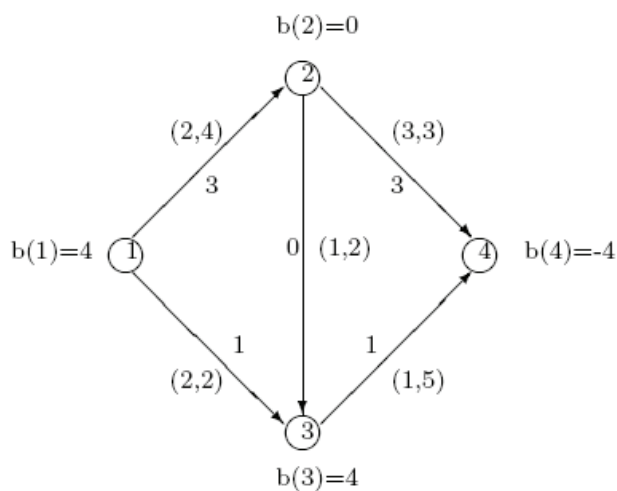
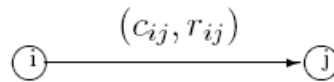
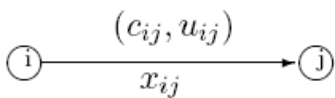
$$\delta := \min\{r_{ij} : (i,j) \in W\};$$

povećati  $\delta$  jedinica protoka u ciklusu  $W$  i ažurirati  $G(x)$ ;

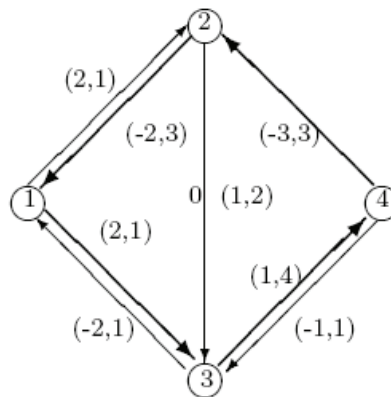
end;

end;

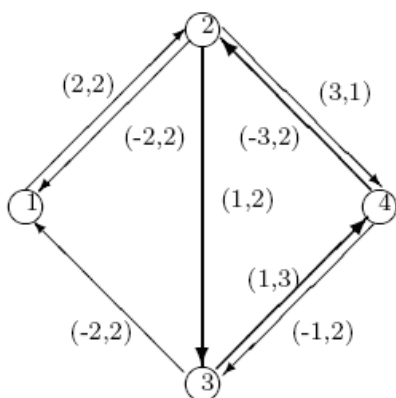
Za ilustraciju ovog algoritma razmotrićemo sledeći primer:



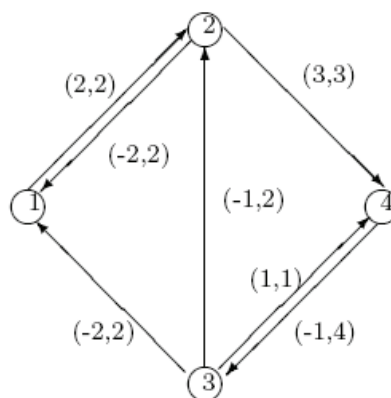
(a) primer mreže sa dopustivim protokom  $x$



(b) rezidualna mreža  $G(x)$



(c) rezidualna mreža nakon povećanja 1 jedinice protoka duž ciklusa 4-2-1-3-4



(d) rezidualna mreža nakon povećanja 2 jed. protoka duž ciklusa 4-2-3-4

Slika 1.a) prikazuje primer mreže sa protokom  $x$ , a slika 1.b) odgovarajuću rezidualnu mrežu. Pretpostavimo da algoritam počinje izborom ciklusa 4-2-1-3-4, čija je cena -2 i minimalni rezidualni kapacitet 1. Algoritam poboljšava 1 jedinicu protoka duž ovog ciklusa i slika 1.c) prikazuje modifikovanu rezidualnu mrežu. Dalje, pretpostavimo da algoritam bira ciklus 4-2-3-4, čija je cena -1 i minimalni rezidualni kapacitet 2. Algoritam poboljšava 2 jedinice protoka duž ovog ciklusa i slika 1.d) prikazuje modifikovanu rezidualnu mrežu. Pošto ova rezidualna mreža ne sadrži negativan ciklus, algoritam se završava.

Sada ćemo razmotriti broj iteracija koje algoritam obavlja. Kako je  $c_{ij} \leq C$  i  $x_{ij} \leq U$  za sve  $(i,j) \in A$ ,  $mCU$  je gornja granica početne cene protoka, a kako je  $c_{ij} \geq -C$  i  $x_{ij} \leq U$  za sve  $(i,j) \in A$ ,  $-mCU$  je donja granica za optimalnu cenu protoka, problema protoka sa minimalnom cenom. U svakoj iteraciji, algoritam menja vrednost funkcije cilja za vrednost  $(\sum_{(i,j) \in W} c_{ij})\delta$ , koja je strogo negativna. Pošto pretpostavljamo da su svi podaci celi brojevi, algoritam se završava u roku od  $O(mCU)$  iteracija. Obratimo pažnju da u algoritmu treba pronaći negativne cikluse za šta postoje mnogi algoritmi. Jedan od algoritama se naziva „FIFO label-correcting“ algoritam za problem najkraćeg puta. On je trenutno najbolji jako polinomijalan algoritam za rešavanje problema najkraćeg puta sa negativnim dužinama grana i zahteva  $O(mn)$  vremena. Znači, algoritam „cycle-canceling“ se izvršava u  $O(mCU)$  iteracija i zahteva  $O(m^2CU)$  vremena.

## ZAKLJUČAK

U radu je pokazano da je svaki problem protoka sa minimalnom cenom ekvivalentan transportnom problemu i izložen je „Cycle-canceling algorithm“ za rešavanje problema protoka sa minimalnom cenom. Takođe, dokazano je da algoritam uspeva da pronađe optimalno rešenje u pseudopolinomijalnom vremenu, u mrežama sa celobrojnim podacima.

## LITERATURA

- [1] D. Cvetković, M. Čangalović, V. Kovačević-Vujičić, Đ. Dugošija, S. Simić, J. Vuleta, „Kombinatorna optimizacija“, DOPIS,1996.
- [2] Lavoslav Čaklović, Matematičko modeliranje (transportni problem), Univerzitet u Zagrebu, Zagreb 21.12.2004.
- [3] R. R. Weber, Mathematics of Operational Research (lecture notes), UNIVERSITY OF CAMBRIDGE, Cambridge, UK, December 20, 2010.
- [4] M. Klein, A Primal Method For Minimal Cost Flows With Applications To the Assignment And Transportation Problems, Management Science, 14, 205-220 (1967).
- [5] L. R. Ford and D. R. Fulkerson, *Flows In Networks*, Princeton Univ. Press, Princeton,NJ (1962).
- [6] Andrew V. Goldberg and Robert E. Tarjan (1989). "Finding minimum-cost circulations by canceling negative cycles". Journal of the ACM 36 (4): 873–886.
- [7] ALGORITHMS FOR MINIMUM-COST FLOWS, Jingping Liu, Computer Science Department, The University of Western Ontario 2003.
- [8] Ravindra K. Ahuja, Thomas L. Magnanti, and James B. Orlin (1993), Network Flows: Theory, Algorithms and Applications. Prentice-Hall, Inc.. ISBN 0-13-617549-X.



# METAHEURISTIKA TABU PRETRAŽIVANJA ZA REŠAVANJE GENERALIZOVANOG PROBLEMA PRIDRUŽIVANJA

Miloš Stanković

Univerzitet u Beogradu, Matematički fakultet  
e-mail: [milos.stankovic@fondmt.rs](mailto:milos.stankovic@fondmt.rs)

**Abstrakt:** U radu je razmatran generalizovani problem pridruživanja (*Generalized Assignment Problem - GAP*), poznati *NP*-teški problem kombinatorne optimizacije čije je rešavanje od velikog praktičnog značaja. *GAP* podrazumeva pridruživanje skupa poslova skupu agenata koji ih izvršavaju. Svaki agent ima ograničene resurse i svaki posao se mora dodeliti tačno jednom agentu zauzimajući određenu količinu njegovih resursa. U ovom radu, za rešavanje *GAP*-a se primenjuje metaheuristika tabu pretraživanja (*Tabu Search - TS*). Algoritam tabu pretraživanja zasniva se na adaptivnoj memoriji koja u procesu pretrage ima ulogu u prevazilaženju lokalnih optimuma i približavanju globalnom optimumu. U adaptivnoj memoriji se čuvaju informacije o procesu pretrage i na osnovu njih se izvršavaju faze intenzifikacije i diversifikacije. Tokom intenzifikacije, algoritam pokušava da na osnovu informacija iz adaptivne generiše dobro rešenje, dok se tokom diversifikacije pretražuju slabo posećeni delovi pretraživackog prostora. Najznačajnije karakteristike ovog algoritma su jednostavnost i fleksibilnost. U radu je detaljno prikazana implementacija predloženog algoritma i njegovih komponenti, kao i analiza dobijenih eksperimentalnih rezultata.

**Ključne reči:** Tabu pretraživanje, Generalizovani problem pridruživanja, heuristika, metaheuristika, kombinatorna optimizacija.

## 1. Uvod

U ovom radu razmatra se poznati problem kombinatorne optimizacije – generalizovani problem pridruživanja (*Generalized Assignment Problem – GAP*). Cilj ovog problema je raspodeliti  $n$  poslova između  $m$  agenata ( $n \geq m$ ) tako da troškovi pridruživanja i izvršavanja tih poslova budu minimalni. Druga varijanta ovog problema podrazumeva raspoređivanje poslova tako da profit bude maksimalan. Agenti imaju (različite) kapacitete izražene jedinicama resursa i svaki posao mora biti dodeljen tačno jednom agentu. Jednom agentu može biti dodeljen veći broj poslova, pri čemu se mora voditi računa da to dodeljivanje zadovoljava kapacitete agenata, dok jedan posao može biti izvršavan od strane samo jednog agenta. Pošto je reč o *NP*-teškom problemu, u njegovom rešavanju veliki značaj imaju heuristički algoritmi.

Na osnovu svojih prethodnih radova, *Fred Glover* je 1986. godine predložio novi algoritam koji je nazvao Tabu pretraživanje (*Tabu Search – TS*) [5, 6]. Slični pristup nazvan najstrmiji uspon / najblaži pad (*Steepest ascent / mildest descent*) predložio je *Hansen* iste godine [7]. Devedesetih godina XX veka *TS* postaje veoma popularan algoritam za rešavanje optimizacijskih problema. Danas je to jedna od najrasprostranjenijih pretraživačkih metaheuristika. Ključnu ulogu u radu *TS*-a ima adaptivna memorija. U njoj se čuvaju informacije vezane za pretragu u cilju prevazilaženja lokalnih optimuma i približavanja globalnom optimumu kao i izbegavanja ciklusa u procesu pretrage. Zbog ovih osobina *TS* se povezuje sa veštačkom inteligencijom [3].

U radovima koji se bave *GAP*-om mogu se sresti i heurističke i egzaktne metode. U [11] može se naći pregled nekih ranije objavljenih radova u kojima se razmatra ovaj problem. *TS* metoda se koristi u [1] gde se *GAP* relaksira odbacivanjem uslova ograničenja koja se odnose na kapacitete agenata a sva prekoračenja se dodatno naplaćuju tako što se vrednost prekoračenja množi koeficijentom koji se dinamički menja. U poslednje vreme za rešavanje *GAP*-a sve se više koriste novije metaheuristike, kao što je optimizacija zasnovana na ponašanju pčela (*Bee Colony Optimization – BCO*). Detaljan prikaz algoritma i dobijenih rezultata može se naći u [8]. Takođe, koriste se i hibridne metode. U [10] se *GAP* najpre relaksira tako što se binarne promenljive

proglase za realne a zatim se TS metoda hibridizuje sa egzaktnom metodom grananja i ograničavanja (*Branch & Bound* – BB) tako što se ove dve metode naizmenično smenjuju u izvršavanju.

Rad je koncipiran na sledeći način. U poglavlju 2 dat je detaljan opis problema, na koji način se predstavlja rešenje u računaru, koje su moguće okoline rešenja i koja se okolina koristi u radu. Takođe, razmatra se i problem početnog rešenja. Poglavlje 3 sadrži opis TS metode sa svim svojim komponentama kao i opis implementacije tih komponenti u radu. Poglavlje 4 bavi se rezultatima algoritma.

## 2. Generalizovani problem pridruživanja

Neka je  $I = \{1, 2, \dots, m\}$  skup agenata, a  $J = \{1, 2, \dots, n\}$  skup poslova. GAP može biti formulisan kao problem celobrojnog programiranja na sledeći način:

$$\min z = \sum_{i=1}^m \sum_{j=1}^n c_{ij} x_{ij} \quad (1)$$

$$\text{pri ograničenjima} \quad \sum_{j=1}^n a_{ij} x_{ij} \leq b_i \quad \forall i \in I, \quad (2)$$

$$\sum_{i=1}^m x_{ij} = 1 \quad \forall j \in J, \quad (3)$$

$$x_{ij} \in \{0, 1\} \quad \forall i \in I, \quad \forall j \in J, \quad (4)$$

gde  $c_{ij}$  predstavlja cenu pridruživanja agenta  $i$  poslu  $j$ ,  $a_{ij}$  predstavlja količinu resursa koju agent  $i$  troši na izvršavanje posla  $j$  i  $b_i$  predstavlja kapacitet agenta  $i$ . U nekim varijantama GAP-a količina resursa za izvršavanje posla  $j$  ista je za sve agente tako da se  $a_{ij}$  zamenjuje sa  $a_j$ . Ograničenje (2) obezbeđuje da kapaciteti agenata ne budu prekoračeni, ograničenje (3) omogućuje da svaki posao bude dodeljen tačno jednom agentu dok se uslov (4) odnosi na binarnu prirodu promenljivih  $x_{ij}$ , pri čemu  $x_{ij}$  uzima vrednost 1 ako je posao  $j$  dodeljen agentu  $i$ , 0 inače.

Brojni problemi iz realnog života mogu biti formulisani kao GAP. Na primer, dizajn komunikacionih mreža sa ograničenim kapacitetima u svakom čvoru, pridruživanje poslova računarima u računarskim mrežama, raspoređivanje procesa procesorima kod računara sa više procesora i sl. Takođe, GAP se javlja kao podproblem u mnogim drugim problemima, na primer kod rutiranja vozila, raspoređivanja kapaciteta, lokacijskih problema, itd. GAP je NP-težak problem [2, 9], tako da sa porastom vrednosti  $m$  i  $n$  njegovo egzaktno rešavanje postaje znatno složenije, a u pojedinim slučajevima i nemoguće. Zbog toga raste značaj heurističkih algoritama za rešavanje GAP-a i drugih kombinatornih problema.

### 2.1. Predstavljanje rešenja i njegove okoline

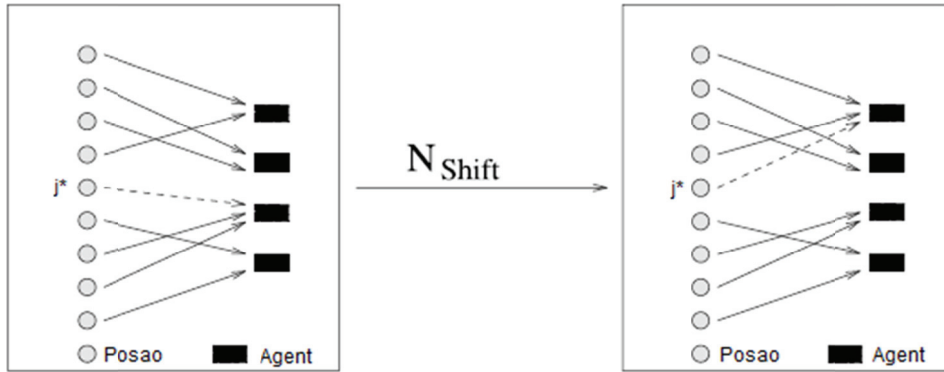
Matematički gledano, rešenje GAP-a je matrica  $[x_{ij}]$  dimenzije  $m \times n$  čiji su elementi iz skupa  $\{0, 1\}$ . Ako je  $x_{ij} = 1$  to znači da je posao  $j$  pridružen agentu  $i$ . Ovakvo rešenje se može predstaviti i kao  $n$ -dimenzioni vektor  $\pi = (\pi_1, \pi_2, \dots, \pi_n)$ , gde je na  $j$ -tom mestu redni broj agenta kome je pridružen posao  $j$ , tj.  $\forall \pi_j \in I \quad \pi_j = i \Leftrightarrow x_{ij} = 1$ . Za potrebe računara rešenje se predstavlja kao  $n$ -dimenzioni niz čiji su elementi iz skupa  $\{0, 1, \dots, m - 1\}$ .

Neka je sa  $S$  označen skup svih dopustivih rešenja. Za svako rešenje  $\pi$  definiše se njegova okolina  $S_\pi \subseteq S$  kao skup svih rešenja iz  $S$  koja su na neki način "blizu" rešenju  $\pi$ . Okolina nekog rešenja se može definisati na razne načine. Jedna od najčešće korišćenih okolina je okolina promene ( $N_{Shift}$  okolina), koja se još naziva i 1-okolina i definiše se na sledeći način. Neka je dato neko



rešenje  $\pi = (\pi_1, \pi_2, \dots, \pi_n)$ . Tada se pod  $N_{Shift}$  okolinom (slika 1) podrazumeva skup svih rešenja koja se od rešenja  $\pi$  razlikuju u samo jednoj koordinati, tj:

$$N_{Shift}(\pi) = \{(\pi'_1, \pi'_2, \dots, \pi'_n) : \exists j^* \in J \text{ tako da } \pi'_{j^*} \neq \pi_{j^*} \text{ i } \forall j \neq j^* \pi'_j = \pi_j\}.$$



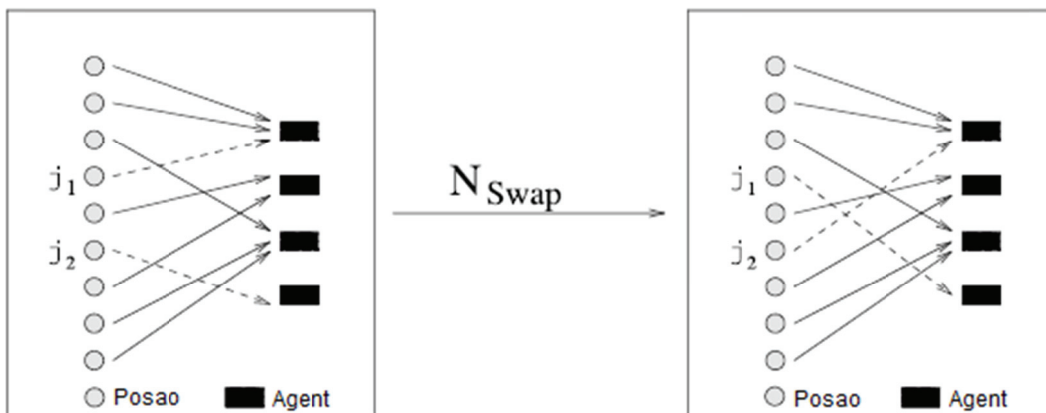
Slika 1. Okolina promene ( $N_{Shift}$  okolina)

Promena agenta poslu  $j$  naziva se potez promene, pa se  $N_{Shift}$  okolina rešenja  $\pi$  dobija primenom svih poteza promene na rešenje  $\pi$ . Druga veoma često korišćena okolina jeste okolina zamene ( $N_{Swap}$  okolina) koja se dobija tako što u rešenju  $\pi = (\pi_1, \pi_2, \dots, \pi_n)$  dva posla međusobno razmene dodeljene agente (slika 2), što se može zapisati na sledeći način:

$$N_{Swap}(\pi) = \{(\pi'_1, \pi'_2, \dots, \pi'_n) : \exists j_1, j_2 \in J, \pi_{j_1} \neq \pi_{j_2} \text{ tako da } \pi'_{j_1} = \pi_{j_2}, \pi'_{j_2} = \pi_{j_1}, \forall j \neq j_1, j_2 \pi'_j = \pi_j\}.$$

Zamena agenata između dva posla naziva se potez zamene, a  $N_{Swap}$  okolina rešenja  $\pi$  dobija primenom svih poteza zamene na rešenje  $\pi$ . Okolina  $N_{Swap}(\pi)$  se sastoji od rešenja koja su iste strukture kao i rešenje  $\pi$ , tj. broj poslova pridružen svakom agentu ostaje isti.

U slučaju jednostavne okoline TS neće dati željene rezultate. Zbog toga se u radu koristi okolina koja predstavlja uniju navedenih okolina, tj.  $N(\pi) = N_{Shift}(\pi) \cup N_{Swap}(\pi)$ . Na ovaj način dobija se na raznovrsnosti rešenja u toku pretrage.



Slika 2. Okolina zamene ( $N_{Swap}$  okolina)

## 2.2. Početno rešenje i relaksirani GAP

Kao i u mnogim drugim heuristikama i ovde se javlja problem početnog rešenja. Jedan od načina da se dođe do početnog rešenja jeste slučajno biranje elemenata niza koji predstavlja rešenje. Takvo rešenje je najčešće nedopustivo. Ako bi se ovaj postupak nastavio sve do prvog dopustivog rešenja, vreme izvršavanja programa bi znatno poraslo. Kod instanci problema velikih dimenzija generisanje dopustivog rešenja na slučajan način bilo bi gotovo nemoguće, pa samim tim bi vreme izvršavanja težilo ka beskonačnosti.

U radu se koristi sledeći pristup za generisanje početnog rešenja. Najpre, može se primetiti da su elementi matrice cena  $[c_{ij}]$  i matrice kapaciteta  $[a_{ij}]$  obrnuto proporcionalni, tj. kako raste cena izvršavanja posla  $j$  od strane agenta  $i$ , tako se smanjuje količina resursa koje posao  $j$  zauzima agentu  $i$ . Zato se na početku izvršavanja programa generiše matrica proizvoda  $[p_{ij}]$  istih dimenzija kao i matrica  $[c_{ij}]$  (odnosno  $[a_{ij}]$ ) tako da za njene elemente važi  $p_{ij} = a_{ij} \cdot c_{ij}$ . Prilikom generisanja početnog rešenja razmatra se matrica  $[p_{ij}]$  i  $j$ -tom poslu se dodeljuje agent  $k$  takav da je  $p_{kj} = \min_{i \in I} p_{ij}$ . Dakle, u koloni  $j$  matrice  $[p_{ij}]$  nađe se najmanji element i redni broj vrste tog elementa određuje agenta koji se pridružuje poslu  $j$ . Dodeljivanje agenata na ovaj način može da dovede do prekoračenja kapaciteta agenata. Zbog toga se kod svakog dodeljivanja proverava da li ono zadovoljava kapacitet određenog agenta. U slučaju da ne zadovoljava onda se traži sledeći po veličini element matrice  $[p_{ij}]$  u koloni  $j$ , i redni broj te kolone predstavlja redni broj agenta koji treba dodeliti poslu  $j$ , itd. Zbog zauzetosti svih agenata, može se desiti da se posao  $j$  ne može dodeliti ni jednom agentu. Tada se poslu  $j$  ipak mora dodeliti neki agent, pa se u tom slučaju uzima poslednji izračunati agent.

Ovakav algoritam za generisanje početnog rešenja može proizvesti nedopustivo rešenje. Međutim, to rešenje se sastoji od elemenata koji su intuitivno "dobri", pa je pretpostavka da će se tokom pretrage vrlo brzo doći do veoma dobrog dopustivog rešenja. Zbog toga se to početno rešenje ne odbacuje, ali je zato potrebno modifikovati početni problem. Početni problem GAP se u radu modifikuje relaksacijom, tj. odbacivanjem ograničenja za kapacitete agenata, i tako se dobija relaksirani GAP – RGAP. Na taj način ne pravi se razlika između dopustivih i nedopustivih rešenja. Da bi se tokom pretrage izbegla nedopustiva rešenja potrebno je modifikovati i funkciju cilja, što je urađeno na sledeći način:

$$z = \sum_{i=1}^m \sum_{j=1}^n c_{ij} x_{ij} + M * excess.$$

U datom izrazu  $M$  predstavlja veliki pozitivan broj, dok je  $excess$  ukupno prekoračenje kapaciteta svih agenata i računa se na sledeći način:  $excess = \sum_{i=1}^m excess_i$ , gde je  $excess_i = \max\{\sum_{j=1}^n a_{ij} x_{ij} - b_i, 0\}$ . Ako je rešenje dopustivo, onda će biti  $\forall i \in I \quad excess_i = 0$ , a samim tim i  $excess = 0$ , pa se vrednost funkcije cilja neće menjati. Ako je rešenje nedopustivo, sva prekoračenja se dodatno naplaćuju i funkcija cilja se uvećava za tu vrednost, pa takvo rešenje postaje jako skupo i vrlo verovatno neće biti razmatrano u procesu pretrage. Eksperimentalno se pokazuje da ako je početno rešenje nedopustivo, proces pretrage vrlo brzo napušta nedopustivi prostor upravo zbog skupog naplaćivanja prekoračenja.

U opštem slučaju ideja relaksacije podrazumeva odbacivanje nekog od (veoma restriktivnih) uslova ograničenja u cilju svođenja problema na novi problem kojeg je lakše rešiti. Pomenuta relaksacija GAP-a koristi se i u [1] gde se veličina  $M$  dinamički menja u toku izvršavanja programa, dok se u [10] koristi linearna relaksacija, tj. binarne promenljive se posmatraju kao realne.

### 3. Tabu pretraživanje

Tabu pretraživanje tokom pretrage prelazi u lošije rešenje od tekućeg kada u tekućoj okolini ne postoji bolje rešenje. Cilj prelaska u lošije rešenje je prevazilaženje lokalnih optimuma. U procesu pretrage deterministički se pretražuje cela okolina tekućeg rešenja i pronalazi se najbolje rešenje iz te okoline. Ukoliko je novo rešenje bolje, pretraga se nastavlja u okolini tog rešenja. U suprotnom slučaju pretraga je dospela do lokalnog optimuma i proces pretrage se nastavlja izborom lošijeg rešenja i prelaskom u njegovu okolinu. Zatim se u novoj okolini pronalazi najbolje rešenje i prelazi se u njega čak iako to rešenje nije bolje od najboljeg rešenja dostignutog tokom pretrage. Ovakav način pretrage može generisati cikluse, tj. već posećena rešenja mogu ponovo biti izabrana.

#### 3.1. Kratkoročna memorija

Tokom svog rada TS algoritam koristi više različitih memorija. Da bi se izbegli ciklusi, TS prilikom pretrage okoline odbacuje rešenja koja su u prošlosti bila izabrana. Ovo je moguće zbog toga što se tokom pretrage pamti put (trajektorija) pretrage koji se smešta u tabu listu. Tabu lista predstavlja takozvanu kratkoročnu memoriju (*Short-Term Memory*) i sastoji se od elemenata koji se nazivaju tabui. U svakoj iteraciji TS-a tabu lista se ažurira, tj. na kraju liste se smeštaju informacije o poslednjem rešenju. Čuvanje kompletnih rešenja (kompletno rešenje je tabu) nije mnogo efikasno jer predstavlja trošenje kako prostora tako i vremena. U svakoj iteraciji neophodno je proveriti da li novo rešenje (najbolje rešenje u okolini tekućeg rešenja) ne pripada listi svih posećenih rešenja. Zbog toga se za tabu uzima inverzni potez poteza koji vodi od tekućeg do novog rešenja, tj. potez kojim bi se proces pretrage vratio u već posećeno rešenje. Takav potez postaje zabranjen u narednim iteracijama TS-a. Takođe, javlja se problem dužine liste. Ukoliko bi lista bila neograničena vrlo brzo bi postala predugačka pa bi ispitivanje da li je neki potez tabu zahtevalo pretragu kompletne tabu liste, što je veoma sporo. Zato se za dužinu liste najčešće uzima konstanta vrednost koja se naziva tabu mandat (*tabu tenure*). Ukoliko je tabu lista puna, a potrebno je dodati novi tabu potez, najpre se iz tabu liste briše najstariji tabu pa se tek onda na kraju liste dodaje novi tabu potez. Dužina liste je veoma bitan parametar koji se mora pažljivo odrediti. Prekratka tabu lista može omogućiti stvaranje ciklusa čija je dužina veća od dužine liste, što nije poželjno. Takođe, predugačka tabu lista je previše restriktivna.

Na primer, ako se rešenje predstavlja kao niz dužine  $n$  i ako je prilikom prelaska iz tekućeg u novo rešenje potrebno promeniti vrednost  $j$ -tog elementa niza sa  $i$  na  $k$ , tada se tabu može predstaviti kao uređeni par  $(j, i)$ . To znači da će potez koji  $j$ -ti element postavlja na vrednost  $i$  biti zabranjen. Međutim, ovo nije jedini način predstavljanja zabranjenih poteza. Mnogo restriktivniji način jeste da se umesto čuvanja uređenog para  $(j, i)$  čuva samo vrednost  $j$ , što bi značilo da su svi potezi koji menjaju  $j$ -ti element niza zabranjeni. Kao što se može zaključiti, postoje različiti načini predstavljanja zabranjenih poteza.

U radu se kao i u prethodnom primeru za tabue uzimaju uređeni parovi  $(j, i)$  koji se smeštaju u listu dužine  $L$ , s tim što se zabranjuju samo potezi promene. Zabranjivanje i poteza zamene deluje previše restriktivno pa se ti potezi ne zabranjuju.

#### 3.2. Kriterijum aspiracije

Zabranjeni potezi vrlo često ne vode ka prethodno posećenom rešenju, već ka novom boljem rešenju. Poštovanjem zabranjenih poteza mogu se odbaciti jako dobra rešenja, jer tabu lista može biti veoma restriktivna. Zbog toga se TS algoritam proširuje mehanizmom koji može da poništi tabu status nekog poteza. Taj mehanizam se naziva kriterijum aspiracije (*Aspiration Criteria*) i služi da smanji restriktivnost tabu liste. Ukoliko se primenom tabu poteza dobija rešenje koje je bolje od

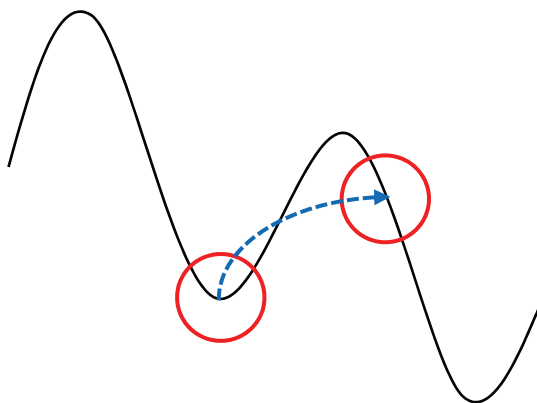
najboljeg rešenja dobijenog do tog trenutka, tom potezu se poništava tabu status. Za poteze kojima je poništen tabu status kaže se da zadovoljavaju kriterijum aspiracije. Naravno, tabu status nekog poteza se može poništiti i u raznim drugim situacijama koje zavise od problema do problema. Iako se kriterijum aspiracije može definisati na mnogo složenije načine, najčešće se definiše tako da dopušta one zabranjene poteze koji vode ka boljem rešenju od najboljeg rešenja nadjenog do tog trenutka. U radu se na isit način definiše kriterijum aspiracije.

Ako je dato rešenje  $\pi$  i ako je  $N(\pi)$  označena okolina rešenja  $\pi$ , tada se može definisati i dopustiva okolina rešenja  $\pi$ , u oznaci  $N'(\pi)$ , kao skup svih rešenja iz okoline  $\pi$  koja nisu zabranjena (nisu tabu) ili zadovoljavaju kriterijum aspiracije.

### 3.3. Dugoročna memorija

Problem kod svih heuristika zasnovanih na pretraživanju okolina rešenja, pa i TS algoritma, jeste da su one sklone da budu previše "lokalne", tj. da su sklone da najveći deo svog vremena troše na pretraživanje ograničenog dela pretraživačkog prostora. Negativne posledice ove činjenice su da ove heuristike mogu zaobići zanimljive delove pretraživačkog prostora i tako završiti sa rešenjima koja su lokalni optimumi i koja su daleko od globalnog optimuma. TS algoritam poseduje mehanizme za prevazilaženje pomenutih problema koji koriste informacije o procesu pretrage od samog početka. Takve informacije se smeštaju u takozvanoj dugoročnoj memoriji i u njoj se čuva za svaku komponentu rešenja ukupni broj iteracija u kojima je ta komponenta bila prisutna u tekućem rešenju. Otuda i drugi naziv – frekventna memorija. Podaci iz dugoročne memorije koriste se u fazi diversifikacije i intenzifikacije rešenja.

Diversifikacija je mehanizam kojim TS heuristika pokušava da prevaziđe lokalni optimum tako što odvodi proces pretrage ka neistraženim delovima pretraživačkog prostora. Na osnovu frekventne memorije nalaze se komponente rešenja koje su bile najmanje prisutne u tekucem rešenju i uključuju se u sledeće rešenje. Na taj način TS prelazi iz lokalnog optimuma u neko lošije rešenje (slika 3) i iz tog dela pretraživačkog prostora pokušava dostići globalni optimum. U radu se dugoročna memorija predstavlja kao matrica  $T$  dimenzije  $m \times n$  tako da element na mestu  $(i, j)$  predstavlja broj iteracija u kojima je posao  $j$  bio dodeljen agentu  $i$ . Ovakva reprezentacija dugoročne memorije omogućuje njeno veoma jednostavno ažuriranje koje se svodi na prolazak kroz niz (koji predstavlja rešenje) i uvećenje odgovarajućih elemenata matrice  $T$ . Diversifikacija je implementirana na sledeći način. Slučajno se bira  $k$  poslova i njima se dodeljuju agenti koji su tokom pretrage najmanji broj iteracija bili dodeljeni tim poslovima. Ovakva implementacija zavisi od parametra  $k$  koji treba pažljivo odabrati. Ukoliko je  $k$  mali broj, izvršavanje diversifikacije je brzo, međutim u tom slučaju je mala raznovrsnost rešenja koja se mogu tako dobiti, a samim tim se smanjuje verovatnoća odlaska u rešenje koje će proces pretrage dovesti do globalnog optimuma. Obrnuto važi za veliko  $k$ . U radu se za  $k$  uzima vrednost jednaka četvrtini broja poslova i do nje se došlo eksperimentalnim putem.



Slika 3. Mehanizam diversifikacije za izbegavanje lokalnih optimuma

Intenzifikacija je procedura kojom TS pokušava da oponaša postupke čoveka. Intenzifikacijom se proces pretrage navodi na delove pretraživačkog prostora koji deluju obećavajuće. S vremena na vreme pretraga se zaustavlja i na osnovu podataka iz dugoročne memorije generiše se novo rešenje koje u sebi sadrži komponente najčešće prisutne u tekućim rešenjima. Ove komponente se fiksiraju, jer se na osnovu broja ponavljanja može pretpostaviti da će se sadržati u rešenju koje predstavlja globalni optimum. Intenzifikacija je u radu implementirana kao procedura koja generiše novo rešenje od komponenti koje su bile prisutne u više od 85% ukupnog broja iteracija. Ove komponente se fiksiraju što predstavlja smanjenje dimenzije problema. U nastavku rada program pokušava da nađe najbolje vrednosti za komponente koje nisu fiksirane. Eksperimentalno se pokazuje da ako se navedeni procenat smanji veliki broj komponenti postaje fiksiran, pa intenzifikacija neće dati očekivane rezultate. U slučaju većeg procenta, mali broj elemenata rešenja će biti fiksiran, pa intenzifikacija neće imati efekta.

Važno je napomenuti da se u određenom broju implementacija TS algoritma za potrebe intenzifikacije koristi posebna memorija, tzv. srednjeročna memorija (*Medium-Term Memory, Recency Memory*). Za svaku komponentu rešenja i svaku poziciju u rešenju ova memorija čuva maksimalan broj uzastopnih iteracija u kojima je ta komponenta bila prisutna u tekućem rešenju na datoj poziciji. Komponente rešenja najčešće prisutne na određenim pozicijama se smatraju dobrim i one se fiksiraju. Zbog jednostavnosti u radu se koristi dugoročna memorija za potrebe intenzifikacije, a slično je urađeno i u [1, 10].

### 3.4. Shema algoritma tabu pretraživanja

Sledeći pseudo kod opisuje TS algoritam:

```

TabuSearch(){
     $x = x_0$ ;
     $x^* = x_0$ ;
    while(nije KriterijumZaustavljanja){
        Neka je  $N'(x) \subseteq N(x)$  skup svih rešenja iz okoline rešenja  $x$  koja nisu
        tabu ili zadovoljavaju kriterijum aspiracije;
        Naći najbolje rešenje  $x' \in N'(x)$ ;
         $x = x'$ ;
        Ažurirati kratkoročnu i dugoročnu memoriju;
        if( $x'$  bolje od  $x^*$ )
             $x^* = x'$ ;
        if(KriterijumDiverzifikacije)
            Diverzifikacija();
        if(KriterijumIntenzifikacije)
            Intenzifikacija();
    }
}

```

Sa  $x_0$  je označeno početno rešenje. Kriterijum zaustavljanja je zadati broj uzastopnih iteracija bez poboljšanja u odnosu na najbolje nađeno rešenje. Deo algoritma od početka *while* petlje do *if* izraza koji proverava kriterijum diversifikacije naziva se faza kratkoročne memorije (*Short-Term Memory Phase*). U toj fazi se vrši pretraga okoline tekućeg rešenja i pronalazi najbolje rešenje koje nije tabu ili zadovoljava kriterijum aspiracije. Tako dobijeno rešenje postaje novo tekuće rešenje. Ova faza se ponavlja dokle god ima poboljšanja, tj. kriterijumi diversifikacije i intenzifikacije imaju vrednost '*netačno*'. Kada se u fazi kratkoročne memorije dobije rešenje bez poboljšanja, to znači da je pretraga došla u lokalni optimum. Tada se prelazi se na fazu diversifikacije ili fazu intenzifikacije. U radu je imlementirano naizmenično izvršavanje ovih faza. Posle faze kratkoročne memorije izvršava se faza diversifikacije nakon koje ponovo sledi faza kratkoročne memorije. Kada se u fazi kratkoročne memorije ponovo dođe do rešenja koje nije bolje od tekućeg, tada nastupa faza intenzifikacije, itd.

#### 4. Eksperimentalni rezultati

Za testiranje predloženog algoritma korišćene su instance dostupne na <http://people.brunel.ac.uk/~mastjib/jeb/orlib/gapinfo.html>. Za testiranje korišćen je računar Asus K53SV-SX471 sa procesorom Intel Core i7 – 2630QM 2.0 GHz (up to 2.9 GHz) sa 4 jezgra i 6 GB RAM memorije. Algoritam je implementiran u programskom jeziku C#. Testiranje svake instance ponovljeno je 20 puta i dobijeni su rezultati prikazani u tabeli 1.

U prikazanoj tabeli 1. kolona označena sa  $O$  odnosi se na teorijske optimalne vrednosti funkcije cilja. Sa  $f_{best}$  označena je najbolja vrednost funkcije cilja od 20 vrednosti dobijenih testiranjem.  $t_{best}$  označava vreme kada algoritam prvi put dođe u najbolju vrednost koju može da dostigne, i računa se na sledeći način:

$$t_{best_p} = \sum_{i=1}^{20} t_{best_{pi}} / 20,$$

gde je  $t_{best_p}$  vreme koje se odnosi na  $p$ -tu instancu, a  $t_{best_{pi}}$  vreme  $p$ -te instance u  $i$ -tom izvršavanju. Sa  $t_{total}$  označeno je ukupno vreme izvršavanja programa i ono se računa na sledeći način:

$$t_{total_p} = \sum_{i=1}^{20} t_{total_{pi}} / 20,$$

a oznake  $t_{total_p}$  i  $t_{total_{pi}}$  imaju isto značenje kao  $t_{best_p}$  i  $t_{best_{pi}}$ . Vreme je izraženo u milisekundama. Kvalitet rešenja u svih 20 izvršavanja meri se relativnom greškom rešenja izraženom u procentima (*prg*), koja prikazuje odnos dobijenog rešenja i optimalnog rešenja. Procentualna greška rešenja se računa po formuli:

$$prg_p = \frac{1}{20} \sum_{i=1}^{20} rg_{pi},$$

pri čemu se sa  $rg_{pi}$  označava relativna greška u  $i$ -tom izvršavanju  $p$ -te instance. U situacijama kada je vrednost  $O_p$  poznata,  $rg_{pi}$  se izračunava na osnovu formule:

$$rg_{pi} = 100 \cdot \frac{TS_{pi} - O_p}{O_p},$$

gde je sa  $TS_{pi}$  označena vrednost funkcije cilja rešenja  $p$ -te instance u  $i$ -tom izvršavanju. Kada  $O_p$  nije poznato,  $rg_{pi}$  se izračunava na osnovu formule:  $rg_{pi} = 100 \cdot \frac{TS_{pi} - NDS}{NDS}$ , gde je sa  $NDS$  (najbolje do sada) označeno najbolje postignuto rešenje u svim izvršavanjima. Kvalitet rešenja ocenjuje se i sa  $\sigma$ , odnosno standardnom devijacijom relativne greške za  $rg_{pi}$  koja se dobija iz formule:

$$\sigma_p = \sqrt{\frac{1}{20} \sum_{i=1}^{20} (rg_{pi} - prg_p)^2}.$$

Tabela 1. Rezultati testiranja

	$m$	$n$	$O$	$f_{best}$	$t_{best}$	$t_{total}$	$\sigma$
<i>gap1</i>	5	15	261	261	0.5	1.0	0.0
<i>gap2</i>	5	20	277	277	2.65	6.15	0.98
<i>gap3</i>	5	25	438	438	4.25	11.05	0.28
<i>gap4</i>	5	30	406	406	7.15	19.6	0.56
<i>gap5</i>	8	24	403	403	4.3	13.3	0.47
<i>gap6</i>	8	32	525	525	9.4	32.2	0.35
<i>gap7</i>	8	40	646	651	29.7	85.45	0.28
<i>gap8</i>	8	48	797	800	65.7	169.1	0.36
<i>gap9</i>	10	30	482	483	15.75	36.05	0.25
<i>gap10</i>	10	40	638	638	33.45	92.1	0.42
<i>gap11</i>	10	50	573	574	76.4	206.5	0.51
<i>gap12</i>	10	60	974	977	169.15	452.25	0.26
<i>gapa1</i>	5	100	1698	1698	72.45	714.5	0.0
<i>gapa2</i>	5	200	3235	3235	91.8	8442.95	0.0
<i>gapa3</i>	10	100	1360	1360	526.65	1770.2	0.04
<i>gapd4</i>	10	200	---	12834	27640.2	55378.95	0.18
<i>gapd5</i>	20	100	---	6460	2125.35	5068.85	0.25

Iz tabele 1. prvo što se može primetiti jeste da je algoritam u većini slučajeva dosegao teorijski optimum. U slučajevima gde nije tako, algoritam je bio veoma blizu teorijskog optimuma. Zatim, na osnovu kolone  $t_{best}$  primećuje se da algoritam vrlo brzo dolazi do najboljeg posećenog rešenja. Razlog za to leži u efikasnosti metode za konstrukciju početnog rešenja.

Na osnovu testiranih instanci može se videti da tačnost algoritma ne opada sa porastom dimenzija problema. U pojedinim instancama (*gapa1*, *gapa2*) algoritam je bio veoma precizan bez obzira na njihove (relativno) velike dimenzije. Zbog toga, može se pretpostaviti da su rešenja poslednje dve instance veoma bliska teorijskom optimumu, mada se ova tvrdnja ne može dokazati.

## 5. Zaključak

U ovom radu predložena je metaheuristika tabu pretraživanja za rešavanje generalizovanog problema pridruživanja. Prilikom prilagođavanja problema za predloženu metaheuristiku, izvršena je relaksacija problema, čime je omogućeno razmatranje i nedopustivih rešenja. Dodatnim naplaćivanjem prekoračenja takva rešenja postaju jako skupa. Diversifikacija i intenzifikacija TS algoritma zasnovane su na dugoročnoj (frekventnoj) memoriji. Diversifikacijom se napušta lokalni optimum, dok se intenzifikacijom smanjuje dimenzija problema. Parametri i performanse algoritma testirane su na javno dostupnim instancama i prikazani su postignuti rezultati. Iz rezultata može se zaključiti da je algoritam vrlo brzo dostizao najbolje posećeno rešenje. Tačnost algoritma ne zavisi od dimenzije testiranih instanci.

## 6. Literatura

- [1] **J.A. Diaz, E. Fernandez.** A Tabu search heuristic for the generalized assignment problem, *European Journal of Operation Research*, 2001, 132, 22-38.
- [2] **M.L. Fisher, R. Jaikumar, L.N. Van Wassenhove.** A multiplier adjustment method for the generalized assignment problem, *Management Science*, 1986, 32, 1095–1103.
- [3] **M.Gendreau, J.Y. Potvin.** *Handbook of Metaheuristics (Second Edition)*, Springer New York Dordrecht Heidelberg London, 2010.
- [4] **El-Ghazali Talbi.** *Metaheuristics – From design to implementation*, John Wiley & Sons, Inc., Hoboken, New Jersey, 2009.
- [5] **F. Glover.** Tabu search: Part 1, *ORSA Journal of Computing* 1, 1989, 190-206.
- [6] **F. Glover.** Tabu search: Part 2, *ORSA Journal of Computing* 2, 1990, 4-32.
- [7] **P. Hansen.** *The steepest ascent mildest descent heuristic for combinatorial programming*, Congress on Numerical Methods in Combinatorial Optimization, Capri, Italy, 1986.
- [8] **L.Ozbakir, A.Baykasoglu, P.Tapkan.** *Bees algorithm for generalized assignment problem*, Applied Mathematics and Computation, 2010, 215, 3782–3795.
- [9] **S. Sahni, T. Gonzalez.** *P-complete approximation problems*, *Journal of ACM*, 1976, 23, 555–565.
- [10] **A.J. Woodcock, J.M. Wilson.** *A hybrid tabu search/branch & bound approach to solving the generalized assignment problem*, *European Journal of Operation Research*, 2010, 207, 566–578.
- [11] **M. Yagiura, T. Ibaraki.** *Recent metaheuristics for the generalized assignment problem*, Proceedings of the 12th International Conference on Informatics Research for Development of Knowledge Society Infrastructure (IKS'04). IEEE, 2004.



# HIBRIDNI GENETSKI ALGORITAM ZA REŠAVANJE HAB LOKACIJSKOG PROBLEMA NEOGRANIČENIH KAPACITETA SA VIŠESTRUKIM ALOKACIJAMA

Maja Đukić

Matematički fakultet, Univerzitet u Beogradu  
e-mail: mdjukic85@gmail.com

**Apstrakt:** U ovom radu je opisan genetski algoritam za rešavanje hab lokacijskog problema neograničenih kapaciteta sa višestrukim alokacijama. Ovaj NP-težak problem ima značajnu primenu u praksi. Hab lokacijski problemi se dosta koriste u modernim transportnim i telekomunikacijskim sistemima. Najčešće se primenjuju za dizajniranje drumskih i železničkih sistema, poštanskih sistema, sistema brze isporuke i slično. U cilju poboljšanja efikasnosti predloženog genetskog algoritma, primenjena je hibridizacija genetskog algoritma sa heuristikom lokalnog pretraživanja, pa su tako nastale metode koje su veoma uspešne i pri rešavanju problema velikih dimenzija. U radu su data rešenja i za probleme velikih dimenzija ( $n = 130, 200$ ) za koje optimalna rešenja nisu poznata. Genetski algoritam je testiran na instancama čije optimalno rešenje nije poznato, ali dao je rezultate za koje se može pretpostaviti da su kvalitetni.

**Ključne reči:** genetski algoritmi, hab lokacijski problemi, kombinatorna optimizacija, metaheuristike

## 1. Uvod

### 1.1. Genetski algoritmi

Klasične heuristike razvijane su u cilju rešavanja pojedinačnih problema. Međutim, nastale su nove „moderne“ heuristike tj. metaheuristike koje se mogu primeniti na veliki broj zadataka iz različitih oblasti. Genetski algoritmi spadaju među najpoznatije metaheuristike. Ovakvi algoritmi su prvobitno kreirani da simuliraju proces genetske evolucije jedne populacije jedinki pod dejstvom okruženja i genetskih operatora. One jedinke iz populacije koje su u većoj meri prilagođene okruženju, međusobno se dalje reprodukuju i tako se stvara nova generacija jedinki. Ovaj proces se ponavlja pri čemu se iz generacije u generaciju prosečna prilagođenost članova populacije povećava. Čitav postupak se zaustavlja nakon zadovoljenja jednog ili više kriterijuma zaustavljanja. Najbolji član trenutne populacije predstavlja rešenje genetskog algoritma. Danas se genetski algoritmi koriste za rešavanje široke klase problema kombinatorne optimizacije.

Svaka jedinka u populaciji je predstavljena genetskim kodom nad određenom konačnom azbukom. Najčešće se koristi binarno kodiranje, gde se genetski kod sastoji od niza bitova. Početna populacija je obično slučajno generisana, što obezbeđuje raznovrsnost genetskog materijala. Svakoј jedinki populacije (u praksi ih je najviše do nekoliko stotina) se dodeljuje *funkcija prilagođenosti* ili *funkcija cilja (fitness function)* koja određuje kvalitet date jedinke tj. odgovarajućeg rešenja. Zatim se primenjuje *operator selekcije* – uzimaju se bolje jedinke, po nekom postupku odabira i formira nova generacija. Nakon toga neki članovi populacije su podvrgnuti uticajima genetskih operatora: *ukrštanja* i *mutacije*.

```
Unošenje_Ulaznih_Podataka();  
Generisanje_Početne_Populacije();  
while(!Kriterijum_Zaustavljanja_Genetskog_Algoritma())  
{
```

```

for (i=1;i<=N_populacije;i++ )
    obj[i]=Funkcija_Cilja(i);

Funkcija_Prilagođenosti();

Selekcija();

Ukrštanje();

Mutacija();

}

Štampanje_Izlaznih_Podataka();

```

## 1.2. Hab lokacijski problemi

Hab lokacijski problemi (*hub location problems*) su postali veoma važna oblast teorije lokacije u poslednjih dvadeset godina. Ovo je uzrokovano korišćenjem u velikoj meri mreže habova (*hub networks*) u modernim transportnim i telekomunikacijskim sistemima. Hab lokacijski problemi se koriste za dizajniranje transportnih mreža (železnički i drumski sistem, poštanski sistemi, prevoženje putnika i robe u avio saobraćaju, sistem brze isporuke itd.). Protok između čvorova u mreži možemo definisati kao broj putnika ili količinu robe koju treba transportovati od početnog čvora (snabdevača) do krajnjeg čvora (korisnika). Transport između ova dva čvora se ne uspostavlja direktno, već je usmeren preko skupa čvorova koji su označeni kao *habovi*. Ovi problemi uglavnom spadaju u NP-teške probleme [3], [7], [12], [8]. NP-teški problemi su problemi koji su nerešivi egzaktnim metodama. Ove metode najčešće ne daju rezultate, bilo zbog prekoračenja postojeće memorije bilo zbog zahtevane dužine izvršavanja, kada se primene na probleme velikih dimenzija. Međutim, iako su rezultati dobijeni primenom heurističkih metoda dosta nepouzdaniji (često možemo dobiti optimalno rešenje, ali ne uvek), NP-teške probleme je bolje rešavati njima.

## 2. Hab lokacijski problem neograničenih kapaciteta sa višestrukim alokacijama

Neka je  $I = \{1, \dots, n\}$  skup različitih čvorova mreže, pri čemu svaki čvor označava lokaciju korisnika/snabdevača ili potencijalnu lokaciju haba. Rastojanje od  $i$ -tog do  $j$ -tog čvora je  $C_{ij}$ , i može se pretpostaviti da važi nejednakost trougla [5]. Količina robe koju treba transportovati od  $i$ -tog snabdevača do  $j$ -tog korisnika je označena sa  $W_{ij}$ . U formulaciji problema promenljive su korišćene na sledeći način:

- Binarna promenljiva  $y_k$  uzima vrednost 1 ako je hab lociran na  $k$  – tom čvoru, u suprotnom ima vrednost 0.
- $x_{ijkm}$  je količina protoka koja polazi iz čvora  $i$  koji se sakuplja u habu  $k$ , distribuirira se preko haba  $m$  i prosleđuje u čvor  $j$ .

Protok od čvora-snabdevača do čvora-korisnika sastoji se od tri komponente: transfer od snabdevača do prvog haba, transport između habova i distribucija od poslednjeg haba do korisnika, pri čemu se podrazumeva šema višestruke alokacije. Parametri  $\chi$  i  $\delta$  redom označavaju troškove (cenu) kolekcije i distribucije robe po jedinici količine dok  $1 - \alpha$  predstavlja koeficijent uštede za transport između habova. Troškovi uspostavljanja snabdevača preko haba  $k$  su označeni sa  $f_k$ . Koristeći gornju notaciju, UMAHLP (*Uncapacitated Multiple Allocation Hub Location Problem*) matematički se može zapisati kao (videti u [6]):

$$\min \sum_{i,j,k,m} W_{ij}(\chi C_{ik} + \alpha C_{km} + \delta C_{mj})x_{ijkm} + \sum_k f_k y_k \quad (2.1)$$

Uz uslove:

$$\sum_{k,m} x_{ijkm} = 1 \quad , \text{ za svako } i, j \quad (2.2)$$

$$\sum_m x_{ijkm} + \sum_{m,m \neq k} x_{ijmk} \leq y_k \quad , \text{ za svako } i, j, k \quad (2.3)$$

$$y_k \in \{0,1\} \quad , \text{ za svako } k \quad (2.4)$$

$$x_{ijkm} \geq 0 \quad , \text{ za svako } i, j, k, m \quad (2.5)$$

Funkcija cilja (2.1) minimizuje sumu transportnih troškova snabdevač-hab, hab-hab i hab-korisnik, pomnoženih sa koeficijentima  $\chi$ ,  $\alpha$  i  $\delta$  respektivno i na tu sumu se dodaju fiksni troškovi  $f_k$  za svaki uspostavljeni hab (tamo gde je  $y_k = 1$ ). Ograničenje (2.2) označava da se celokupni protok kreće kroz sve parove čvorova, dok ograničenje (2.3) osigurava da se protok kreće samo kroz otvorene habove. Ograničenja (2.4) i (2.5) označavaju binarnu, odnosno ne-negativnu reprezentaciju promenljivih  $y_k$  i  $x_{ijkm}$  respektivno.

UMAHLP je NP-kompletan, sa izuzetkom specijalnih slučajeva koji se mogu rešiti u polinomijalnom vremenu (na primer, kada je matrica protoka  $W_{ij}$  retka). Ako je skup habova unapred izabran, problem je takođe polinomijalno rešiv koristeći algoritam najkraćeg puta (*Shortest-Path Algorithm*) u  $O(n^3)$  vremenu izvršavanja ([9]).

Postoji nekoliko pristupa u literaturi koji razmatraju UMAHLP. Problem je prvi put formulisan u [4]. Tehnika dualnog penjanja unutar metode grananja i ograničavanja (*Branch-and-Bound scheme*) je prvi put primenjena za rešavanje UMAHLP u [13] na ORLIB (*Operation Research Library-Standardna kolekcija instanci za operaciona istraživanja*) (videti u [1]) hab instancama sa brojem čvorova  $n \leq 25$ . Sličan pristup je korišćen u [18]. Rezultati su prikazani na primeru koji je imao do 40 čvorova.

U [2] su korišćene mešovite linearne i celobrojne programske formulacije (*MILP-Mixed Integer Linear Programming*) za tri višestruke alokacije hab lokacijskih problema, uključujući i UMAHLP. Za svaki problem su razvijane pretprocesorske procedure i određivane preciznije granice. Ovaj pristup je testiran na standardnim ORLIB AP (*Australian Post*) instancama problema sa najviše 50 čvorova.

Glavna ideja u [17] je da učvrsti MILP formulaciju i da smanji broj uslova koristeći rezultate na polju poliedarske strukture problema pakovanja grupa. Druga ideja, koja je prikazana u [6], je razmatrala problem dualnosti MILP formulacije. Autori su prvo konstruisali heuristički metod, baziran na tehnici dualnog penjanja, koji daje skoro 70% optimalnih rešenja na ORLIB instancama do 120 čvorova. Ta heuristika je kasnije ubačena u *Branch-and-Bound* algoritam koji je davao optimalna rešenja u svim slučajevima.

### 3. Predloženi hibridni genetski aloritam

U praksi vrlo često se javlja potreba da se izvršenje genetskog algoritma na neki način ubrza. U tom cilju se uvode nove heuristike čija primena i po nekoliko puta može ubrzati rad genetskog algoritma. Heuristike je moguće primeniti na neku pojedinačnu jedinku (najčešće je to najbolja jedinka), na nekoliko jedinki ili na celu populaciju. Hibridni genetski algoritam koristi heuristiku lokalnog pretraživanja za poboljšanje kvaliteta rešenja. Heuristika se izvršava u svakoj generaciji algoritma, pre primene genetskih operatora selekcije, ukrštanja i mutacije. Ona se primenjuje samo

na najbolju jedinku, ukoliko se ta jedinka promenila u odnosu na prethodnu generaciju. Heuristika je deterministička, tako da ako ne uspeva da popravi najbolju jedinku u tekućoj generaciji, ne može dati nikakvo dalje poboljšanje.

U svakoj generaciji, pre primene genetskih operatora, kod UMAHLP-a heuristikom lokalnog pretraživanja pokušaćemo da poboljšamo kvalitet rešenja. To ćemo uraditi tako što ćemo na najboljoj jedinki na nekom čvoru promeniti status tj. ako je na određenoj poziciji ne-hab čvor on postaje hab i obrnuto. Posle svake promene izračunaćemo vrednost funkcije cilja. Ukoliko dobijemo poboljšanje nastavljamo dalje sve dok se jedinka može poboljšati. Inače, nastavljamo sa radom genetskog algoritma.

### 3.1. Kodiranje i funkcija cilja

U implementaciji genetskog algoritma jedinke su binarno kodiranje. Svako rešenje je predstavljeno binarnim stringom dužine  $N$ . U genetskom kodu cifra 1 označava da je uspostavljen hab na toj lokaciji, a cifra 0 da nije uspostavljen. Pošto snabdevač i korisnik mogu biti pridruženi samo uspostavljenim hab-lokacijama, iz genetskog koda se dobija niz  $(y_k)$ . Takođe, ne postoje kapaciteti, tako da vrednosti  $x_{ijkm}$  mogu biti izračunate za vreme ocene funkcije cilja. Neka je jedinka predstavljena binarnim stringom dužine  $N = 8$  (01011001). Habovi su uspostavljeni na prvoj, trećoj, četvrtoj i sedmoj poziciji.

Za fiksiran skup habova  $(y_k)$  koristi se modifikacija poznatog *Floyd-Warshall* algoritma najkraćeg puta, opisanog u [9]. Posle nalaženja najkraćih puteva za svaki par čvorova u mreži, lako se računa funkcija cilja. To se postiže jednostavnim sumiranjem najkraćih rastojanja snabdevač-hab, hab-hab i hab-korisnik pomnoženih odgovarajućim parametrima  $\chi$ ,  $\alpha$  i  $\delta$ .

### 3.2. Genetski operatori i drugi aspekti genetskog algoritma

Predloženi genetskog algoritma koristi fino-gradiranu turnirsku selekciju, opisanu u [11]. Umesto celobrojnog parametra  $N_{tour}$ - veličina turnirske grupe, fino-gradirana turnirska selekcija zavisi od realnog parametra  $F_{tour}$ - željena srednja veličina turnira. Operator fino-gradirane turnirske selekcije realizuje dva tipa turnira. Prvi tip je održan  $k_1$  puta i njegova veličina je  $[F_{tour}] + 1$ . Drugi tip se realizuje  $k_2$  puta sa  $[F_{tour}]$  jedinki-učesnika turnira. Pošto je vrednost  $F_{tour} = 5.4$  korišćena u ovoj implementaciji genetskog algoritma, odgovarajuće vrednosti  $k_1$  i  $k_2$  (za  $N_{tour} = 50$  ne-elitnih jedinki) su 20 i 30 respektivno.

U implementaciji genetskog algoritma se koristi jednopoziciono ukrštanje. Pri jednopozicionom ukrštanju na početku se bira  $N/2$  parova jedinki gde je  $N$  veličina populacije. Zatim se za svaki par jedinki, sa unapred zadatom verovatnoćom (nivoom) ukrštanja, na slučajan način bira tačka ukrštanja i vrši razmena delova genetskih kodova datih parova jedinki posle date pozicije.

Jedinke-potomci generisane operatorom ukrštanja podležu mutaciji sa zaleđenim bitovima. Operator mutacije se realizuje promenom slučajno izabranog gena u genetskom kodu jedinki (0 u 1, 1 u 0), sa osnovnim nivoom mutacije od  $0.4/n$  za nezaleđene i  $1.0/n$  za zaleđene bitove. Oba nivoa mutacije su konstantna tokom svih generacija genetskog algoritma.

Može se desiti da tokom izvršavanja genetskog algoritma da (skoro) sve jedinke u populaciji imaju isti gen na određenoj poziciji. Takvi geni (može ih biti više) se nazivaju „zaleđeni“ geni. Ako je broj zaleđenih bitova  $l$ , pretraživački prostor postaje  $2^l$  puta manji i mogućnost preuranjene konvergencije rapidno raste. Operatori selekcije i ukrštanja ne mogu da promene vrednost nijednog zaleđenog bita, a osnovni nivo mutacije je često veoma mali da bi povratili izgubljeni regioni prostora pretrage. Ako se osnovni nivo mutacije značajno poveća, genetski algoritam postaje slučajna pretraga (*random search*). Iz ovih razloga, nivo mutacije se povećava samo na zaleđenim

genima. Dakle, u ovoj implementaciji zaleđeni bitovi se mutiraju sa 2.5 puta većom verovatnoćom ( $1.0/n$ ) u odnosu na one koji nisu zaleđeni ( $0.4/n$ ).

Početna populacija broji 150 jedinki i ona je slučajno generisana obezbeđujući pri tom maksimalnu raznovrsnost genetskog materijala. U ovom pristupu se primenjuje tzv. *stacionarni genetski algoritam sa elitističkom strategijom* ([15], [16]). Dakle, trećinu populacije ćemo zameniti u svakoj generaciji, a dve trećine tj. 100 jedinki će direktno proći u sledeću generaciju. Funkcije cilja tih 100 najboljih (elitnih) jedinki računamo samo u prvoj generaciji i tako obezbeđujemo uštedu vremena izračunavanja. Njihove fitnes vrednosti se postavljaju na nulu, tako da će operator selekcije izbeći da ih prenese u narednu generaciju. Ovo je veoma efikasna metoda za čuvanje raznolikosti genetskog materijala i sprečava preuranjenu konvergenciju. Jedinke sa istom funkcijom cilja, ali različitim genetskim kodovima mogu nekad dominirati u populaciji. U slučaju da su im i genetski kodovi slični, genetski algoritam se može zaglaviti u lokalnom optimumu. Iz tih razloga je korisno ograničiti njihovo pojavljivanje nekom konstantom  $N_{rv}$  (to je postavljeno na 40 u ovoj implementaciji genetskog algoritma).

Keširanje ima veliki značaj jer omogućava uštedu vremena pri izvršavanju genetskog algoritma ([14], [16]). Kod genetskog algoritma veći deo vremena se potroši na računanje vrednosti funkcije cilja jedinke. Ta vrednost zajedno sa genetskim kodom jedinke se smešta u memoriju. Pre računanja funkcije cilja proverava se da li se ta jedinka pojavljivala u keš memoriji. Ako jeste, njena funkcija cilja se ne računa ponovo, već se uzima vrednost iz keš memorije. Inače, vrednost funkcije cilja se računa i zajedno sa genetskim kodom se smešta u keš memoriju.

#### 4. Rezultati

U ovom poglavlju predstavljeni su rezultati predloženog genetskog algoritma i sva testiranja su izvedena na računaru Intel sa 2.5GHz procesorom i sa 1 GB RAM memorije pod Linux (Knoppix 5.3.1) operativnim sistemom. Algoritam je kodiran u programskom jeziku C. Za testiranje algoritma korišćene su AP instance. AP instance su dobijene iz studije o australijskom poštanskom sistemu isporuke. Instanca najveće dimenzije iz ovog skupa uključuje 200 čvorova (regione poštanskih brojeva), dok se manje instance mogu dobiti iz najveće agregacijom skupa čvorova. Rastojanje između gradova zadovoljava nejednakost trougla, ali protok između uređenih parova početnih i krajnjih čvorova nije simetričan. Fiksne cene su uključene u AP setovima podataka kao u [10]. Koeficijenti  $\chi$ ,  $\delta$  i  $\alpha$  koji odgovaraju kolekciji, distribuciji i transportu između habova uzimaju iste vrednosti kao u [6].

Kolone u tabeli 4.1 sadrže sledeće podatke (po navedenom redu):

- Dimenzija trenutne AP instance gde L (*loose*) označava „lakše“, a T (*tight*) „teže“ fiksne cene;
- Optimalno rešenje ( $opt_{vr}$ ) ako je poznato unapred, u suprotnom pišemo „ – “;
- Najbolje rešenje genetskog algoritma ( $GA_{najb}$ ) u slučaju kad genetski algoritam dostiže optimum za trenutnu instancu;
- Prosečno vreme  $t$  (u sekundama) potrebno da bi se dobio najbolji rezultat genetskog algoritma;
- Prosečno ukupno vreme  $t_{uk}$  (u sekundama) za završetak genetskog algoritma;
- Prosečan ukupan broj generacija ( $gen$ );
- Prosečna relativna greška  $Gr$  (u procentima) rešenja genetskog algoritma u odnosu na  $opt_{vr}$  ili  $GA_{najb}$ , računa se po formuli

$$Gr = \left| \frac{GA_{najb} - opt_{vr}}{GA_{najb}} \right| \cdot 100$$

- Standardna devijacija  $\sigma$  relativne greške (u procentima);
- Prosečan broj izračunavanja funkcije cilja (*eval*);
- Prosečna vrednost uštede primenom tehnike keširanja *Keš* (u procentima)

U svakoj AP instanci genetski algoritam je bio pokrenut 20 puta. Maksimalan broj generacija u ovoj implementaciji genetskog algoritma je  $N_{gen} = 1000$ . Ponavljanje vrednosti najbolje funkcije cilja je ograničeno konstantom  $N_{rep} = 500$ .

Tabela 4.1 Rezultati genetskog algoritma na AP instancama sa  $\chi=3$ ,  $\alpha=0.75$  i  $\delta=2$

<i>inst.</i>	<i>opt<sub>vr</sub></i>	<i>GA<sub>najb</sub></i>	<i>t</i> (s)	<i>t<sub>uk</sub></i> (s)	<i>gen</i>	<i>Gr</i> (%)	$\sigma$ (%)	<i>eval</i>	<i>Keš</i> (%)
10L	221 032.734	221 032.734	<0.001	0.012	202.7	0.000	0.000	524.8	94.900
10T	257 558.086	257 558.086	<0.001	0.012	201.0	0.000	0.000	557.5	94.500
20L	230 385.454	230 385.454	<0.001	0.041	203.5	0.000	0.000	1633.8	84.200
20T	266 877.485	266 877.485	<0.001	0.047	207.1	0.000	0.000	1675.1	84.000
25L	232 406.746	232 406.746	<0.001	0.075	204.8	0.000	0.000	2090.8	79.900
25T	292 032.080	292 032.080	0.001	0.069	207.0	0.000	0.000	2124.2	79.800
40L	237 114.749	237 114.749	0.016	0.237	218.6	0.000	0.000	3118.2	71.800
40T	293 164.836	293 164.836	<0.001	0.202	201.0	0.000	0.000	2893.2	71.600
50L	233 905.303	233 905.303	0.020	0.424	212.4	0.000	0.000	3648.6	66.100
50T	296 024.896	296 024.896	0.021	0.404	214.7	0.000	0.000	3601.9	66.900
60L	225 042.310	225 042.310	0.018	0.620	205.7	0.000	0.000	3772.0	63.800
60T	243 416.450	243 416.450	0.036	0.868	215.8	0.000	0.000	4090.1	62.600
70L	229 874.500	229 874.500	0.078	1.223	223.3	0.041	0.185	4578.5	59.500
70T	249 602.845	249 602.845	0.050	1.097	214.8	0.000	0.000	4335.6	60.200
80L	225 166.922	225 166.922	0.294	2.449	257.9	0.000	0.000	5360.7	58.700
80T	268 209.406	268 209.406	0.129	1.885	230.3	0.000	0.000	4886.3	58.100
90L	226 857.465	226 857.465	0.153	2.405	227.3	0.000	0.000	5208.1	54.600
90T	277 417.972	277 417.972	0.144	2.465	225.4	0.000	0.000	5078.5	55.500
100L	235 097.228	235 097.228	0.447	4.218	250.2	0.000	0.000	5669.8	55.200
100T	305 097.949	305 097.949	0.049	2.513	204.6	0.000	0.000	4716.6	54.600
110L	218 661.965	218 661.965	0.168	3.679	215.9	0.067	0.299	4955.1	54.700
110T	223 891.822	223 891.822	0.326	4.623	235.8	0.000	0.000	5554.6	53.400
120L	222 238.922	222 238.922	0.201	4.312	216.4	0.000	0.000	5054.7	53.900
120T	229 581.755	229 581.755	0.725	6.418	254.1	0.000	0.000	6339.4	50.600
130L	–	223 814.109	0.891	9.876	260.3	0.000	0.000	6334.9	51.900
130T	–	230 865.451	0.822	10.017	258.1	0.000	0.000	6550.1	49.700
200L	–	230 204.343	6.596	47.401	285.9	1.487	1.837	7722.7	46.500
200T	–	268 787.633	8.173	61.300	297.8	0.126	0.160	8376.1	44.300

Za koeficijente  $\chi = 3$ ,  $\alpha = 0.75$  i  $\delta = 2$  (tabela 4.1),  $\chi = 1$ ,  $\alpha = 0.1$  i  $\delta = 1$ ,  $\chi = 1$ ,  $\alpha=0.5$  i  $\delta = 1$ ,  $\chi = 1$ ,  $\alpha = 0.9$  i  $\delta = 1$  za sve AP instance (osim za 130L, 130T, 200L, 200T), se postiže optimalno rešenje. U sva četiri slučaja najbolje rešenje genetskog algoritma se poklapa sa optimalnim u prve 24 instance.

Predloženi genetski algoritam brzo dostiže sva poznata optimalna rešenja ( $n \leq 120$ ) za  $t \leq 0.781s$ . Za ostale instance većih dimenzija, za koje optimum nije poznat, genetski algoritam dobija rešenje za  $t \leq 8.173s$ . Međutim, kolone  $t_{uk}$  pokazuju da naš algoritam prolazi kroz dodatno vreme  $t_{uk} - t$  (dok ne bude zadovoljen završni kriterijum), iako je dostigao optimalno rešenje. Pristup predloženog genetskog algoritma ne može da potvrdi optimalnost dobijenih rešenja, ali predstavlja značajan doprinos postojećoj metodi za rešavanje UMAHLP zato što omogućava rešavanje instanci problema velikih dimenzija koje su ranije bile nerešive.

## 5. Zaključak

U ovom radu opisan je hibridni genetski algoritam koji je posebno dizajniran za rešavanje UMAHLP. Predloženi pristup je primenjen na NP-težak problem kombinatorne optimizacije koji ima značajnu primenu u praksi.

Razmatrani problem je težak za rešavanje i to je jedan od razloga što postojeće egzaktne metode mogu rešiti samo instance problema manjih dimenzija. Postojeće heuristike, zasnovane na principu lokalnog pretraživanja, uglavnom ne daju rešenja zadovoljavajućeg kvaliteta na instancama problema većih dimenzija. Predložena hibridizacija evolutivnog pristupa i lokalnog pretraživanja hibridnog genetskog algoritma je vrlo robustna i efikasna u rešavanju datog problema, čak i na instancama velikih dimenzija.

Koncept genetskog algoritma opisan u ovom radu koristi binarno kodiranje jedinki-rešenja. Primenjeni genetski operatori koji su pokazali najbolje rezultate pri rešavanju UMAHLP su: fino-gradirana turnirska selekcija, operator jednopozicionog ukrštanja i operator mutacije sa zaleđenim genima. Kod implementacije genetskog algoritma keširanje u velikoj meri poboljšava performanse, ali ne utiče na ostale aspekte algoritma. Takođe, primenjene su razne metode za sprečavanje preuranjene konvergencije usled gubljenja raznovrsnosti genetskog materijala i spore konvergencije genetskog algoritma. Predložena implementacija primenjuje stacionarni genetski algoritam sa elitističkom strategijom. Heuristika lokalnog pretraživanja se primenjuje na najbolju jedinku u svakoj generaciji genetskog algoritma, ukoliko se ona promenila u odnosu na prethodnu generaciju. Lokalno pretraživanje se koristi pre primene genetskih operatora selekcije, ukrštanja i mutacije. Prikazani rezultati hibridnog genetskog algoritma jasno pokazuju da su dobijena rešenja visokog kvaliteta. Na svim rešavanim instancama problema dobijena su optimalna rešenja. Predloženi koncept hibridnog genetskog algoritma takođe daje rešenja na instancama problema velikih dimenzija koje do sada nisu razmatrane u literaturi.

## Literatura

- [1] **Beasley J.E.**, "Obtaining Test Problems via Internet", *Journal of Global Optimization*, Vol. 8, pp. 429-433 (1996)  
<http://msmga.ms.ic.ac.uk/jeb/orlib/info.html>
- [2] **Boland N., Krishnamoorthy M., Ernst A.T., Ebery J.**, "Preprocessing and Cutting for Multiple Allocation Hub Location Problems", *European Journal of Operational Research*, Vol. 155, pp.638-653 (2004)
- [3] **Brassard G., Bratley P.**, „Algorithms: Theory and Practice“, Prentice-Hall Int., Englewoods Cliffs NJ (1988)
- [4] **Campbell, J. F.**, " Integer programming formulations of discrete hub location problems", *European Journal of Operational Research* Vol. 72, pp. 387-405 (1994)
- [5] **Campbell J.F., Ernst A. and Krishnamoorthy M.**, "Hub Location Problems" in Hamacher H. and Drezner Z., eds.: *Facility Location : Applications and Theory*, Springer-Verlag, Berlin-Heidelberg, pp. 373-407 (2002)
- [6] **Cánovas L., García S., Marin A.**, „Solving the Uncapacitated Multiple Allocation Hub Location Problem by Means of a Dual-ascent Technique“, *European Journal of Operational Research*, Vol. 179, pp. 990-1007 (2007)
- [7] **Crescenci P., Kann V.**, "A compendium of NP optimization problems" (1997)  
<http://www.nada.kth.se/theory/problemlist.html>
- [8] **Drezner Z., Klamroth K., Schöbel A., Weslowsky G.O.**, The Weber problem, in H. Hamacher and Z. Drezner, eds.: *Facility Location : Applications and Theory*, Springer-Verlag, Berlin-Heidelberg, 1-25 (2002)
- [9] **Ernst, A.T., Krishnamoorthy M.**, "An Exact Solution Approach Based on Shortest-paths for p-hub Median Problem", *INFORMS Journal of Computing*, Vol. 10, pp. 149-162 (1998)
- [10] **Ernst, A.T., Krishnamoorthy M.**, "Solution algorithms for the capacitated single allocation hub location problem“, *Annals of Operational Research*, Vol.86, pp.141-159 (1999)
- [11] **Filipović V.** "Predlog poboljšanja operatora turnirske selekcije kod genetskih algoritama", *Magistarski rad*, Univerzitet u Beogradu, Matematički fakultet (1998)
- [12] **Jungnickel D.**, "Graphs, Networks and Algorithms", Springer Verlag, Berlin (1998)
- [13] **Klincewitz J.G.**, "A Dual Algorithm for the Uncapacitated Hub Location Problem", *Location Science*, Vol.4, No.3, pp.173-184 (1996)
- [14] **Kratica J.**, "Improving Performances of the Genetic Algorithm by Caching", *Computers and Artificial Inteligence*, Vol. 18, No. 3, pp.271-283 (1999)
- [15] **Kratica J.**, "Paralelizacija genetskih algoritama za rešavanje nekih NP- kompletnih problema", Doktorska disertacija, Matematički fakultet, Beograd (2000)
- [16] **Kratica J., Tošić D., Filipović V., Ljubić I.**, "Solving the Simple Plant Location Problem by Genetic Algorithm", *RAIRO Operations Research*, Vol. 73, No. 1, pp.127-142 (2001)
- [17] **Marín A., Cánovas L., Landete M.**, "New formulations for the Uncapacitated Multiple Allocation Hub Location Problem.", *European Journal of Operational Research*, Vol.172 (1),pp. 274-292 (2006)
- [18] **Mayer G., Wagner B.**, "An exact solution method for the multiple allocation hub location problem“, *Computers and Operational Research*, Vol. 29, pp. 715-739 (2002)



# PRIMENA GREBNEROVIH BAZA NA PROBLEM INVERZNE KINEMATIKE U ROBOTICI

## B. Banjac

student doktorskih studija Softverskog Inženjerstva  
Elektrotehničkog fakulteta Univerziteta u Beogradu  
Bulevar kralja Aleksandra 73, 11000 Beograd, Srbija,  
e-mail: bojan.b.mail@gmail.com

## G. Dimitrijević

student master studija Primenjene Matematike  
Elektrotehničkog fakulteta Univerziteta u Beogradu  
Bulevar kralja Aleksandra 73, 11000 Beograd, Srbija,  
e-mail: gocadmail@gmail.com

## S. Obradović, N. Ajduković

studenti master studija Softverskog Inženjerstva  
Elektrotehničkog fakulteta Univerziteta u Beogradu  
Bulevar kralja Aleksandra 73, 11000 Beograd, Srbija  
e-mail: smiljana.obradovic@gmail.com,  
nikolaajdukovic@yahoo.com,

## M. Čampara

student master studija Računarske tehnike ETH Zurich,  
Switzerland  
email: milan\_campara@yahoo.com

## B. Malešević

docent, Katedra za Primenjenu matematiku Elektrotehničkog fakulteta Univerziteta u Beogradu  
Bulevar kralja Aleksandra 73, 11000 Beograd, Srbija, e-mail: malesevic@etf.rs

**APSTRAKT:** U okviru rada je predstavljena primena Grebnerovih baza polinomijalnih ideala na problem inverzne kinematike u robotici. Aplikacija razvijena u okviru rada vizuelno demonstrira na koji način se ova oblast matematike može primeniti i na postojeće industrijske robote.

**Ključne reči:** Grebnerove baze polinomijalnih ideala, robotika

## 1. Računarske implementacija Grebnerovih baza

Kratak pregled teorije Grebnerovih baza polinomijalnih ideala dat je u potpunosti u radu [2] (simpozijum „Matematika i primene” 2011.). Zainteresovane upućujemo takođe i na reference [4] i [6].

Ovde razmatramo neke računarske implementacije za Grebnerove baze. Prvo detaljnije razmatramo programski paket Maple [12] i odgovarajuću biblioteku “Groebner” koja se inicijalizuje sa

```
with(Groebner);
```

Ilustrujmo upotrebu biblioteke “Groebner” na primeru sistema polinomskih jednačina:

$$f_1 := c*x^2+y = 0; f_2 := c*y^2+z*c = 0; f_3 := c*z^2-2 = 0;$$

Formirajmo listu polinoma:

$$F := [c*x^2+y, c*y^2+z*c, c*z^2-2];$$

Razmotrimo F sa jedne strane kao listu sa promenljivima  $x$ ,  $y$ ,  $z$  i konstantom  $c$ . Grebnerova baza  $GB_1$  ideala  $I_1$ , formiranog listom F, u odnosu na leksikografski poredak je određena komandom:

$$GB_1 := \text{Basis}(F, \text{plex}(x,y,z));$$

Maple komandama `Groebner[Solve](GB1, {x,y,z}): map(L -> Solve(convert(L[1], set), {x,y,z}), %): allvalues(%)`; dobijamo rezultat:

$$\begin{aligned} & \left\{ \left\{ x = 2^{(1/8)} \left( \frac{1}{c^5} \right)^{(1/8)}, y = -c 2^{(1/4)} \left( \frac{1}{c^5} \right)^{(1/4)}, z = -c^2 \sqrt{2} \sqrt{\frac{1}{c^5}} \right\} \right\}, \\ & \left\{ \left\{ x = 2^{(1/8)} \left( \frac{1}{c^5} \right)^{(1/8)} (-1)^{(1/4)}, y = -I c 2^{(1/4)} \left( \frac{1}{c^5} \right)^{(1/4)}, z = c^2 \sqrt{2} \sqrt{\frac{1}{c^5}} \right\} \right\}, \\ & \left\{ \left\{ x = 2^{(1/8)} \left( \frac{1}{c^5} \right)^{(1/8)} I, y = c 2^{(1/4)} \left( \frac{1}{c^5} \right)^{(1/4)}, z = -c^2 \sqrt{2} \sqrt{\frac{1}{c^5}} \right\} \right\}, \\ & \left\{ \left\{ x = 2^{(1/8)} \left( \frac{1}{c^5} \right)^{(1/8)} (-1)^{(3/4)}, y = c 2^{(1/4)} \left( \frac{1}{c^5} \right)^{(1/4)} I, z = c^2 \sqrt{2} \sqrt{\frac{1}{c^5}} \right\} \right\}, \\ & \left\{ \left\{ x = -2^{(1/8)} \left( \frac{1}{c^5} \right)^{(1/8)}, y = -c 2^{(1/4)} \left( \frac{1}{c^5} \right)^{(1/4)}, z = -c^2 \sqrt{2} \sqrt{\frac{1}{c^5}} \right\} \right\}, \\ & \left\{ \left\{ x = -2^{(1/8)} \left( \frac{1}{c^5} \right)^{(1/8)} (-1)^{(1/4)}, y = -I c 2^{(1/4)} \left( \frac{1}{c^5} \right)^{(1/4)}, z = c^2 \sqrt{2} \sqrt{\frac{1}{c^5}} \right\} \right\}, \end{aligned}$$

iz koga zaključujemo da je dati sistem rešiv po promenljivima  $x, y, z$  i ima konačno mnogo rešenja (gde je  $c$  nenulta konstanta). Sa druge strane razmotrimo  $F$  kao listu sa promenljivima  $c, x, y, z$  bez javljanja konstanti. Grebnerova baza  $GB_2$  ideala  $I_2$ , formiranog listom  $F$ , u odnosu na leksikografski poredak je određena komandom:

$$GB_2 := \text{Basis}(F, \text{plex}(c, x, y, z));$$

Maple komandama `Groebner[Solve](GB1, {c,x,y,z}): map(L->solve(convert(L[1],set),{c,x,y,z}),%)`: `allvalues(%)`; dobijamo rezultat  $c = \frac{2}{t^4}, x = \pm \sqrt{-\frac{t}{2}}, y = t, z = -t^2$  na osnovu koga je dati sistem rešiv po promenljivima  $c, x, y, z$ . Rešenja se dobijaju u funkciji parametra  $y = t \neq 0$ , samim tim dati sistem nije konačno rešiv.

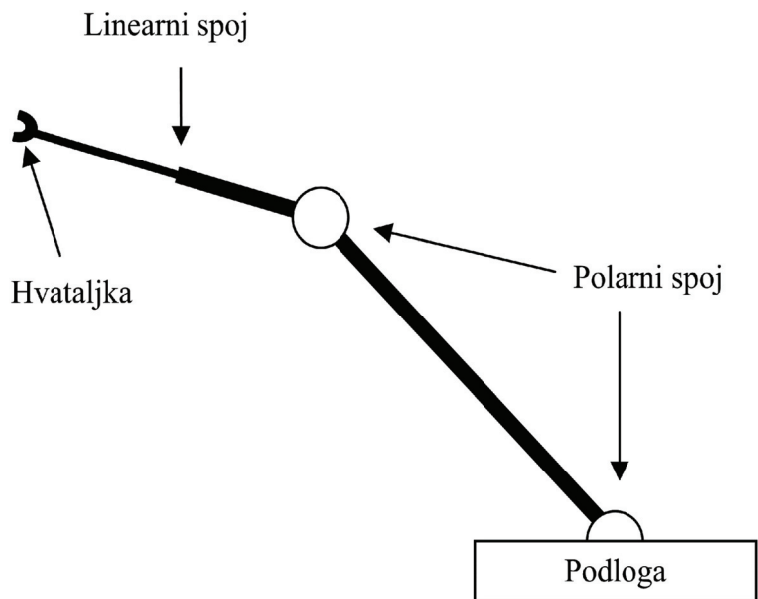
Osnovni algoritmi teorije Grebnerovih baza na nivou pseudokoda su dati u radu [2]. Napomenimo da je moguće razmatrane aplikacije Grebnerovih baza razviti i u Javi kao što je urađeno u [1], [2] i [3]. U ovom radu zbog obimnosti računa koji se koristi u problemima inverzne kinematike koristili smo programski paket Maple [12].

## 2. Problem inverzne kinematike

U ovoj sekciji je najpre objašnjen pojam robotske ruke i opisan model za koji je predloženo rešenje problema inverzne kinematike. Zatim sledi matematička predstava problema inverzne kinematike i opis postupka kojim se dolazi do rešenja. Konkretno opisani primer robotske ruke sa tri zglobova se opisuje sa 6 međusobno nezavisnih promenljivih. U praksi se koriste robotske ruke i sa većim brojem zglobova kao što je navedeno u [11].

Robotska ruka jeste programabilna mehanička ruka sa funkcijama sličnim ljudskoj ruci. Može predstavljati samostalan mehanizam ili biti deo nekog složenijeg robota. U opštem slučaju, zglobovi robotske ruke omogućavaju dve vrste pokreta, rotaciono kretanje i translacioni pomeraj. Jedan od načina za kontrolisanje pokreta robotske ruke jeste programiranje baze znanja unapred za sve moguće položaje u koje ruka može dospeti. Ovaj pristup je korišćen prilikom izrade programa za simulaciju robotske ruke jer je simulirana industrijska robotska ruka koja u većini slučajeva ima predefinisani ograničen skup položaja u kojima se može naći, a samim tim i skup pokreta koje može načiniti.

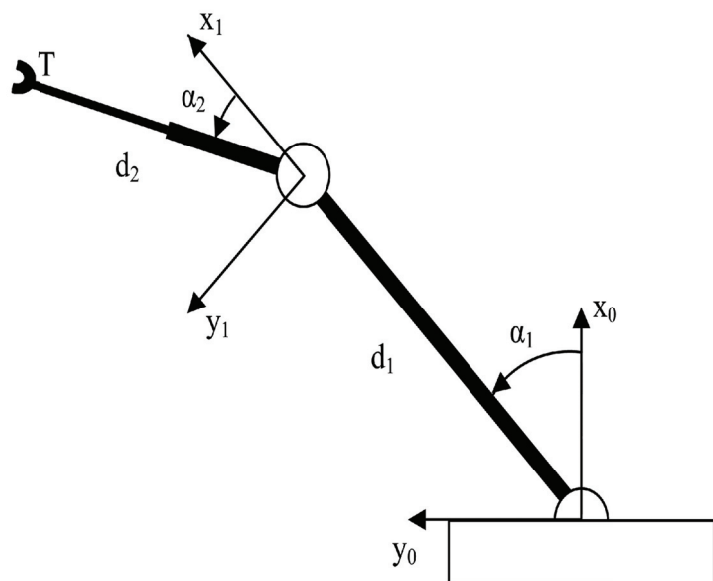
Opisaćemo model dvodimenzionalne robotske ruke sa dva zgloba i rešenje problema inverzne kinematike za taj slućaj. Matematićka predstava problema i postupak rešavanja su analogni i za ruku sa tri zgloba, ali je rešenje komplikovanije, te je zato opis dat za ruku sa dva zgloba. Model ruke se sastoji od niza segmenata (analogno ljudskim kostima) međusobno povezanih polarnim spojevima (analogno zglobovima ljudske ruke) koji leže u jednoj ravni. Postoje i linearni spojevi koji omogućavaju kretanje robotske ruke po jednom pravcu, tj izduživanje ruke. Položaj ruke određen je uglovima koje obrazuju susedni segmenti u odnosu na polarne spojeve.



Robotska ruka je fiksirana za podlogu, tj prvi segment je preko prvog polarnog zgloba povezan sa nepokretnom podlogom. Na suprotnom kraju robotske ruke se nalazi hvataljka pomoću koje ruka dohvata predmet i prebacuje ga u neki drugi položaj (ovaj deo robotske ruke nije modelovan i nije od znaćaja za problem inverzne kinematike). Na Slici 1 nalazi se skica robotske ruke sa oznaćenim segmentima i spojevima.

Najpre će biti prikazano rešavanje direktnog problema kinematike, a zatim će ti rezultati biti iskorišćeni za određivanje parametara potrebnih za pozicioniranje ruke u željeni položaj. Direktni problem kinematike jeste nalaženje položaja u koji robotska ruka dospeva kada se segmenti postave u položaje definisane uglovima u polarnim spojevima i izduženjima linearnih spojeva. Problem inverzne kinematike je dosta složeniji jer je potrebno da se na osnovu početne taćke (koordinata prvog polarnog spoja) i krajnje taćke (položaja hvataljke robotske ruke) odrede uglovi u polarnim spojevima i izduženja linearnih spojeva. Na Slici 2 oznaćene su promenljive koje će biti korišćene za definisanje problema.

Segmenti su numerisani po svom položaju u nizu segmenata počev od segmenta koji je fiksiran za podlogu. Dužina segmenta  $i$  obeležen je sa  $d_i$  i može biti promenljiva ili konstanta vrednost u zavisnosti od toga da li je segment  $i$  fiksni segment ili predstavlja linearni spoj. Za svaki polarni spoj vezan je po jedan koordinatni sistem, tako da spoj vezan za osnovu predstavlja koordinatni početak sistema  $(x_0, y_0)$ , a početak svakog narednog koordinatnog sistema  $(x_i, y_i)$  nalazi se na kraju segmenta  $i$  tako da osa  $x_i$  leži na pravcu segmenta  $i$ . Ugao koji segment  $i + 1$  obrazuje sa osom  $x_i$  oznaćen je sa  $\alpha_i$ . Ciljna taćka



$T$  određena je koordinatama  $(t_x, t_y)$  u sistemu  $(x_i, y_i)$ .

Neka su poznate koordinate  $(t_{x_1}, t_{y_1})$  ciljne tačke  $T$ , odnosno koordinate ciljne tačke u sistemu  $(x_1, y_1)$ . Da bismo izračunali koordinate tačke  $T(t_{x_0}, t_{y_0})$  u sistemu  $(x_0, y_0)$  najpre transliramo  $(t_{x_1}, t_{y_1})$  za vektor  $(d_1, 0)$  a zatim rotiramo za ugao  $\alpha_1$ . Koordinate nakon translacije i rotacije računamo na sledeći način:

$$\text{Translacija } (x, y) \text{ za vektor } (l_x, l_y) : \begin{cases} x' = x + l_x, \\ y' = y + l_y, \end{cases}$$

$$\text{Rotacija } (x, y) \text{ za ugao } \theta : \begin{cases} x' = \cos(\theta) \cdot x - \sin(\theta) \cdot y, \\ y' = \sin(\theta) \cdot x + \cos(\theta) \cdot y. \end{cases}$$

Dobijamo da su koordinate hvataljke  $T$  u sistemu  $(x_0, y_0)$

$$\begin{bmatrix} t_{x_0} \\ t_{y_0} \end{bmatrix} = \begin{bmatrix} \cos(\alpha_1) & -\sin(\alpha_1) \\ \sin(\alpha_1) & \cos(\alpha_1) \end{bmatrix} \cdot \begin{bmatrix} d_1 \\ 0 \end{bmatrix} + \begin{bmatrix} t_{x_1} \\ t_{y_1} \end{bmatrix}.$$

Ako uvedemo matrice operacija na sledeći način

$$\text{Translacija } (x, y) \text{ za vektor } (l_x, l_y) : \begin{bmatrix} 1 & 0 & l_x \\ 0 & 1 & l_y \\ 0 & 0 & 1 \end{bmatrix}$$

$$\text{Rotacija } (x, y) \text{ za ugao } \theta : \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

dobijene koordinate se mogu predstaviti na sledeći način

$$\begin{bmatrix} t_{x_0} \\ t_{y_0} \\ 1 \end{bmatrix} = \begin{bmatrix} \cos(\alpha_1) & -\sin(\alpha_1) & 0 \\ \sin(\alpha_1) & \cos(\alpha_1) & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & d_1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} t_{x_1} \\ t_{y_1} \\ 1 \end{bmatrix} = M_1 \cdot \begin{bmatrix} t_{x_1} \\ t_{y_1} \\ 1 \end{bmatrix} = M_1 \cdot \vec{r}_1.$$

U opštem slučaju ćemo imati

$$\begin{aligned} \vec{r}_{i-1} &= M_i \cdot \vec{r}_i, \\ \vec{r}_0 &= M_1 \cdot M_2 \cdot \dots \cdot M_i \cdot \vec{r}_i. \end{aligned}$$

Ako uzmemo u obzir da hvataljka robotske ruke leži u koordinatnom početku sistema  $(x_i, y_i)$ , dobijamo formulu

$$\vec{r}_0 = M_1 \cdot M_2 \cdot \dots \cdot M_i \cdot \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}.$$

Navedena formula predstavlja rešenje direktnog problema kinematike – na osnovu poznatih uglova polarnih spojeva  $\alpha_1, \alpha_2, \dots, \alpha_i$  i dužina linearnih spojeva  $d_1, d_2, \dots, d_i$  računamo poziciju hvataljke  $\vec{r}_0$  u odnosu na podlogu za koju je ruka pričvršćena.

U nastavku će biti opisan metod za rešavanje problema inverzne kinematike. Potrebno je odrediti sve načine da hvataljka robotske ruke dospe do zadate ciljne tačke. Usvojeno je da su segmenti robotske ruke fiksne dužine, tj da ne postoje linearni spojevi. Metod za rešavanje zahteva da se koriste polinomijalne funkcije, te se zato uvode sledeće smene

$$s_i \equiv \sin(\alpha_i),$$

$$c_i \equiv \cos(\alpha_i).$$

U sistem se takođe uvode i jednačine koje predstavljaju osnovnu zavisnost trigonometrijskih funkcija

$$c_1^2 + s_1^2 - 1 = 0,$$

$$c_2^2 + s_2^2 - 1 = 0.$$

Nakon uvedenih smena, matrica  $M_i$  koja predstavlja operaciju rotiranja i transliranja dobija sledeći oblik

$$M_i = \begin{bmatrix} c_i & -s_i & 0 \\ s_i & c_i & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & d_i \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} c_i & -s_i & c_i d_i \\ s_i & c_i & s_i d_i \\ 0 & 0 & 1 \end{bmatrix}.$$

Računamo koordinate ciljne tačke u zavisnosti od promenljivih  $c_1, s_1, c_2, s_2$ , dok vrednosti  $t_{x_0}, t_{y_0}, d_1, d_2$  smatramo poznatim.

$$\vec{r}_0 = M_1 \cdot M_2 \cdot \vec{r}_2 = \begin{bmatrix} c_1 & -s_1 & c_1 d_1 \\ s_1 & c_1 & s_1 d_1 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} c_2 & -s_2 & c_2 d_2 \\ s_2 & c_2 & s_2 d_2 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} t_{x_0} \\ t_{y_0} \\ 1 \end{bmatrix} = \begin{bmatrix} d_2 c_1 c_2 - d_2 s_1 s_2 + d_1 c_1 \\ d_2 s_1 c_2 + d_2 c_1 s_2 + d_1 s_1 \\ 1 \end{bmatrix}.$$

Potrebno je rešiti sistem polinomijalnih jednačina

$$f_1 = d_2 c_1 c_2 - d_2 s_1 s_2 + d_1 c_1 - t_{x_0} = 0,$$

$$f_2 = d_2 s_1 c_2 + d_2 c_1 s_2 + d_1 s_1 - t_{y_0} = 0;$$

uz polinomsko-trigonometrijska ograničenja

$$f_3 = c_1^2 + s_1^2 - 1 = 0,$$

$$f_4 = c_2^2 + s_2^2 - 1 = 0.$$

Izračunaćemo Grebnerovu bazu ideala generisanog listom polinoma  $F = [f_1, f_2, f_3, f_4]$ . Ako usvojimo da je  $c_2 > s_2 > c_1 > s_1$  i primenimo leksikografski monomijalni poredak, dobijamo Grebnerovu bazu  $G = [g_1, g_2, g_3, g_4]$ :

$$g_1 = s_1^2 - \frac{t_{y_0} (d_1^2 - d_2^2 + t_{x_0}^2 + t_{y_0}^2)}{d_1 (t_{x_0}^2 + t_{y_0}^2)} s_1 + \frac{(d_1^2 - d_2^2)^2 - 2t_{x_0}^2 (d_1^2 + d_2^2) + 2t_{y_0}^2 (d_1^2 - d_2^2) + (t_{x_0}^2 + t_{y_0}^2)^2}{4d_1^2 (t_{x_0}^2 + t_{y_0}^2)},$$

$$g_2 = c_1 + \frac{t_{y_0}}{t_{x_0}} s_1 - \frac{d_1^2 - d_2^2 + t_{x_0}^2 + t_{y_0}^2}{2d_1 t_{x_0}},$$

$$g_3 = s_2 - \frac{t_{y_0}}{d_2} c_1 + \frac{t_{x_0}}{d_2} s_1,$$

$$g_4 = c_2 - \frac{t_{x_0}}{d_2} c_1 - \frac{t_{y_0}}{d_2} s_1 + \frac{d_1}{d_2}.$$

Izjednačavanjem redom polinoma  $g_1, g_2, g_3, g_4$  sa nulom, dobijamo četiri homogene polinomijalne jednačine koje je jednostavno rešiti. Najpre se iz prve jednačine izračuna vrednost  $s_1$ , zatim iz druge vrednost  $c_1$ , iz treće  $s_2$  i na kraju iz četvrte jednačine vrednost  $c_2$ . Kako je prva jednačina kvadratna jednačina po promenljivoj  $s_1$ , a ostale jednačine su linearne po promenljivama  $s_1, s_2, c_1, c_2$ , mogući su sledeći slučajevi:

- Rešenja za  $s_1$  su različiti realni brojevi
  - Postoje dva skupa rešenja za  $s_1, s_2, c_1, c_2$ , tj dva moguća položaja segmenata koji postavljaju robotsku ruku u istu ciljnu poziciju
  - Dobijeni položaji su osno-simetrični oko ose provučene kroz početnu tačku i ciljnu tačku hvataljke robotske ruke
- Postoji jedinstven realan broj  $s_1$  koji zadovoljava  $g_1 = 0$ 
  - Ovo je granični slučaj u kome je robotska ruka maksimalno opružena
- Oba rešenja za  $s_1$  su imaginarni brojevi
  - Ovakvo rešenje ukazuje na to da se ciljna tačka nalazi van domašaja robotske ruke.

Iz dobijenih vrednosti za  $s_1, s_2, c_1, c_2$  jednostavno je izračunati vrednosti uglova  $\alpha_1, \alpha_2$ . Važno je reći da se mora izvršiti provera izračunatih vrednosti koje određuju položaj robotske ruke. Potrebno je uvrstiti konkretne vrednosti za  $t_{x_0}, t_{y_0}, d_1, d_2$  i proveriti da li su dobijeni položaji validni. Položan je nevalidan ako robotska ruka prolazi kroz podlogu ili se dobiju takve vrednosti za uglove da se segmenti robotske ruke međusobno presecaju. Ilustracije radi, prikazaćemo rezultate koji se dobijaju ako se usvoji da je  $d_1 = d_2 = 1m$ . Grebnerova baza dobija oblik:

$$g_1' = s_1^2 - t_{y_0} s_1 + \frac{-4t_{x_0}^2 + (t_{x_0}^2 + t_{y_0}^2)^2}{4(t_{x_0}^2 + t_{y_0}^2)},$$

$$g_2' = c_1 + \frac{t_{y_0}}{t_{x_0}} s_1 - \frac{t_{x_0}^2 + t_{y_0}^2}{2t_{x_0}},$$

$$g_3' = s_2 - t_{y_0} c_1 + t_{x_0} s_1,$$

$$g_4' = c_2 - t_{x_0} c_1 - t_{y_0} s_1 + 1.$$

a rezultati su

$$s_1 = \frac{t_{y_0}}{2} \pm \frac{1}{2} \sqrt{t_{x_0}^2 \cdot \frac{4 - t_{x_0}^2 - t_{y_0}^2}{t_{x_0}^2 + t_{y_0}^2}},$$

$$c_1 = -\frac{t_{y_0}}{t_{x_0}} s_1 + \frac{t_{x_0}^2 + t_{y_0}^2}{2t_{x_0}},$$

$$s_2 = t_{y_0} c_1 - t_{x_0} s_1,$$

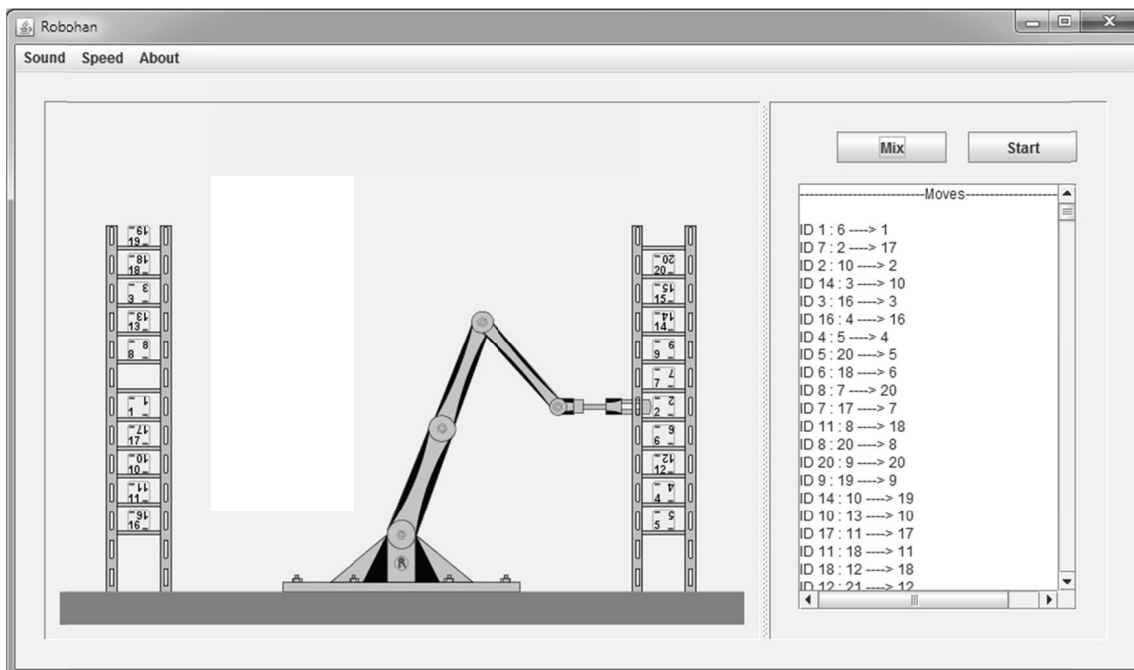
$$c_2 = t_{x_0} c_1 + t_{y_0} s_1 - 1.$$

Simulacija je urađena za robotsku ruku sa tri segmenta jednakih fiksnih dužina  $d_1 = d_2 = d_3 = 0.32m$ . Sva izračunavanja su vršena pomoću softverskog alata za simboličku algebru Maple. Postavljanje sistema jednačina i njegovo rešavanje analogno je objašnjenju postavci i rešenju za robotsku ruku sa dva segmenta. Objašnjenje je dato za dvosegmentnu robotsku ruku jer je sistem koji se dobija za trosegmentnu ruku dosta složen, a samo rešenje (u opštim brojevima) zauzima desetine štampanih stranica.

### 3. Programska realizacija

Aplikacija **Robohan** koja je razvijena u okviru ovog rada koristi programski jezik Java. Ovaj programski jezik je odabran iz više razloga. Aplikacija je namenjena kao primer studentima, jedne od primena Grebnerovih baza, na kursu simboličke algebre. Sama distribucija aplikacije je morala biti takva da se omogući pristup što većem broju korisnika, nezavisno od platformi na kojima žele tu aplikaciju da pokrenu. Java web start tehnologija omogućava da se aplikacija pokreće direktno sa internet stranice nezavisno od operativnog sistema koji korisnik koristi. Java takođe poseduje veliki broj veoma razvijenih biblioteka za grafiku i zvuk, od kojih su neke korišćene u ovoj aplikaciji, što omogućava da se korisnicima na multimedijalan način prikaže željena materija.

U okviru aplikacije korisnicima se prikazuje kretanje robotske ruke na primeru sortiranja kutija po policama u određenom redosledu. Napomenimo da opšte primene robotskih ruku u industriji, koje se odnose na kratke i proste serije pokreta koji se ponavljaju, nisu predmet ovog rada.



Prelazimo na opis rada aplikacije. Uglovi koje svaki zglob ruke mora zauzeti su unapred izračunati za svaku policu. Moguće je implementirati algoritme za proračun uglova unutar same aplikacije. Napomenimo da takva implementacija bi zahtevala višestruko puta duže vreme izrade, a samo vreme izvršavanja aplikacije bi se uvećalo za nekoliko stepena veličine. U realnim uslovima se takva implementacija se ne koristi, pošto zahteva razvoj veoma složenih sistema, i korišćenje procesorske snage koja prevazilazi snagu većine računara ugrađenih u robotske ruke, pa samim tim i demonstracija zasnovana na takvom principu ne bi imala smisla.

Glavni ekran aplikacije, kao što je prikazano na Slici 3, se sastoji iz dve celine i menija koji sadrži deo kontrola toka izvršavanja. Najveći deo ekrana zauzima panel za iscertavanje rada robotske ruke. Za grafički deo aplikacije i grafički korisnički interfejs je korišćen programski paket *Swing* programskog jezika Java. Ovaj paket je veoma razvijen, stabilan, dobro dokumentovan i omogućava programiranje grafičkih elemenata na niskom nivou. Paket omogućava podešavanje opcija grafičkog hardvera, kako bi se u slučaju postojanja istog omogućilo efikasnije i prijatnije izvršavanje aplikacije. Kako aplikacija ne zahteva neke složene grafičke mogućnosti, korišćenje nekog naprednijeg paketa za grafiku u Javi bi samo dovelo do većeg opterećenja sistema na kome se aplikacija izvršava.

Sa desne strane ekrana se može videti deo u kom se vrši tekstualni prikaz koraka koje robotska ruka mora da izvrši sortiranje elemenata. Unutar tog polja se prikazuje koji element se namerava prebaciti na koju lokaciju, a zatim koje pokrete robotska ruka mora da načini kako bi se dovela u odgovarajuće položaje. Ovo korisnicima demonstrira izrazitu složenost zadatka koji se izvršava. Tu su takođe i kontrole za nasumični raspored kutija i pokretanje izvršavanja. Unutar gornjeg menija se nalaze kontrole za zvučne efekte i brzinu izvršavanja.

Zvučni efekti unutar aplikacije su postignuti na dva načina. Za uvodni i odjavni zvuk aplikacije je korišćena biblioteka *Javax.Sound* koja je sastavni deo osnovnog okruženja za pokretanje programskog jezika Java. Zbog kašnjenja koje se javlja između momenta zadavanja naredbe za zvuk, i momenta kada se zvuk čuje na zvučnicima, ova biblioteka nije bila primenjiva za generisanje zvuka motora unutar aplikacije. Kako bi se prevazišao ovaj problem, koji je veoma čest kod korišćenja unapred generisanih zvukova, korišćena je biblioteka *JSyn*. Ova biblioteka omogućava generisanje zvukova korišćenjem softverskih oscilatora i definisanjem funkcije po kojoj će dotični zvuk oscilovati. Korišćenjem dva oscilatora koji osciluju na dvema približnim frekvencijama, i primenom efekta izbijanja dobijen je karakterističan zvuk industrijskih mašina.

Napomenimo na kraju da opšti problem računanja Grebnerove baze je NP težine [4]. Pri tome je u literaturi poznato da postoje sistemi  $3 \times 3$  polinomskih jednačina čije rešavanje dovodi do problema sa radom Buhbergerovog algoritma usled javljanja numeričkih koeficijenata velike dužine, videti npr. [9].

## Zaključak

Na kraju rada napomenimo da neki modeli realnih robotskih ruku, koje se upotrebljavaju u industriji, koriste Grebnerove baze kao što je prikazano u [11]. Takođe, navedimo da razni problemi robotike su usko povezani sa Grebnerovim bazama [5], [7], [9] i [10].

Na Elektrotehničkom fakultetu u Beogradu primene Grebnerovih baza na probleme robotike se razmatraju u okviru delu kursa Simbolička algebra na master i doktorskim studijama [8]. U tom cilju razvijena aplikacija **Robohan** ima edukativni karakter i biće dostupna na sajtu <http://simba.etf.rs>

**Zahvalnica.** Istraživanje je delimično finansirano od strane Ministarstva za nauku i prosvetu, projekti broj III44006 i ON174032.



## Literatura

- [1] **B. Banjac.** *Java apleti za vizuelizacije u teoriji Grebner-ovih baza*, master rad, Elektrotehnički fakultet, Univerzitet u Beogradu, 2011. <http://simba.etf.rs>
- [2] **B. Banjac, V. Katić, A. Jovanović, A. Pejović, M. Makragić, I. Jovović, B. Malešević.** *Buhbergerov algoritam i vizuelizacija monomijalnih ideala*, Drugi simpozijum „Matematika i primene”, Matematički fakultet, Beograd, 27-28. maj 2011. <http://simba.etf.rs>
- [3] **V. Danilović.** *Java aplet za primene Grebnerovih baza na osnovne problem vezane za polinomske ideale*, master rad, Elektrotehnički fakultet, Univerzitet u Beogradu, 2011. <http://simba.etf.rs>
- [4] **K. Forsman.** *The Hitch Hiker's Guide to Gröbner's Bases*, Dept. Electrical Engineering Linköping University, Tehnical report 0374, 1992, 1-9.
- [5] **P. Grayson.** *Robotic Motion Planning*, MIT Undergraduate Journal of Mathematics, No.1, 1999, 57-68.
- [6] **A. Heck.** *A Bird's-Eye View of Gröbner Bases*, Nuclear Inst. and Methods in Physics Research A, No.389, 1997, 16-21.
- [7] **K. Kendricks.** *Solving the Inverse Kinematic Robotics Problem: A Comparison Study of the Denavit-Hartenberg Matrix & Groebner Basis Theory* (Dissertation), Graduate Faculty of Auburn University, 2007.
- [8] **B. Malešević, I. Jovović.** *Simbolička algebra* (kurs na master studijama Primenjena matematika), *Odabrana poglavlja simboličke algebre* (kurs na doktorskim studijama), Elektrotehnički fakultet, Beograd 2012. <http://simba.etf.rs>
- [9] **T. Pajdla.** *Advanced Robotics*, National Centre of Applied Cybernetics and Center for Machine Perception, Prague, 2010, <http://cmp.felk.cvut.cz/cmp/courses/PRO/2010/>
- [10] **F. Winkler.** *Gröbner bases at work: Inverse Kinematics in Robotics*, Computer Algebra RISC 2010. <http://www.risc.jku.at/education/courses/ws2011/ca/>
- [11] **W. Yan, H. Lu-bin, Y. Ting-li.** *Inverse Kinematics Analysis of General 6R Serial Robot Mechanism Based on Groebner Base*, Frontiers of Mechanical Engineering in China, Vol. 1, No. 1 (2006), 115-124.
- [12] **Maple** (trial version), Maplesoft.



# SKRIVENI MODELI MARKOVA KAO METODA ZA GENERACIJU SCENARIJA

**Milena Kresoja**

*Univerzitet u Novom Sadu, Prirodno-matematički fakultet  
e-mail, [milena.kresoja@dmi.uns.ac.rs](mailto:milena.kresoja@dmi.uns.ac.rs)*

**Apstrakt.** Jedno od najčešćih pitanja koje se postavlja u problemima finansijske optimizacije je svakako vezano za generaciju scenarija budućih realizacija vremenskih serija. Najpoznatije metode za generaciju su geometrijsko Braunovo kretanje i ARCH-GARCH modeli. Nažalost, sve do sada razvijene metode, uključujući navedene, imale su brojne nedostatke. Upravo zbog tih ograničenja ovi modeli nisu prihvaćeni kao adekvatni generatori scenarija i modelari finansijskih vremenskih serija. Najveći nedostatak geometrijskog Braunovog kretanja je jaka pretpostavka da su i drift i volatilitnost konstantni, dok je kod ARCH-GARCH modela malo oslabljena pretpostavka, volatilitnost je sada stohastička veličina, ali drift ostaje konstantan. Kako je cilj generisati model u kom su obe veličine stohastičke, u ovom radu opisana jedna od predloženih alternativa: model zasnovan na kombinaciji skrivenih lanaca Markova i geometrijskog Braunovog kretanja. ([1], [2],[6] ) Na taj način, parametri geometrijskog Braunovog kretanja (drift i volatilitnost) mogu da se menjaju kroz različita stanja lanca i stoga realistično opišu vremenskih serija.

**Keywords.** Geometrijsko Braunovo kretanje, skriveni modeli Markova, generacija scenarija

## 1. Uvod

Optimalan izbor rizičnih aktiva za dati kapital predstavlja osnovni problem u svetu finansija. Činjenica da su prinosi rizičnih aktiva neizvesni, povlači činjenicu da će i ishodi investiranja biti nepoznati. U takvim situacijama cilj je postići najbolju moguću to jest optimalnu kombinaciju: minimalan rizik i maksimalan očekivani prinos. Ipak neprekidne raspodele svakako otežavaju programiranje, pa je neophodno pojednostaviti modele kako bi se rešili problemi optimizacije. Jedan od načina da se to uradi jeste da se neprekidne raspodele zamene diskretnim raspodelama sa konačnim brojem mogućih ishoda. Ti ishodi se zovu scenariji, a sam postupak generacija scenarija. Dakle, nepoznati parametri modela predstavljaju slučajne promenljive, a njihove vrednosti su scenariji. U ovom radu je opisan generator scenarija, koji za razliku od do sada najpoznatijih modela uspeva da objasni karakteristike kretanja cena rizičnih aktiva. Generator se konstituiše pomoću skrivenih lanaca Markova. Skriveni lanci Markova su poseban model lanaca Markova, u kom figurišu dva stohastička procesa. Jedan proces je vezan za vremensku seriju koju želimo da opišemo, dok drugi osnovni stohastički proces opisuje stanje sistema. Kod ovog procesa ne mogu direktno registrovati njegove vrednosti pa se kaže da je proces skriven. Cilj je da se iskoristi ovaj model kako bi se modelirale finansijske vremenske serije i potom generisane scenarije iskoristiti u problemima finansijskog optimizacije. Motivacija potiče iz [1].

## 2. Skriveni modeli Markova kao metoda za generaciju scenarija

Kako bi se poboljšale do sada najpoznatije metode za opisivanje finansijskih vremenskih serija, u ovom poglavlju opisan je generator zasnovan na skrivenim Markovskim modelima. Kao što je napomenuto u uvodu, kod skrivenih lanaca Markova posmatramo dva stohastička procesa. To su: osnovni proces čije vrednosti možemo registrovati i proces koji opisuje stanje sistema i koji je skriven. Taj skriveni proces je klasični lanac Markova kod koga stanje sistema u nekom trenutku zavisi samo od stanja sistema u prethodnom trenutku, a ne od cele prošlosti. Dakle, proces čija budućnost ne zavisi od prošlosti već samo od sadašnjosti. Pretpostavićemo da sistem ima ukupno  $N$  stanja i da se u svakom trenutku može da se pređe iz jednog u bilo koje drugo stanje sa određenim verovatnoćama prelaza koje su nezavisne. Konstrukciju generatora započinjemo na sledeći način.

Pretpostavljamo da nam je dat prostor verovatnoće sa standardnom uređenom trojkom  $(\Omega, \mathcal{F}, P)$ . Pošto želimo da predviđamo cene rizičnih aktiva, potreban nam je proces koji ih opisuje. Taj niz aktiva označićemo sa  $\{S_k\}$ . Koristimo standardnu činjenicu da logaritam procesa prinosa tih rizičnih aktiva, u oznaci  $y_k = \ln \frac{S_k}{S_{k-1}}$ , prati sledeće kretanje:  $y_{k+1} = M_k + \Sigma_k z_{k+1}$ , gde su  $z_k$  nezavisne slučajne promenljive sa  $N(0, 1)$  raspodelom, proces  $\{M_k\}$  predstavlja proces drifta, dok proces  $\{\Sigma_k\}$  proces volatilnosti. Na ovaj način konstruisan je model po uzoru na geometrijsko Braunovo kretanje ali tako da su drift i volatilnost stohastičke veličine. Kako su u pitanju stohastički procesi ideja je da se  $\{(M_k, \Sigma_k)\}$  predstavi skriveni Markovski model. Najveća prednost ovakvog pristupa jeste što u svakom stanju postoje različiti parametri geometrijskog Braunovog kretanja. Motivacija je preuzeta iz [1] i [4].

Već je napomenuto da je skup stanja lanac  $\{(M_k, \Sigma_k)\}$  dimenzije  $N$ . Označićemo ga sa:

$$S = \{(\mu_1, \sigma_1), \dots, (\mu_N, \sigma_N)\}.$$

Matricu prelaza označavamo sa  $\Pi = [\pi_{ij}]_{i,j=1,\dots,N}$ , i važi

$$\pi_{ij} = P((M_{k+1}, \Sigma_{k+1}) = (\mu_{k+1}, \sigma_{k+1}) | (M_k, \Sigma_k) = (\mu_k, \sigma_k)).$$

Kako bi se ovaj model mogao upotrebiti, neophodno je oceniti parametre navedene parametre. Drugim rečima, moramo oceniti:

- broj stanja lanca  $N$ ,
- skup stanja lanca  $S$
- i verovatnoće prelaza  $\pi_{ij}, i, j = 1, \dots, N$ .

Za razliku ocena skupa stanja verovatnoća prelaza koje možemo izvesti rekursivno, pretpostavićemo da je broj elemenata skupa stanja unapred zadat. Na samom početku odmah možemo uočiti novu prepreku, a to je da je skup stanja dvodimenzionalan. Elementi skupa stanja u formi uređenih parova svakako otežavaju ocenjivanje. Ipak u literaturi je predložena transformacija pomoću injektivnog preslikavanja koje će skup stanja preslikati u skup vektora kanoničke baze prostora  $\mathbb{R}^N - (e_1, \dots, e_N)$ ; gde je  $e_i$  vektor dimenzije  $N \times 1$ , sa jedinicom na  $i$ -tom mestu i nulama na svim ostalim,  $i = 1, \dots, N$  i na taj način je znatno olakšana situacija. To jest, literatura sugerise sledeću transformaciju pomoću injekcije  $f, S \rightarrow \{e_1, \dots, e_N\}$  definisane sa  $f((\mu_i, \sigma_i)) = e_i; i = 1, \dots, N$ . Što se postojanja ove funkcije tiče, ona je obezbeđena jer  $\dim(S) = \dim(\{e_1, \dots, e_N\})$ . Uz pomoć ovog preslikavanja možemo da definišemo proces  $\{X_k\}$ :

$$X_k = f((M_k, \Sigma_k)).$$

Veoma važna činjenica je da i novi proces ima osobinu Markova, to jest:

$$P(X_{k+1} | X_k, X_{k-1}, \dots, X_0) = P(X_{k+1} | X_k).$$

Skup stanja novog lanca je  $\{e_1, \dots, e_N\}$ , a verovatnoće prelaza su iste kao kod lanca  $\{(M_k, \Sigma_k)\}$ .

Kako bi se model formalizovao neophodno je da se uvedu sledeće oznake za filtracije:

$$\begin{aligned} \{\mathcal{F}_k\} &= \sigma(X_0, X_1, \dots, X_k), \\ \{\mathcal{Y}_k\} &= \sigma(y_0, y_1, \dots, y_k) \end{aligned}$$

i

$$\{\mathcal{G}_k\} = \mathcal{F}_k \bigvee \mathcal{Y}_k = \sigma(X_0, X_1, \dots, X_k; y_0, y_1, \dots, y_k).$$

Neka je  $\mu = (\mu_1, \dots, \mu_N)^T$  i  $\sigma = \{\sigma_1, \dots, \sigma_N\}^T$ . Tada je,

$$(\mu_i, \sigma_i) = (\langle \mu, e_i \rangle, \langle \sigma, e_i \rangle); \quad i = 1, \dots, N.$$

Konačno dobijamo finalnu formu modela za generaciju scenarija:

$$y_{k+1} = \langle \mu, X_k \rangle + \langle \sigma, X_k \rangle z_{k+1} \quad (1)$$

$$X_{k+1} = \Pi X_k + V_{k+1}, \quad (2)$$

gde je  $\{V_{k+1}\}$  niz priraštaja martingala s obzirom na filtraciju  $\{\mathcal{F}_k\}$ .

## 2.1 Promena prostora verovatnoće

I pored transformacije modela opisane u prethodnom odeljku, model je ostaje suviše komplikovan. U nastavku je data ideja koja se provlači kroz literaturu vezanu za skrivene modele Markova, a ona je sadržana u promeni prostora verovatnoće. To jest, zamisao je da se uvede idealan prostor verovatnoće u kom bi se izvršila sva neophodna računanja, a potom svi dobijeni rezultati vratili u realni prostor verovatnoće uz pomoć filtera koji će se izvesti u međuvremenu. Data skica promene prostora verovatnoće, dok se detaljno izvođenje kao i dokazi teorema koje slede mogu pronaći u [4].

Dakle, cilj je uvesti novi prostor verovatnoće i samim tim novu meru verovatnoće. Da bi se ta ideja realizovala, prvo se moraju uvesti procesi:

$$\lambda_l = \frac{\phi\left(\frac{y_l - \langle \mu, X_{l-1} \rangle}{\langle \sigma, X_{l-1} \rangle}\right)}{\langle \sigma, X_{l-1} \rangle \phi(y_l)}$$

$$\Lambda_k = \prod_{l=1}^k \lambda_l, \quad \Lambda_0 = 1,$$

gde je  $\phi(z)$  funkcija gustine standardizovane normalne slučajne promenljive.

Nova mera verovatnoće  $\bar{P}$  se potom definiše na prostoru  $(\Omega, \bigvee_{l=1}^{\infty} \mathcal{G}_l)$  uz ograničenje da je Radon-Nikodym izvod  $\frac{d\bar{P}}{dP}$  na  $\sigma$ -polju  $\mathcal{G}_k$  baš  $\Lambda_k$ , to jest,  $\frac{d\bar{P}}{dP}|_{\mathcal{G}_k} = \Lambda_k$ . Ovo znači da za bilo koji skup  $B \in \mathcal{G}_k$ ,  $\bar{P} = \int_B \Lambda_k dP$ . Za bilo koju  $\mathcal{G}_k$  – merljivu slučajnu promenljivu  $R$ ,

$$\bar{E}(R) = \int R d\bar{P} = \int R \frac{d\bar{P}}{dP} dP = E(\Lambda_k R) \quad (3)$$

gde je sa  $\bar{E}$  označeno očekivanje pod merom  $\bar{P}$ .

**Teorema 1. (Uslovna Bajesova teorema).** [4] Neka je  $(\Omega, \mathcal{F}, P)$  prostor verovatnoća i  $\mathcal{G} \subset \mathcal{F}$  pod  $\sigma$  – algebra. Pretpostavimo dalje da je  $\bar{P}$  druga mera verovatnoća, apsolutno neprekidna sa obzirom na  $P$  i na Radon-Nikodim izvod  $\frac{d\bar{P}}{dP} = \Lambda$ . Tada, ako je  $R$  bilo koja  $\bar{P}$  integrabilna slučajna promenljiva važi

$$\bar{E}(R|\mathcal{G}) = \begin{cases} \frac{E(\Lambda R|\mathcal{G})}{E(\Lambda|\mathcal{G})}, & \text{ako } E(\Lambda|\mathcal{G}) \\ 0, & \text{inače} \end{cases}$$

Koristimo ovu teoremu za idealnu  $\bar{P}$  i realnu  $P$  verovatnoću, pa imamo sledeću lemu.

**Lema 1.** [4] Ako je  $\{R_k\}$   $\mathcal{G}$  - prilagođen niz integrabilnih slučajnih promenljivih, tada je

$$E(R_k|\mathcal{Y}_k) = \frac{\bar{E}(\Lambda_k R_k|\mathcal{Y}_k)}{\bar{E}(\Lambda_k|\mathcal{Y}_k)}. \quad (4)$$

Uvodimo oznaku  $\gamma_k(R_k) = \bar{E}(\Lambda_k R_k|\mathcal{Y}_k)$ .  $\gamma_k(R_k)$  je u suštini nenormalizovano uslovno očekivanje procesa  $R_k$  za dato  $\mathcal{Y}_k$ .

Prema Teoremi 1,

$$E(R_k|\mathcal{Y}_k) = \frac{\bar{E}(\Lambda_k R_k|\mathcal{Y}_k)}{\bar{E}(\Lambda_k|\mathcal{Y}_k)} = \frac{\gamma_k(R_k)}{\gamma_k(1)}.$$

Ako odaberemo  $\gamma_k(X_0) = E(X_0)$ ; dobijamo početne vrednosti za dalje rekurzije.

## 2.2. Rekurzivni filteri

Da bismo mogli sve rezultate da vratimo u stari prostor verovatnoće potrebno je da razvijemo odgovarajuće filtere. Detaljan opis izvođenja se može pronaći u [7].

Neka je dat niz  $\{R_k\}$  koji zadovoljava sledeće uslove:

$$\Delta R_{k+1} = R_{k+1} - R_k$$

i

$$\gamma_k(R_{k+1}) = \bar{E}(\Lambda_{k+1} R_k|\mathcal{Y}_{k+1}) + \bar{E}(\Lambda_{k+1} \Delta R_{k+1}|\mathcal{Y}_{k+1}).$$

Ako uočimo da važi

$$\bar{E}(\Lambda_{k+1} R_k|\mathcal{Y}_{k+1}) = \bar{E}\left(\Lambda_{k+1} R_k \frac{\phi\left(\frac{y_{k+1} - \langle \mu, X_k \rangle}{\langle \sigma, X_k \rangle}\right)}{\langle \sigma, X_k \rangle \phi(y_{k+1})} | \mathcal{Y}_{k+1}\right),$$

i uvdemo oznaku:

$$\Gamma^i(y_k) = \frac{\phi\left(\frac{y_k - \mu_i}{\sigma_i}\right)}{\sigma_i \phi(y_k)},$$

važi

$$\bar{E}(\Lambda_{k+1} R_k|\mathcal{Y}_{k+1}) = \sum_{i=1}^N \langle \gamma_k(R_k X_k), e_i \rangle \Gamma^i(y_{k+1}). \quad (5)$$

Dakle, da bismo dobili ocenu za  $\gamma_{k+1}(R_{k+1})$  moramo oceniti  $\gamma_k(R_k X_k)$ .

Uređena n-torka jedinica u daljem tekstu će biti označena sa  $\underline{1}$ . Kako je  $\langle X_k, \underline{1} \rangle = \sum_{i=1}^N \langle X_k, e_i \rangle = 1$ , pa

$$\langle \gamma_k(R_k X_k), \underline{1} \rangle = \gamma_k(R_k \langle X_k, \underline{1} \rangle) = \gamma_k(R_k). \quad (6)$$

Rečima, formula (6) govori da ocenu za  $\gamma_k(R_k)$  možemo dobiti sumiranjem elemenata  $\gamma_k(R_k X_k)$ .

Primetimo, ako je  $R_k = 1$ ,

$$\gamma_k(1) = \gamma_k(\langle X_k, \underline{1} \rangle) = \langle \gamma_k(X_k), \underline{1} \rangle = \bar{E}(\Lambda_k|\mathcal{Y}_k).$$

**Teorema 2.** [4] Neka je  $R_k$  skalarni  $\mathcal{G}$ -prilagođen proces sledeće forme

$$R_{k+1} = R_k + \alpha_{k+1} + \langle \beta_{k+1}, V_{k+1} \rangle + \delta_{k+1} f(y_{k+1}),$$

gde je  $V_{k+1} = X_{k+1} - \Pi X_{k+1}$ ,  $f$  skalarna funkcija,  $\alpha, \beta, \delta \mathcal{G}$  –predvidivi procesi  $\alpha, \delta$  su skalari, a  $\beta$  vektor dimenzije  $N \times 1$ ). Tada važi

$$\begin{aligned} \gamma_{k+1}(R_{k+1}X_{k+1}) &= \sum_{i=1}^N \langle \gamma_k(R_k X_k), e_i \rangle \Gamma^i(y_{k+1}) \pi_i + \gamma_k(\alpha_{k+1} \langle X_{k+1}, e_i \rangle) \Gamma^i(y_{k+1}) \pi_i \\ &+ \gamma_{k+1}(\delta_{k+1} \langle X_k, e_i \rangle) \Gamma^i(y_{k+1}) f(y_{k+1}) \pi_i + (\text{diag}(\pi_i) \\ &- \pi_i \pi_i^T) \gamma_k(\beta_{k+1} \langle X_{k+1}, e_i \rangle) \Gamma^i(y_{k+1}) \end{aligned}$$

gde je  $\pi_i = \Pi e_i$ .

Ukoliko u teoremi 2. stavimo  $R_k = R_0 = 1$ ;  $\alpha_k = 0$ ;  $\beta_k = (0, \dots, 0)^T$ ;  $\delta_k = 0$ ; dobijamo

$$\gamma_{k+1}(X_{k+1}) = \sum_{i=1}^N \langle \gamma_k(X_k), e_i \rangle \pi_i \Gamma^i(y_{k+1}).$$

### 2.3. Stanja, prelazi i vreme zadržavanja

Sada kada su izvedeni filteri, mogu se izvesti ocene nepoznatih parametra. Nephodno je samo, izvesti još ocenjivače za očekivani broj prelaza kao i vreme zadržavanja lanca u svakom stanju.

*Ocenjivač za broj skokova*

Polazimo od ocenjivača za broj prelaza između svaka dva stanja. Na samom početku, definisan je lanac Markova na sledeći način:  $X_k = \Pi X_{k-1} + V_k$ . Posmatramo dva proizvoljna stanja lanca  $e_r$  i  $e_s$ . Pretpostavićemo da ako se u vremenskom trenutku  $l - 1$  lanac nalazi u stanju  $e_r$  i u narednom trenutku prelazi u stanje  $e_s$ , tada brojimo 1, a inače 0. Matematički zapisano ukoliko dođe do navedene promene stanja važi da je  $\langle X_{l-1}, e_r \rangle \langle X_l, e_s \rangle = 1$ ; inače je 0. Ako posmatramo vremenski interval  $[0, k]$ , tada broj svih skokova iz stanja  $e_r$  u stanje  $e_s$  u datom intervalu možemo zapsisati kao:

$$J_k^{r,s} = \sum_{l=1}^k \langle X_{l-1}, e_r \rangle \langle X_l, e_s \rangle.$$

Koristimo teoremu 2 sa  $R_k = J_k^{r,s}$ ,  $R_0 = 0$ ,  $\alpha_k = \langle X_{k-1}, e_r \rangle \pi_{rs}$ ,  $\beta_k = \langle X_{k-1}, e_r \rangle e_s^T$ ,  $\delta_k = 0$  pa dobijamo,

$$\gamma_k(J_k^{r,s} X_k) = \sum_{i=1}^N [\langle \gamma_{k-1}(J_{k-1}^{r,s} X_{k-1}), e_i \rangle \Gamma^i(y_k) \pi_i + \langle \gamma_{k-1}(X_{k-1}), e_r \rangle \Gamma^i(y_k) \pi_{rs}].$$

*Ocenjivač za vreme zadržavanja*

Sada nas zanima vreme zadržavanja lanca u svakom od stanja. Posmatramo proizvoljno stanje  $e_r$  i hoćemo da vidimo koliko će puta u intervalu  $[0, k]$  lanac biti u tom stanju. Koristimo istu logiku kao u prethodnom slučaju. Računamo 1 ako i samo ako proces bude u datom stanju tokom intervala to jest  $\langle X_{l-1}, e_r \rangle = 1$  ako i samo ako  $X_{l-1} = e_r$ , inače je skalarni proizvod jednak 0.

$$O_k^r = \sum_{l=1}^k \langle X_{l-1}, e_r \rangle = O_{k-1}^r + \langle X_r, e_r \rangle.$$

Opet koristimo teoremu 2 sa odgovarajućim vrednostima, konkretno:  $R_k = O_k^r, R_0 = 0, \alpha_k = \langle X_k, e_r \rangle, \beta_k = (0, \dots, 0)^T, \delta_k = 0$  i dobijamo,

$$\gamma_k (O_k^r X_k) = \sum_{i=1}^N [\langle \gamma_{k-1} (O_k^r X_{k-1}), e_i \rangle \Gamma^i(y_k) \pi_i + \langle \gamma_{k-1} (X_{k-1}), e_r \rangle \Gamma^i(y_k) \pi_r.$$

*Ocenjivač za pomoćni proces*

Prethodna dva ocenjivača koristimo za izvođenje verovatnoća prelaza. Međutim, da bi se izvele ocene za nepoznate parametre geometričkog Braunovog kretanja neohodno je uvedemo još jedan pomoćni proces. Koristi se sledeći proces:

$$\mathcal{J}_k^r(h) = \sum_{l=1}^k \langle X_{l-1}, e_r \rangle h(y_k) = \mathcal{J}_{k-1}^r(h) + \langle X_{k-1}, e_r \rangle h(y_k),$$

gde  $h$  označava  $h(y) = y$  ili  $h(y) = y^2$ .

Opet uz pomoć teoreme 2 sa  $R_k = \mathcal{J}_k^r(h), R_0 = 0, \alpha_k = 0, \beta_k = (0, \dots, 0)^T, \delta_k = \langle X_k, e_r \rangle$  dobijamo:

$$\gamma_k (\mathcal{J}_k^r X_k) = \sum_{i=1}^N [\langle \gamma_{k-1} (\mathcal{J}_k^r(h) X_{k-1}), e_i \rangle \Gamma^i(y_k) h(y_k) \pi_i + \langle \gamma_{k-1} (X_{k-1}), e_r \rangle \Gamma^i(y_k) h(y_k) \pi_r.$$

Na ovaj način, izvedeni su filteri za  $\gamma_k (\mathcal{J}_k^{r,s} X_k), \gamma_k (O_k^r X_k), \gamma_k (\mathcal{J}_k^r X_k)$ . Kako su nam neophodni filteri za  $\gamma_k (\mathcal{J}_k^{r,s}), \gamma_k (O_k^r), \gamma_k (\mathcal{J}_k^r)$ , na osnovu prethodnog poglavlja znamo da ih možemo lako dobiti sumiranjem komponenti pomenutih sračunatih filtera. To jest,  $\gamma_k (\mathcal{J}_k^{r,s}) = \langle \gamma_k (\mathcal{J}_k^{r,s} X_k), \underline{1} \rangle, \gamma_k (O_k^r) = \langle \gamma_k (O_k^r X_k), \underline{1} \rangle, \gamma_k (\mathcal{J}_k^r) = \langle \gamma_k (\mathcal{J}_k^r X_k), \underline{1} \rangle$ .

## 2.4. Ocena parametara maksimiziranjem očekivanja

Sada kada su izvedeni ocenjivači, konačno se može pristupiti oceni nepoznatih parametara. Postoje mnoge metode za ocenu parametara skrivenih modela Markova, među kojima se kao najfunkcionalnija izdvaja metoda maksimiziranja očekivane vrednosti funkcije verodostojnosti. Maksimiziranje očekivanja je iterativna metoda koja se sastoji iz sledeća dva koraka, računanja očekivane vrednosti logaritma funkcije verodostojnosti s obzirom na trenutnu ocenu distribucije neopažene promenljive i računanja parametara koji maksimiziraju izračunatu očekivanu vrednost. [7]. Pomoću ove metode dobijamo da su ocene za data opažanja do vremena  $k$  verovatnoća prelaza  $\widehat{\pi}_{rs}(k)$  date sa:

$$\widehat{\pi}_{rs}(k) = \frac{\gamma_k (\mathcal{J}_k^{r,s})}{\gamma_k (O_k^r)},$$

vektora drifta  $\widehat{\mu}_r(k)$ ,

$$\widehat{\mu}_r(k) = \frac{\mathcal{J}_k^r(y_k)}{\gamma_k (O_k^r)}$$

i vektora volatilnosti  $\widehat{\sigma}_r(k)$ ,

$$\widehat{\sigma}_r(k) = \sqrt{\frac{\gamma_k (\mathcal{J}_k^r(y_k^2)) - 2\widehat{\mu}_r(k)\gamma_k (\mathcal{J}_k^r(y_k)) - \widehat{\mu}_r^2(k)\gamma_k (O_k^r)}{\gamma_k (O_k^r)}}$$

Kada se dobiju ocene parametara, metod je spreman za integraciju u generaciji scenarija prinosa rizičnih aktiva.

Pomoću ocena koje smo izveli za matricu prelaza, drift i volatilnost, možemo predvideti cenu rizične aktive za

sledeći dan trgovanja. Dakle, cilj nam je da izvedemo formulu  $E(S_{k+1} | \mathcal{Y}_k, \Pi, \mu, \sigma)$  Važi:



$$\begin{aligned}
E(S_{k+1}|\mathcal{Y}_k, \Pi, \mu, \sigma) &= E(S_{k+1}e^{y_{k+1}}|\mathcal{Y}_k, \Pi, \mu, \sigma) = S_{k+1}E(e^{y_{k+1}}|\mathcal{Y}_k, \Pi, \mu, \sigma) \\
&= S_{k+1}E\left(\sum_{i=1}^N \langle X_k, e_i \rangle e^{\langle \mu, X_k \rangle + \langle \sigma, X_k \rangle z_{k+1}} \middle| \mathcal{Y}_k, \Pi, \mu, \sigma\right) \\
&= S_{k+1}E\left(\sum_{i=1}^N \langle X_k, e_i \rangle \middle| \mathcal{Y}_k, \Pi, \mu, \sigma\right) E(e^{\mu_i}|\mathcal{Y}_k, \Pi, \mu, \sigma) E(e^{\sigma_i z_{k+1}}|\mathcal{Y}_k, \Pi, \mu, \sigma).
\end{aligned}$$

Dalje znamo,  $E(\langle X_k, e_i \rangle | \mathcal{Y}_k, \Pi, \mu, \sigma) = \langle E(X_k | \mathcal{Y}_k, \Pi, \mu, \sigma), e_i \rangle$ . Kako je  $\mu_i, i = 1, \dots, N$  nezavisno od  $\{\mathcal{Y}_k, \Pi, \mu, \sigma\}$  imamo  $E(e^{\mu_i} | \mathcal{Y}_k, \Pi, \mu, \sigma) = e^{\mu_i}$ . Slično,  $\sigma_i$  je nezavisno  $\{\mathcal{Y}_k, \Pi, \mu, \sigma\}$  pa još uz činjenicu,  $z_{k+1}, N(0, 1)$  dobijamo  $E(e^{\sigma_i z_{k+1}} | \mathcal{Y}_k, \Pi, \mu, \sigma) = e^{\frac{\sigma_i^2}{2}}$ . Konačno, dobijamo  $E(S_{k+1} | \mathcal{Y}_k, \Pi, \mu, \sigma) = S_k \sum_{i=1}^N e^{\mu_i} e^{\frac{\sigma_i^2}{2}} \langle \gamma_k(X_k), e_i \rangle$ .

### 3. Numerički eksperiment

Pretpostavimo da se skup aktiva sastoji iz cena indeksa Standard & Poor 500 (S&P 500). Posmatrali smo dnevne cene tokom 2010. godine (ukupno 252 radnih dana). Numerički eksperiment je imao sledeći tok. Koristili smo skrivene modele Markkova kako bismo generisali scenarije ove finansijske serije. Kako bi se uverili u superiornost modela, generisali smo scenarije za istu seriju ali pomoću GARCH(1,1) modela, i izvršili poređenje dobijenih rezultata.

Kao što je spomenuto na samom početku, ocene za broj stanja lanca Markova se ne izvode već se unapred zadaju. Pretpostavićemo da je broj stanja 3. To je saglasno sa rezultatima koje su izveli Messina i Toscani. Oni su, naime, uz pomoć kako statističkih tehnika tako i ispitivanjem podataka ocenili dimenziju skupa stanja na 3 stanja i pokazali da model sa 3 stanja daje dobru reprezentaciju cena rizičnih aktiva.[6]. Intuitivno tri stanja možemo posmatrati kao tri tipična stanja ekonomije: dobro, srednje, loše.

Uz pomoć navedenih tehnika iz prethodnih poglavlja, dobijamo sledeće ocene parametra.

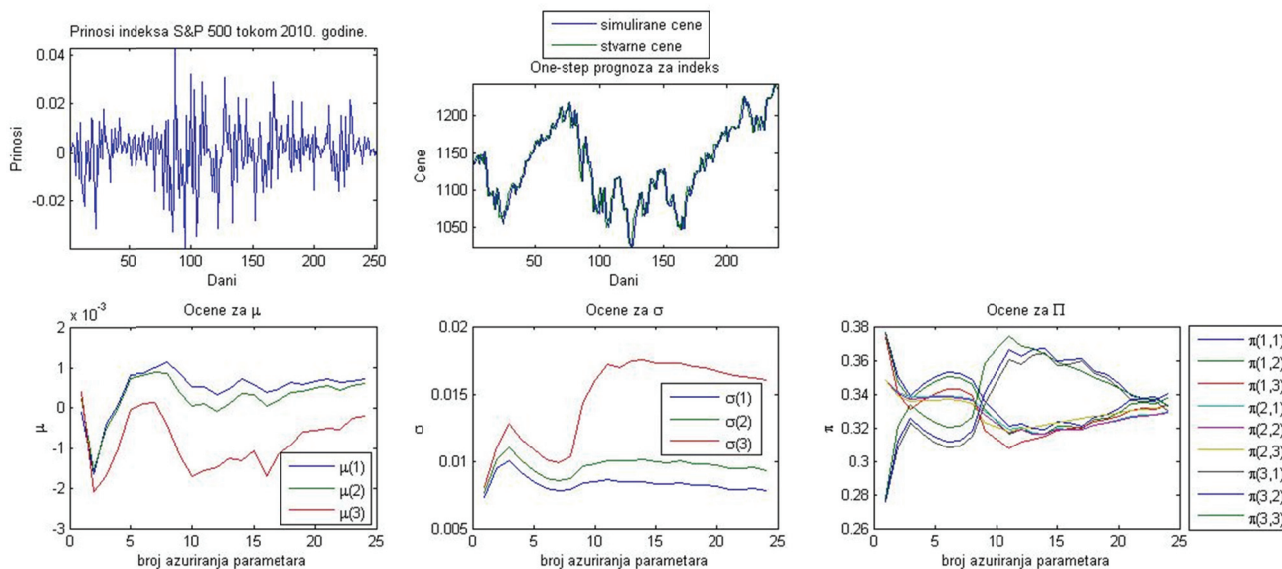


Figura 1. Ocena parametara za indeks S&P 500.

Optimalna ocena matrice verovatnoća prelaza je:

$$\hat{\Pi} = \begin{bmatrix} 0.3404 & 0.3380 & 0.3336 \\ 0.3295 & 0.3292 & 0.3327 \\ 0.3300 & 0.3329 & 0.3337 \end{bmatrix}$$

Optimalna ocena vektora drifta je  $\hat{\mu} = (0.7106 \cdot 10^{-3}, 0.6044 \cdot 10^{-3}, 0.2022 \cdot 10^{-3})^T$ , a vektora volatilnosti  $\hat{\sigma} = (0.0078, 0.0093, 0.0160)^T$ . Pored ocena parametara, na slici 1 predstavljeno je kretanje prinosa indeksa tokom posmatranog perioda kao i kretanje cena indeksa zajedno sa simuliranim cenama. Ukoliko pogledamo prinose vidimo da takvo ponašanje sugerise da bi pogodan izbor za opisivanje ovog procesa bio model skrivenih lanaca Markova koji karakterisu visok i srednji nivo volatilnosti. Ukoliko posmatramo ocene parametara lanca Markova, možemo videti da svi parametri konvergiraju.

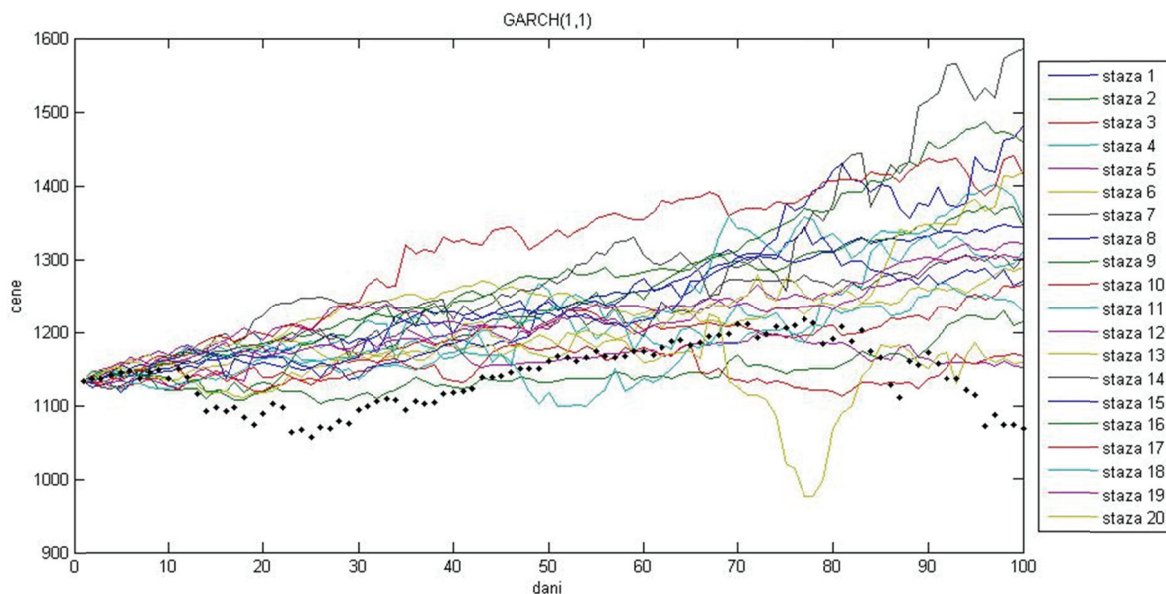


Figura 2. Simulirane cene indeksa S&P 500 pomoću GARCH(1,1) modela, sa 20 staza tokom 100 dana.

Generišemo 20 staza za posmatranu vremensku seriju pomoću oba metoda za 100 dana. Stazu cene aktiva simuliramo pomoću optimalnih ocena parametara. Na slikama 2 i 3 su prikazane staze koje smo dobili. Crnim tačkicama je označeno kretanje stvarnih cena indeksa. I zaista, možemo uočiti da se generator skrivenih lanaca Markova pokazuje kao razuman izbor. Vidimo da su predviđaja koja daje GARCH(1,1) model precenjena, za razliku od modela skrivenih lanaca koja su realističnija, ali sa nekim ekstremno niskim ili visokim predviđanjima.

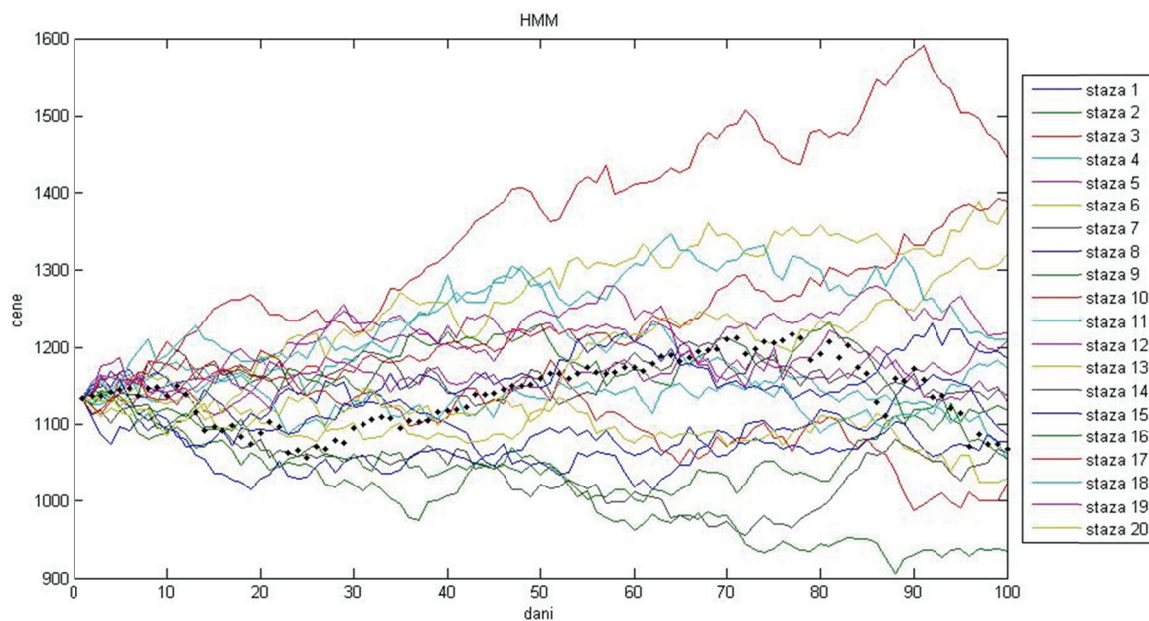


Figura 3. Simulirane cene indeksa S&P 500 pomoću modela skrivenih lanaca Markova, sa 20 staza tokom 100 dana.

#### 4. Zaključak

U ovom radu je ukratko opisana jedna od predloženih alternativa tradicionalnim pristupima modeliranja finansijskih vremenskih serija za generaciju scenarija. Predstavljen je pristup zasnovan na skrivenim modelima Markova. Osnovna pretpostavka modela jeste da je proces zadržava formu geometrijskog Braunovog kretanja, ali sa pretpostavkama koje su slabije. Parametri tog procesa mogu da se menjaju u različitim stanjima lanca. Predloženi model se dakle ne razlikuje fundamentalno od ranijih metoda, ali uspeva da objasni važne osobine vremenskih serija kao što su ekstremna kretanja i grupisanje volatilnosti. U numeričkom eksperimentu, pokazano je da su rezultati implementacije više nego značajni, kao što je neosporna i bolja generacija od GARCH modela. Parametri modela su ocenjeni na osnovu tehnike rekurzivnih filtera. Nakon ocene parametara, na osnovu istorijskih podataka, mogu se generisati staze senarija, što je i urađeno u ovom radu.

#### Literatura.

- [1] Erlwein C., Mitra G., Roman HMM based scenario generation for an investment optimisation problem. Springer Science+Business Media, 2011.
- [2] Zhang Y. Prediction of financial time series with Hidden Markov Models. M.Sc Thesis, Simon Fraser University, China,2004.
- [3] Elliot R. J, Aggoun L., Moore J. B. Hidden Markov models,estimation and control. Springer, 1995.
- [4] Mamom R. S., Elliot R. J. Hidden Markov Models in finance. Springer,2007.
- [5] Elliott R., Van der Hoek J. An application of hidden Markov models to asset allocation problems. Finance and Stochastics, vol. 1, 1997.
- [6] Messina E., Toscani Hidden Markov models for scenario generation. IMA Journal of Management Mathematics 19, 397-401,2008.
- [7] Jalen L. Some contributions to filtering theory with applications in financial modelling. PhD Thesis, Brunel University, 2009.
- [8] <http://finance.yahoo.com/q/hp?s=G^SPC+Historical+Prices>

# ПРИМЕНЕ ПСЕУДО-СЛУЧАЈНОСТИ У ТЕОРИЈИ ВЕРОВАТНОЋА

Богдан Ободић

Математички факултет, Универзитет у Београду, Србија.

*mv09058@alas.matf.bg.ac.rs*

**Апстракт.** Циљ овог рада је класификација генератора псеудо-случајних бројева ради њихове лакше примене у симулацијама случајних догађаја у теорији вероватноће. Како ниједан генератор псеудо-случајних бројева није савршен, из простог разлога што је детерминистичке природе, оваква класификација нам даје увид у предности и мане генератора. Само добрим одабиром генератора, у зависности од проблема пред којим се налазимо, можемо постићи најбоље резултате. Такође, биће нам много лакше да се одлучимо између квалитета неке оцене и брзине оцењивања вероватноћа. Може се констатовати да услови у којима се неки случајни догађаји одигравају јесу неприступачни за мерење, превише скупи, или пак непрактични. Због тога је симулација случајних догађаја некада неизбежна, а извор псеудо-случајности на детерминистичкој машини није увек погодан за рад. Брз развој информacionих технологија које се преплићу са другим научним дисциплинама дају све већи допринос у избору бољих и ефикаснијих извора псеудо-случајности, где због њихових уских примена за конкретну намену морамо имати увид у све, не бисмо ли одабрали нама најбољи.

**Кључне речи:** псеудо-случајност, генератори псеудо-случајних бројева, симулација, вероватноћа

## 1. Увод

Симулирање стохастичких процеса на рачунарима захтева неки извор случајности. У пракси, бројеви који изгледају случајни се на рачунару генеришу потпуно детерминистичким алгоритмима, и зато то никако не могу бити прави случајни бројеви. Еквивалентно, ако би се ови алгоритми извршили у друго време или на другим рачунарима са истим почетним стањем интерних података, добијени низ бројева би био потпуно исти. То је разлог зашто ове бројеве називамо псеудо-случајним бројевима. Ако су добијени рандомизираним алгоритмима<sup>[8]</sup>, такве алгоритме називамо генераторима псеудо-случајних бројева. Како је праву случајност немогуће добити на детерминистичком рачунару, задовољавамо се генераторима псеудо-случајних бројева таквим да добијени низ бројева пролази већину статистичких тестова случајности. Сви генератори псеудо-случајних бројева падају на неком статистичком тесту. Пошто не постоји универзална колекција тестова чији пролаз гарантује да је дати генератор потпуно поуздан, генератори се најчешће конструишу за одређену намену и морамо бити пажљиви при одабиру генератора да бисмо осигурали најбољи резултат. Због тога је оваква класификација псеудо-случајних генератора неопходна. Претпоставимо да нам требају псеудо-случајни природни бројеви између 0 и  $m \in \mathbb{N}$ , униформно распоређени. Да бисмо добили такав дискретни низ на детерминистичкој машини, потребан нам је неки генератор псеудо-случајних бројева. Поред класификације, у овом раду ће бити наведене и предности и мане сваког од генератора, као и начини за њихово побољшање у односу на неке већ постојеће. Ако нам пак требају реални униформно распоређени бројеви из нпр. интервала  $[0,1]$ , најлакше је да малопређашњи дискретни низ нормирамо са  $m$ . Ако хоћемо да нам псеудо-случајни бројеви прате неку другу расподелу, нормалну, геометријску, Пуасонову..., то је могуће постићи одређеним трансформацијама униформног псеудо-случајног низа<sup>[2]</sup>.

## 2. Генератори псеудо-случајних низова

Неформално, псеудо-случајни генератор је алгоритам  $G: \{0,1\}^t \rightarrow \{0,1\}^m$  где је  $m$  много веће од  $t$  тако да за униформно изабрано  $x \in \{0,1\}^t$ , расподела  $G(x)$  је псеудо-случајна, тј. „личи“ на униформну преко  $\{0,1\}^m$ . Ако је  $U_m$  случајна променљива униформно расподељена на  $\{0,1\}^m$  и  $G$  псеудо-случајни генератор, онда је  $G$  добар псеудо-случајни генератор ако су  $G(U_t)$  и  $U_m$  статистички близу.

**Дефиниција 1.** Статистичка дистанца је мера растојања између два статистичка објекта, нпр. узорка, случајне променљиве, расподеле вероватноћа... Облика је  $d: X^2 \rightarrow \mathbb{R}$ . Ако су статистички објекти у некој  $(K, \varepsilon)$ -околини, рећи ћемо да су статистички близу.

**Дефиниција 2.** Две расподеле  $\mu_X$  и  $\mu_Y$  на  $\{0,1\}^m$  се  $(K, \varepsilon)$ -не разликују ако

$$(\forall T \subseteq \{0,1\}^m) (|P_{X \sim \mu_X} \{X \in T\} - P_{X \sim \mu_Y} \{Y \in T\}| \leq \varepsilon) \quad (1)$$

**Дефиниција 3 (псеудо-случајност).** Расподела  $\mu_X$  преко  $\{0,1\}^m$  је  $(K, \varepsilon)$ -псеудо-случајна ако је  $(K, \varepsilon)$ -неразличита од униформне расподеле, т.ј. ако

$$(\forall T \subseteq \{0,1\}^m) \left( \left| P_{X \sim \mu_X} \{X \in T\} - \frac{|T|}{2^m} \right| \leq \varepsilon \right) \quad (2)$$

**Теорема 1 (Грин-Тао-Цеглер).** Ако је  $R$  псеудо-случајни скуп, т.ј. скуп са псеудо-случајном расподелом и  $D \subseteq R$  густ у  $R$ , онда се скуп  $D$  може измоделовати скупом  $M$  који је густ на целом домену тако да се скупови  $D$  и  $M$   $(K, \varepsilon)$ -не разликују.

### 2.1. Линеарни конгруентни генератор (ЛКГ)

ЛКГ је један од најчешће коришћених и најпростијих униформних генератора. Свакако није и најбољи, али његова предност се огледа у томе што даје јако брзу оцену, иако је она некада превише груба. За почетну вредност генератора узимамо неко  $x_0$ ,  $0 < x_0 < m$ , те ЛКГ израчунава низ природних бројева  $x_1, x_2, x_3, \dots$  између 0 и  $m-1$  на основу рекурентне формуле

$$x_{i+1} = ax_i + b \pmod{m}, \quad i=0,1,2,\dots \quad (3)$$

где су  $a$ ,  $1 < a < m$ , пажљиво изабрани природни бројеви. Касније ће бити назначено које вредности за  $a$  и  $m$  ће нам дати најбрже резултате и најкорисније, док за сада можемо рећи  $b \stackrel{\text{def}}{=} 0$ , јер из његове линеарности не следи никаква боља расподељеност чланова низа, а при  $b=0$  можемо само добити на ефикасности, т.ј. брзини, што је и главна одлика овог генератора. Број  $x_0$  називамо семе (енгл. seed). За вредност  $x_0=0$  добијамо нула низ, те нема сврхе рачунати га. Напоменимо само још да за просто  $m$  чланови низа никада нису нула.

ЛКГ су брзи генератори и захтевају минимално меморије. То их чини употребљивим а симулацију више независних токова. Наиме, ЛКГ не би требали да се примењују тамо где је случајност високог квалитета неопходна. Ипак, ЛКГ могу бити добар избор за оно што нам је потребно. Нпр. у кућним рачунарима је количина меморије често врло ограничена, па су ЛКГ одлични за такве услове. Због тога су јако применљиви и у играчким конзолама где је потребан неки извор псеудо-случајности. Битове нижег реда ЛКГ-а када је  $m$  степен двојке не би требало користити ни за један степен случајности. Још један од недостатака ЛКГ-а је да имају висок степен корелације, па нису погодни за Monte Carlo или Las Vegas методе. Ако бисмо их нпр. користили да добијемо тачке у  $n$ -димензионом простору, оне би најчешће лежале у мање од  $(n! \cdot k)^{1/n}$  хиперравни, где је  $k$  степен двојке и  $m$  облика  $2^k$ . То чак директно следи из Марсаглијеве теореме<sup>[3]</sup>.

**Пример 1.** Нека је  $m=11$ ,  $a=3$ , семе  $x_0=1$ . Коришћењем ЛКГ-а добијамо низ  
 $3,9,5,4,1,3,9...$

**Напомена 1.** Када се први пут неки број генерише по други пут (у Примеру 1 је то број 3), добијамо подниз који се понавља, тзв. период.

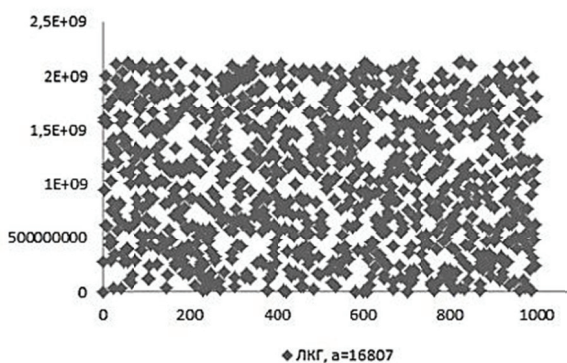
Посматрајмо још мало Пример 1: дужина периода низа је 5. Уопшtimo то сада. Пошто имамо  $m$  могућих вредности за чланове низа најдужи могући период је  $m$ . Како је већ речено да за просто  $m$  никад не добијамо нулу, за такво  $m$  минимални период је  $m-1$ .

**Пример 2.** Нека је  $m=11$ ,  $a=7$ , семе  $x_0=1$ . Коришћењем ЛКГ-а добијамо низ  
 $7,5,2,3,10,4,6,9,8,1,7,5...$

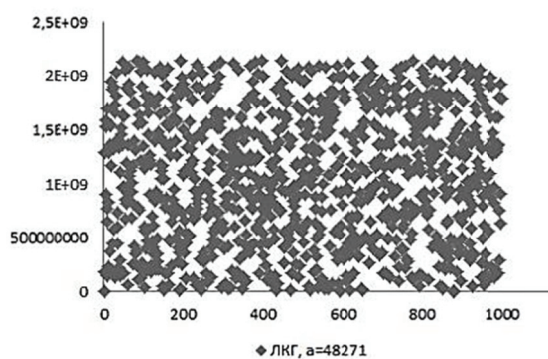
**Напомена 2.** У овом случају је период  $m-1=11-1=10$ , што је најбоља могућа вредност за овако одабране почетне вредности. Дакле за  $m$  прост број разне вредности  $a$  дају пун период независно од вредности семена  $x_0$ .

**Примедба 1.** На рачунару 32-битне архитектуре један бит је резервисан за знак, па је  $m$  максималне дужине 31 бит. Највећи записив 31-битни означени број је  
 $2^{31}-1=2147483647$

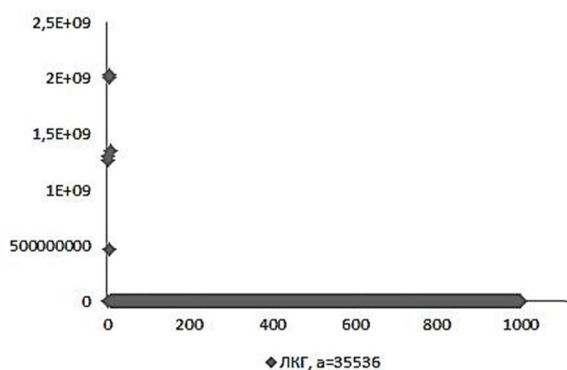
а како је тај број Мерсенов број, он је и прост. Најбољу униформну расподељеност псеудо-случајних низова добијамо за  $a=16807$  и  $a=48271$  (Слике 1 и 2). То се може приметити и са графикона расипања. Ако уместо 1000, за дужину низа узмемо 100000 (Слика 5), видимо потпуну униформност, и у вероватноћи можемо бити задовољни оваквим низом. Овде такође емпиријски видимо и закон великих бројева, јер за већи узорак добијамо бољу расподељеност. Сада, ако смо се одлучили за овај генератор, можемо симулацијама решити неке од проблема у вероватноћи.



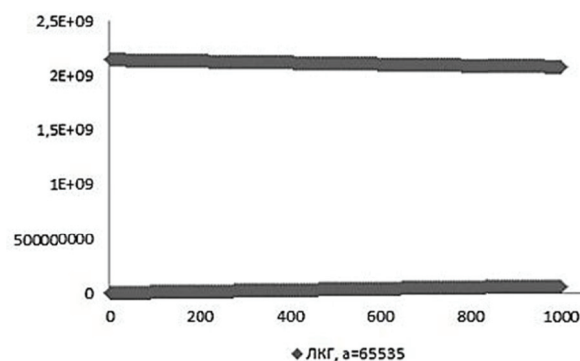
Слика 1. Графикон расипања за MINSTD



Слика 2. Графикон расипања за MINSTD

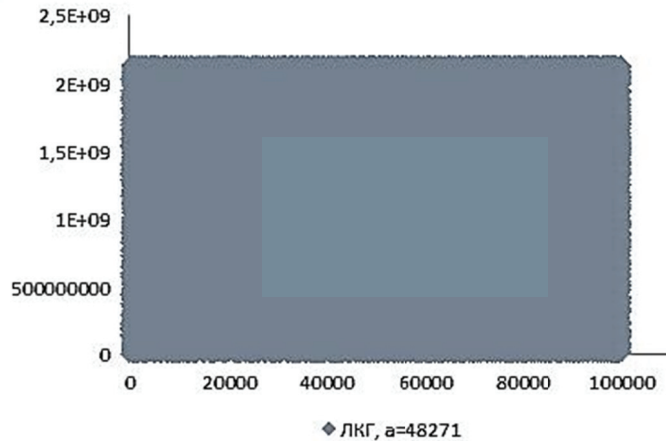


Слика 3. Графикон расипања за произвољно  $a$



Слика 4. Графикон расипања за произвољно  $a$





Слика 5. MINSTD за n=100000

**Пример 3.** ЛКГ-ом симулирајмо бацање новчића 20 пута.

Добили смо: Г,П,Г,П,П,П,Г,Г,П,П,Г,П,Г,П,Г,Г,П,Г,П,П  
 Новим покретањем: Г,П,Г,П,Г,Г,П,П,П,Г,Г,Г,П,П,Г,П,П,Г,Г,Г  
 И још једном: П,П,Г,Г,Г,Г,Г,Г,П,П,П,Г,П,Г,П,Г,П,П,Г,Г  
 И још само једном: Г,П,П,Г,П,Г,П,Г,Г,Г,П,П,Г,П,П,П,П,П

Добар показатељ псеудо-случајности је и то да смо при сваком покретању добили други низ, управо због различитих семена.

**Пример 4.** ЛКГ-ом симулирајмо бацање коцкице 10 пута.

Добили смо: 1,5,6,2,1,3,4,2,3,2  
 Новим покретањем: 5,1,2,2,1,6,2,6,6,1  
 И још једном: 1,2,5,4,5,6,1,1,1,2  
 И још само једном: 2,1,3,4,1,3,6,2,6,3

## 2.2 Мерсенов твистер (МТ)

МТ је униформни генератор псеудо-случајних бројева. Развили су га Макото Матсумото и Такуји Нишимура 1997. године. Његове главне предности су брзина и робустност, као и јако велики период од  $2^{19937}-1$ . Како је  $2^{19937}-1$  Мерсенов прост број, он је заслужан за име овог генератора. Уједно, псеудо-случајни бројеви које МТ генерише пролазе врло строге статистичке тестове. Ово долази уз цену да МТ користи 624 меморијске речи, док ЛКГ користи само једну меморијску реч (32-битну или 64-битну најчешће).

МТ генерише низ бинарних вектора дужине  $w$ , који појединачно представљају униформне псеудо-случајне бројеве из интервала од 0 до  $2^w - 1$  у бинарном запису.

Формула МТ-а је такође рекурентног облика:

$$x_{k+n} = x_{k+m} + x_{k+1} * \begin{bmatrix} 0 & 0 \\ 0 & I_r \end{bmatrix} * A + x_k * \begin{bmatrix} I_{w-r} & 0 \\ 0 & 0 \end{bmatrix} * A, \quad k + 0, 1, 2, \dots \quad (4)$$

где су  $I_r$  и  $I_{w-r}$  јединичне матрице реда  $r$  и  $w-r$ ,  $0$  је нула матрица одговарајућег реда и  $A$  је матрица реда  $w$ . Уочимо да нам треба  $n$  почетних вектора (семена)  $x_0, x_1, \dots, x_{n-1}$ . Тада редом за  $k = 0, 1, 2, \dots$  добијамо остале векторе низа  $x_n, x_{n+1}, x_{n+2}, \dots$

Простији облик горње рекурентне једначине који је много лакши за имплементирање је:

$$x_{k+n} = x_{k+m} + x_{k+1} \text{ XOR}((x_{w-1}^k, \dots, x_r^k, x_{r-1}^{k+1}, \dots, x_0^{k+1}) * A), \quad k + 0,1,2, \dots \quad (5)$$

Најефикаснији кандидат за  $A$  је облика 
$$\begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ a_{w-1} & a_{w-2} & a_{w-3} & \dots & a_0 \end{bmatrix}.$$

Ови параметри се деле у две групе

- 1) Параметри периода, одређују период генерисаног низа бројева. То су цели бројеви  $w$  (величина меморијске речи),  $n$  (ред рекурентне једначине),  $m$  (средњи елемент) и  $r$  (деобна тачка меморијске речи), као и један бинарни вектор  $a$  (матрица  $A$ ).
- 2) Параметри униформности, одређују квалитет униформности генерисаног низа бројева. Ту спадају цели бројеви који се користе при шифтовању у имплементацији и узимају вредности од 1 до  $w-1$ .

Разлог због велике популарности МТ алгоритма данас је једноставно због тога што га технологија подржава. Када је настао, због своје тешке имплементације, био је спор, и многи су га критиковало, укључујући и Џорџа Марсаглија. Данас ћемо ипак рећи да је овај алгоритам брз. Његова највећа предност је наравно дужина његовог периода. Ту је такође висок степен униформности. Псеудо-случајан низ добијен МТ-ом је  $K$ -униформно расподељен за свако  $K$  из [1, 623]. Генерално, високе перформансе овог алгоритма следе из количине меморије коју он троши. Иако то јесте много, он је троши ефикасно. Због тога је МТ default генератор многих значајних програмских пакета, као што су нпр: MATLAB, R, Maple, као и у програмском језику Python, и другим.

**Пример 5.** МТ-ом симулирајмо покретање рулета 10 пута.

Добили смо:

26 19 18 20 3 18 28 17 21 00 21 7 23 0 36 7 10 0 1 23 28 19 3 35 29 4 16 26 22 00 26 23 30 23 2  
10 30 27 35 33 36 36 15 33 4 4 32 21 00 1 7 6 22 20 18 23 26 20 21 9 24 18 16 15 34 36 29 22 4 18  
31 28 1 8 3 23 34 32 9 16 6 11 11 13 10 23 35 25 18 3 00 30 36 9 14 3 23 21 15 16

Нека 0 и 00 значе да је кућа добила. У овом низу имамо 6 пута појављивање 0 или 00. Теоријски, треба да имамо  $100 * 2/38 = 5,26$ . Дакле близу смо очекиваној фреквенцији. Покренимо програм поново. Добијемо низ:

15 4 4 6 25 16 11 6 28 10 13 1 33 5 33 5 33 22 5 3 20 9 28 35 9 34 24 1 22 33 00 10 14 18 28 32 3  
9 14 21 23 26 11 0 1 3 22 9 19 14 27 29 1 6 22 7 1 15 19 17 29 00 34 32 16 00 7 27 11 24 22 4 4 33  
26 14 27 12 27 28 34 4 8 10 28 14 20 33 31 21 35 00 19 24 13 9 7 10 34 21

Имамо 5 појављивања 0 или 00. Покренимо сада 100 пута овај генератор и оценимо појављивање 0 или 00. Добијено је 552 појављивања 0 или 00 укупно, што је 5,52 пута по низу просечно. Покренимо га сада 1000 пута. Добијено је 5212 појављивања укупно, или 5,21 просечно. Покренимо генератор сада чак 10000 пута. Добијемо 52558 појављивања укупно, или 5,26 пута просечно, што је идентично са теоријски добијеним резултатом до на две децимале. Одавде можемо емпиријски закључити да је овај генератор добар.

**Пример 6.** Неки стрелац погађа мету са вероватноћом од 0,2. Ако при сваком поготку добије 1 поен, применом МТ -а одредити вероватноћу да освоји 10 поена.

Добили смо да је 10-и погодак стигао у 46-ом гађању, чиме је вероватноћа догађаја 0,22. У следећем покретању смо добили да је 10-и погодак стигао тек у 30-ом гађању, чиме је вероватноћа догађаја 0,33. Како је ово неконзистентно, израчунајмо просечну вероватноћу у 100 покретања генератора. Просечно је потребно 50,33 гађања да би се освојило 10 поена, тј. вероватноћа догађаја је 0,19. То се још боље види и када генератор покренемо 10000 пута.



Тада вероватноћа догађаја износи 0,200556, тј потребно је око 49,861 гађања. Ако гађања заокружимо на цео број као што то и јесу, добијамо 50. Вероватноћу сво већ добили идентичну теоријској до на две децимале.

### 2.3. Blum Blum Shub (ББШ)

ББШ је алгоритам који су направили Ленора Блум, Мануел Блум и Михаел Шаб 1986 године. Облика је:

$$x_{i+1} = x_i^2 \pmod{m} \quad (6)$$

где  $m$  представља умножак два проста броја  $p$  и  $q$ . Семе  $x_0$  не би требало да буде 1 или неки други цео број дељив са  $m$ . Оба проста броја  $p$  и  $q$  би требала бити конгруентна броју 3 по модулу 4, да би сваки квадратни остатак имао бар један корен који је такође квадратни остатак. Што је мањи заједнички делитељ бројева  $\phi(p-1)$  и  $\phi(q-1)$ , то ће период псеудо-случајног низа бити већи. То је последица Ојлерове теореме, као и чињеница да је могуће било које  $x_i$  израчунати директно из те теореме<sup>[4]</sup>.

### 2.4. Инверзни конгруентни генератор (ИКГ)

ИКГ је врста нелинеарног конгруентног генератора псеудо-случајних бројева, који користи модуларни мултипликативни инверз (ако он постоји) како би генерисао следећи број у низу. Стандардна једначина за инверзни конгруентни генератор је:

$$X_{i+1} \equiv (a * X_{i-1} + c) \pmod{m} \quad (7)$$

Где је  $c$  целобројна константа. Није тешко видети да  $X_i$  узима вредности од 0 до  $m$ .

### 2.5. Метода средина квадрата (МСК)

Није лоше поменути и овај примитивни метод први пут уведен 1949. године. Увео га је Џон фон Нојман. Први пут је имплементиран на ENIAC-у. За низ  $n$ -тоцифрених псеудо-случајних бројева узећемо  $n$ -тоцифрено семе, квадрирати га, узети средњих  $n$  цифара, и то нам је следећи члан низа и уједно ново семе.

Практично МСК метод није применљив због јако кратког периода. Такође, низ уме да постане нула низ почев од неког члана, или неки други константан низ. За  $n=4$ , ово се дешава са семенима 0100, 2500, 3792, и 7600. За  $n=2$  је то најочигледније, јер ни за једно семе од 00 до 99 није могуће генерисати више од 14 чланова низа, а да низ не постане константан, и то 0, 10, 60 низ, или наизменично 24-57 низ.

### 2.6. Генератор линеарно померајућег регистра (ГЛПР)

ГЛПР је специфичан по томе што се псеудо-случајни бројеви се не добијају аритметичким већ логичким операцијама. Најчешћа операција у употреби је XOR, „ексклузивно или”. Због ових особина ГЛПР је често имплементиран и хардверски. Почетна вредност ГЛПР-а се зове семе. Битови који утичу на следеће стање, тј. битови од којих се добија следећи члан низа, зову се славине. Бит најмање тежине се зове спољна славина. Да би низ имао максималан период од  $m=2^n-1$ , под претпоставком да семе није нула, мора постојати паран број славина.

Поступак се изводи тако што се на славине редом, од бита најмање тежине ка биту највеће тежине, примењује логичка операција, и добијени бит се дописује на почетак броја,

док се последњи бит брише. Да би се следећи члан низа израчунао, може се користити и карактеристични полином облика:

$$X^{p_1} + X^{p_2} + \dots + X^{p_k} + 1 \pmod{2} \quad (8)$$

где је  $k$  паран број, и бројеви  $p_1, p_2, \dots, p_k$  су узајамно прости, тј. највећи број који дели све њих је 1.

Дакле, генератор се може записати у облику

$$X_{i+1} = X_i^{p_1} + X_i^{p_2} + \dots + X_i^{p_k} + 1 \pmod{2} \quad (9)$$

са истим претпоставкама за  $k$  и  $p_1, p_2, \dots, p_k$ .

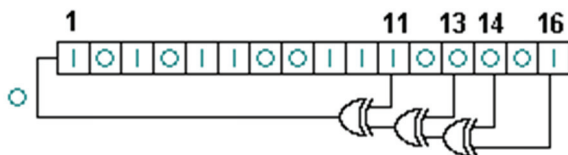
Неки од најчешћих карактеристичних полинома ГЛПР-а су:

$x^8+x^6+x^5+x^4+1$ , који шифтује осмобитне неозначене бројеве и периода је  $255=2^8-1$ .

$x^{16}+x^{14}+x^{13}+x^{11}+1$ , који шифтује шеснаестобитне неозначене бројеве и периода је  $65535=2^{16}-1$ .

Ако погледамо ГЛПР, видећемо да добрим одабиром славина могуће је добити довољно униформно распоређен псеудо-случајан низ. Познато је и да се овај генератор својевремено користио у војне сврхе за криптографију. За то је заслужна веома лака имплементација чак и на електромеханичким уређајима. Мада како је овај генератор један линеаран систем, данас га је могуће релативно лако разбити коришћењем криптоанализе. Јако је погодан за хардверску имплементацију псеудо-случајности, те као хардверски извор, битна особина овог генератора се издваја брзина.

**Пример 7.** Нека је карактеристични полином  $x^{16}+x^{14}+x^{13}+x^{11}+1$ , и логичка функција XOR. Коришћењем ГЛПР-а демонстрирати образовање псеудо-случајног низа за хексадекадно семе ACE1.



Славине су:  
C16, C14, C13, C11.

ACE1 је бинарно:

1010 1100 1110 0001

C16 XOR C14 = 1 XOR 0 = 1

(C16 XOR C14) XOR C13 = 1 XOR 0 = 0

((C16 XOR C14) XOR C13) XOR C11 = 0 XOR 1 = 0

Дописујемо нулу на почетак броја:

0 1010 1100 1110 0001

Бришемо излазну славину (крајњу десну јединицу):

0101 0110 0111 0000

Добили смо нови члан низа, хексадекадно

5670.

Ако наставимо да рачунамо, добићемо елементе:

AB38 559C 2ACE 1567 8AB3 4559 22AC 9156 C8Ab E455 722A 3915 1C8A 8E45 4722 A391  
D1C8 E8E4 7472 BA39 DD1C 6E8E 3747 ...

### 3. Закључак

На основу приказаних података, читалац има увид у важност псеудо-случајности у теорији вероватноћа. Такође, поред битне класификације која је овде дата, не смеју се занемарити ни начини одабира параметара генератора. Овде приказани примери неких од најефикаснијих вредности параметара кроз различите генераторе добијени су емпиријским путем, и читалац се охрабрује да пронађе још неке подједнако или боље ефикасне вредности параметара. Као статистичка провера исправности, може се користити графикон расподељености, као што је представљено у овом раду, као и неки међународно етаблирани статистички пакети тестова, као нпр. DIEHARD, NIST, и други. Овакво тестирање је битно, јер само из добре униформности можемо моделовати добре остале расподеле вероватноћа. Треба још напоменути да иако су прави случајни бројеви бољи, псеудо-случајни бројеви су много бржи и много јефтинији, из тога што на генерисање правих случајних бројева не можемо да утичемо. Важност оваквих симулација вероватноћа је битна и из разлога што нам даје потпору за теоријска рачунања, и увек нам јако брзо може дати до знања да ли се налазимо на правом путу у истраживању или не.

### Референце

- [1] **K. Ćavar, D. Grdenić, I. Kabić, M. Milas, D. Nagradić, D. Zurovec.** Slučajni procesi u igrama na sreću, ([PDF](#)), 2011, p.7-8
- [2] **V. Jevremović.** Verovatnoća i statistika, *Matematički fakultet, Beograd*, 2009, p.96-122
- [3] **G. Marsaglia.** Random numbers fall mainly in the planes, *PNAS, Stanford*, 1968, p.25–28
- [4] **Ž. Mijailović.** Algebra, 1. deo, *MILGOR, Zagreb*, 1993, p.10
- [5] **П. Младеновић.** Вероватноћа и статистика (четврто издање), *Математички факултет, Београд*, 2008, с.25-39
- [6] **L. Trevisan.** Pseudorandomness and Combinatorial Constructions, ([PDF](#)), 2008, p.1
- [7] **L. Trevisan.** Dense Subsets of Pseudorandom Sets, ([PDF](#)), 2008, p.1
- [8] **D. Živkovic.** Osnove dizajna i analize i algoritama, *CET Computer Equipment and Trade*, 2007, p.357-366



# NEKI ELEMENTI MATEMATIČKE ANALIZE SA PRIMENAMA REALIZOVANI U PROGRAMSKOM PAKETU GEOGEBRA

**dr Branko Malešević**

*docent na Katedri za Primenjenu matematiku  
Elektrotehničkog fakulteta Univerziteta u Beogradu  
Bulevar kralja Aleksandra 73, 11000 Beograd, Srbija,  
e-mail: [malesevic@etf.rs](mailto:malesevic@etf.rs)*

**Ivana Jovović, mr Tamara Koledin**

*asistenti na Katedri za Primenjenu matematiku  
Elektrotehničkog fakulteta Univerziteta u Beogradu  
Bulevar kralja Aleksandra 73, 11000 Beograd, Srbija,  
e-mail: [ivana@etf.rs](mailto:ivana@etf.rs), [tamara@etf.rs](mailto:tamara@etf.rs)*

**Matija Brašnjević, Vladica Vasiljević,  
Aleksandar Đajić-Grujić**

*studenti master studija Računarske tehnike i informatike  
Elektrotehničkog fakulteta Univerziteta u Beogradu  
Bulevar kralja Aleksandra 73, 11000 Beograd, Srbija  
e-mail: [matketf@yahoo.com](mailto:matketf@yahoo.com), [vvasiljevic87@gmail.com](mailto:vvasiljevic87@gmail.com),  
[grujic.djajic@gmail.com](mailto:grujic.djajic@gmail.com)*

**Marija Nenezić**

*student master studija Primenjene Matematike  
Elektrotehničkog fakulteta Univerziteta u Beogradu  
Bulevar kralja Aleksandra 73, 11000 Beograd, Srbija, e-  
mail: [majche@rambler.ru](mailto:majche@rambler.ru)*

**Rezime rada.** Korišćenjem alata *GeoGebra* realizovan je niz interaktivnih veb stranica za prikaz rešavanja nekih problema i primera iz graničnih vrednosti funkcija, diferencijalnog računa, simboličke i numeričke integracije, kao i primena na obradu signala u elektrotehnici. Neki od ovih apleta se oslanjaju na Computer Algebra System (CAS) koji *GeoGebra* nudi. U delu koji se bavi graničnim vrednostima funkcija opisani su apleti za ispitivanje neprekidnosti funkcija, određivanje limesa količnika dva polinoma, Lopitalovo pravilo, određivanje Tejlorovih razvoja funkcija kao i vizuelno upoređivanje aproksimacije funkcija u okolini date tačke Tejlorovim polinomima različitih stepena. Dat je aplet za detaljno ispitivanje toka funkcije kroz „deset tačaka“, koji obuhvata određivanje domena, nula i znaka, asimptota, lokalnih ekstremuma i prevojnih tačaka, zatim ispitivanje monotonije i konveksnosti odnosno konkavnosti funkcije, kao i crtanje grafika funkcije. U delu koji se bavi integracijom prikazani su simbolička integracija određenog i neodređenog integrala, numerička integracija, Rimanove sume kao i primena integracije za nalaženje dužine luka krive, površine između dve krive i izračunavanje zapremine i površine omotača rotacionog tela. U okviru primena u elektrotehnici formiran je aplet za ispitivanje da li određeni filter propušta neku frekvenciju ili ne.

**Ključne reči:** *GeoGebra*, granične vrednosti funkcija, tok funkcije, simbolička i numerička integracija, filter

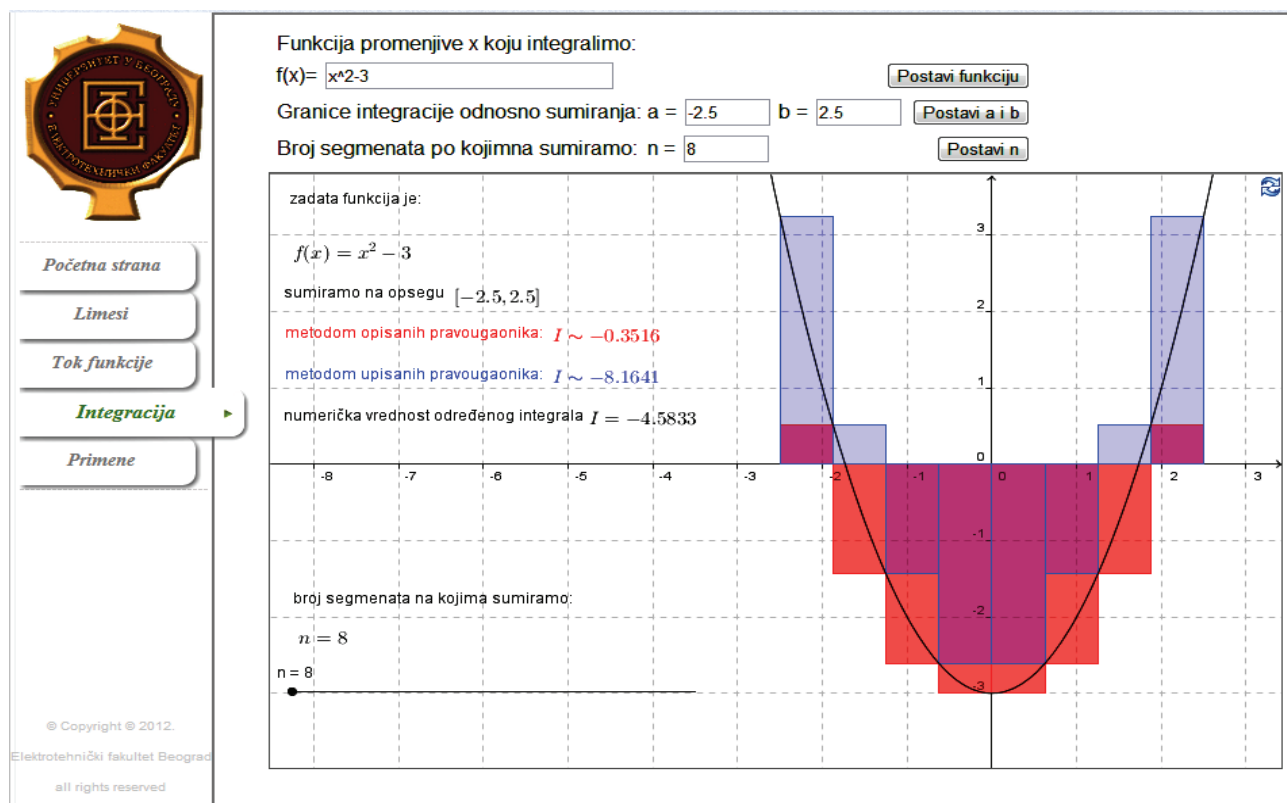
## 1. Uvod

*GeoGebra* je interaktivan i dinamički programski alat, Dynamic Mathematics Software (DMS), za ilustrovanje i rešavanje problema geometrije, algebre i analize kome je primarna orijentacija ka primenama u obrazovanju, [2]. U ovom alatu možemo da pravimo konstrukcije sa tačkama, dužima, vektorima, pravama, konusnim preseccima, kao i sa funkcijama, a zatim da ih dinamički menjamo. Jednačine funkcija i krivih, kao i koordinate tačaka možemo unositi i direktno. Na taj način je programski alat *GeoGebra* u mogućnosti da radi sa promenljivim koje predstavljaju brojeve, vektore i tačke, da traži granične vrednosti, izvode i integrale funkcija, kao i da izvršava neke predefinisane naredbe. Program je urađen u Javi i najvećim delom je „open source“. Trenutna stabilna verzija *GeoGebra* je 4.0, a u izradi je beta verzija 4.2, kao i verzija 5.0. Verzija 4.2 u kojoj je rađena ova veb stranica ima ugrađen računarski algebarski sistem Computer algebra system (CAS) za simbolička izračunavanja. Ovaj alat je za korisnika „ispod haube“ i nije dat na uvid.

Programski paket *GeoGebra* je više okrenut ka rešavanju konkretnih matematičkih problema, a znatno manje pažnje posvećuje prikazivanju dobijenih rezultata, njihovoj upadljivoj prezentaciji i lakoći prenosivosti na druge računare. Naša ideja je da u okviru ovog rada formiramo jednu veb stranicu u koju ćemo integrisati skup problema iz osnova analize. Tokom izrade koristili smo se tehnologijama razvoja veb prezentacija kao što su HTML (Hyper Text Markup Language) i CSS

(Cascading Style Sheets) koji su nam omogućili da prezentacija bude kompaktna, optimalna i veoma jednostavna za snalaženje i svako naredno unapređivanje, [1]. Takođe smo se koristili i JavaScript-om koji smo integrisali u našu prezentaciju kako bismo menijima sa leve i desne strane dali vizuelno lep dojam i kako bismo ostvarili komunikaciju između korisnika, prezentacije i integrisanih matematičkih problema. Pomoć u izradi prezentacije, pored ostalih, pronašli smo i u programskim paketima Aptana, Notepad++, NetBeans i Eclipse, koji su nam omogućili da na brz i efikasan način prilagođavamo izgled stranice našim potrebama i zahtevima, a sa druge strane su nam omogućili jednostavno ubacivanje matematičkih problema koje smo prethodno izvezli iz programskog paketa *GeoGebra*.

Alati Paint.Net i Photopost su nam omogućili da uradimo sve grafičke elemente koji se pojavljuju na našoj prezentaciji, kao što je pozadina, zajednička slika na početnoj strani i slike na drugim stranama, menijima i okvirima, zapravo sve ono što je našu prezentaciju učinilo vizuelno upadljivom. Prezentacija koju smo realizovali može veoma lako da zauzme mesto kod nastavnika, profesora i predavača, kao i da bude jedno od pomagala da približi materiju đacima, studentima i drugim slušaocima. Pored pokaznog tipa prezentacija može da ima mnogobrojne primene i u samostalnoj proveru znanja pređenog gradiva, pošto su u realizaciji prezentacije korišćene tehnologije koje omogućavaju da stranicu postavimo na internet. Posebnu pažnju smo posvetili interakciji između korisnika i prezentacije tako da u realnom vremenu izvršavamo promenu parametara matematičkog problema i gotovo trenutno dobijamo rešenje.



Slika 1. Izgled veb prezentacije

## 2. Apleti za granične vrednosti funkcija

U okviru veb stranice realizovano je pet apleta koji se bave graničnim vrednostima funkcija.

### 2.1. Granična vrednost date funkcije u zadatoj tački

Neka je realna funkcija  $f$  definisana na skupu  $D \subseteq \mathbb{R}$  i neka je  $a$  tačka nagomilavanja skupa  $D$  takva da se u svakoj njenoj okolini nalazi neki element  $x \in D$  sa leve (desne) strane. Ako važi

$$\begin{aligned} & (\forall \varepsilon > 0)(\exists \delta > 0)(\forall x \in D) a - \delta < x < a \Rightarrow |f(x) - A| < \varepsilon \\ & ((\forall \varepsilon > 0)(\exists \delta > 0)(\forall x \in D) a < x < a + \delta \Rightarrow |f(x) - A| < \varepsilon), \end{aligned}$$

onda kažemo da je realan broj  $A$  leva (desna) granična vrednost funkcije  $f$  kada  $x \rightarrow a$  I

pišemo  $\lim_{x \rightarrow a_-} f(x) = A$  ( $\lim_{x \rightarrow a_+} f(x) = A$ ).

Nije teško videti da granična vrednost funkcije  $f$  kad  $x \rightarrow a$  postoji ako i samo ako postoje leva i desna granična vrednost i ako su te granične vrednosti međusobno jednake, videti [4].

*Realizacija u GeoGebri*

Leva (desna) granična vrednost funkcije  $f$  kad  $x \rightarrow a$  se određuje pomoću komande

$$\text{LimitBelow}[f, a] \quad (\text{LimitAbove}[f, a]),$$

a pomoću formirane logičke promenljive ispituje se njihova jednakost. Na osnovu dobijenih rezultata na ekranu će biti ispisane leva i desna granična vrednost funkcije  $f$  kad  $x \rightarrow a$ . Ukoliko su jednake biće ispisana i granična vrednost funkcije, a ukoliko nisu biće ispisano: „Granična vrednost kada  $x \rightarrow a$  funkcije  $f$  ne postoji”.

### 2.2. Granična vrednost količnika polinoma

Neka su  $P(x) = a_n x^n + a_{n-1} x^{n-1} + \dots + a_1 x + a_0$  i  $Q(x) = b_m x^m + b_{m-1} x^{m-1} + \dots + b_1 x + b_0$  dva polinoma,

gde su  $a_0, a_1, \dots, a_n, b_0, b_1, \dots, b_m \in \mathbb{R}$ ,  $a_n \neq 0$  i  $b_m \neq 0$ . Tada je

$$\lim_{x \rightarrow \infty} \frac{P(x)}{Q(x)} = \begin{cases} \text{sgn}(a_n / b_m) \infty, & n > m; \\ a_n / b_m, & n = m; \\ 0, & n < m. \end{cases}$$

*Realizacija u GeoGebri*

Pomoću komande  $\text{Stepen}[P]$  ( $\text{Degree}[P]$ ) određuju se stepeni unetih polinoma, a pomoću komande  $\text{Limes}[P/Q, \infty]$  ( $\text{Limit}[P/Q, \infty]$ ) određuje se granična vrednost količnika polinoma kada  $x \rightarrow \infty$ . Pored toga komandama  $\text{Deli}[P, Q]$  ( $\text{Div}[P, Q]$ ) i  $\text{Ostatak}[P, Q]$  ( $\text{Mod}[P, Q]$ ) nalazimo količnik i ostatak pri deljenju polinoma  $P$  i  $Q$ , što vrlo lepo ilustruje prethodno navedenu teoremu.

### 2.3. Lopitalovo pravilo

Neka su funkcije  $f$  i  $g$  definisane i diferencijabilne u nekoj okolini tačke  $a$  (osim, možda u samoj tački  $a$ ), gde je  $a \in \mathbb{R}$  ili je  $a = \pm\infty$ . Neka je  $\lim_{x \rightarrow a} f(x) = \lim_{x \rightarrow a} g(x) = 0$  ili  $\lim_{x \rightarrow a} f(x) = \lim_{x \rightarrow a} g(x) = \pm\infty$

i neka je  $g'(x) \neq 0$  u nekoj okolini tačke  $a$ . Tada je  $\lim_{x \rightarrow a} \frac{f(x)}{g(x)} = \lim_{x \rightarrow a} \frac{f'(x)}{g'(x)}$ , ako postoji (konačna ili beskonačna) granična vrednost sa desne strane, videti [4].

#### Realizacija u GeoGebri

Pomoću komande  $Limes[f, a]$  ( $Limit[f, a]$ ) određujemo limese unetih funkcija kada  $x \rightarrow a$ , a zatim ispitujemo da li su dati limesi jednaki i da li su jednaki 0 ili  $\pm\infty$ . Ako jedan od ovih uslova nije ispunjen ispisuje se poruka: „Za date funkcije ne možemo da primenimo Lopitalovo pravilo“.Ukoliko su ispunjeni odgovarajućii uslovi, što proveravamo pomoću logičke promenljive, određuju se izvodi polaznih funkcija komandom  $f'$  ( $Izvod[f]$ ,  $Derivative[f]$ ) i na kraju se ispituje da li je limes količnika funkcija jednak limesu količnika izvoda kada  $x \rightarrow a$ . Ukoliko je ispunjen i taj uslov na ekranu se detaljno ispisuje limes količnika funkcija, limes količnika izvoda i vrednost odgovarajućii limesa kada  $x \rightarrow a$ , što zahteva elementarno poznavanje tekstualnog editora TeX.

#### Lopitalovo pravilo

Prva funkcija  $f(x) = (1 + x)^{1/2} - 1$   $\lim_{x \rightarrow 0} f(x) = 0$

Prvi izvod funkcije  $f(x)$  je  $f'(x) = 1 / (2(x + 1)^{1/2})$ .

Druga funkcija  $g(x) = x$   $\lim_{x \rightarrow 0} g(x) = 0$

Prvi izvod funkcije  $g(x)$  je  $g'(x) = 1$ .

Tačka u kojoj tražimo limes  $v = 0$

$$\lim_{x \rightarrow 0} \frac{f(x)}{g(x)} = \lim_{x \rightarrow 0} \frac{(1 + x)^{1/2} - 1}{x} = \lim_{x \rightarrow 0} \frac{1}{2(x+1)^{1/2}} = 0.5.$$

Slika 2. Lopitalovo pravilo

### 2.4. Tejlorov polinom i aproksimacije

Ako funkcija  $f$  ima u tački  $a$  konačne izvode do reda  $n$ , tada se polinom

$$T_n(x) = f(a) + f'(a)(x-a) + \frac{f''(a)}{2!}(x-a)^2 + \dots + \frac{f^{(n)}(a)}{n!}(x-a)^n$$

naziva Tejlorovim polinomom stepena  $n$  funkcije  $f$  u tački  $a$ , videti [4].



## Realizacija u GeoGebri

Najpre se unosi funkcija  $f$ , a potom tačke  $a$  u kojoj želimo da aproksimiramo unetu funkciju. Nakon toga unose se dva prirodna broja,  $n$  i  $m$ , koji predstavljaju stepene željenih Tejlorovih polinoma. Zatim se komandom *TejlorovPolinom*[ $f, a, n$ ] (*TaylorPolynomial* [ $f, a, n$ ]) formiraju dva polinoma i ispisuju na ekranu. Ovaj aplet se može koristiti i za nalaženje tangente na unetu funkciju u zadatoj tački, ukoliko se za jedan od Tejlorovih polinoma izabere stepen jedan. Takođe je pogodan i za vizuelno prikazivanje činjenice da polinomske funkcije većeg stepena bolje aproksimiraju datu funkciju u okolini zadate tačke, budući da aplet grafički prikazuje unetu funkciju i obe polinomske funkcije formirane na osnovu dobijenih Tejlorovih polinoma.

### 3. Apleti za ispitivanje toka i crtanje grafika funkcije

U ovoj glavi opisujemo aplete koji se bave tokom funkcije. Data problematika je razmatrana i u [6].

#### 3.1. Oblast definisanosti (domen) funkcije

Oblast definisanosti funkcije  $f : D \rightarrow \mathbb{R}$  može biti bilo koji neprazan podskup od  $\mathbb{R}$ : interval  $(a, b)$ , segment  $[a, b]$ , polusegmenti  $[a, b)$  i  $(a, b]$ , unije bilo kojih od ovih skupova, uključujući, eventualno, i izolovane tačke. U posebnom slučaju, kada je funkcija  $f$  zadata analitički od interesa je naći njenu prirodnu oblast definisanosti. To je skup svih realnih brojeva  $x \in \mathbb{R}$  za koje takva formula ima smisla. U ovom radu smo se bavili analitičkim izrazima sledećih tipova:  $\frac{f(x)}{g(x)}$ ,  $\ln(f(x))$ ,  $\sqrt{f(x)}$ ,  $\sqrt[3]{f(x)}$ ,  $e^x$ , kao i funkcijama koje se dobijaju njihovim sabiranjem, oduzimanjem, množenjem i kompozicijom.

## Realizacija u GeoGebri

Programski alat *GeoGebra* ne poseduje mogućnost da za zadatu funkciju odredi njen domen, te je stoga ovaj deo ispitivanja toka funkcije možda i najzanimljiviji. Naime, prvo se poziva JavaScript funkcija koja parsira zadatu funkciju znak po znak. Kada se u funkciji naiđe na deljenje, koren, ili logaritam, svaka od tih podfunkcija se izdvaja u posebnu klasu manjih funkcija. Na kraju parsiranja, imamo više klasa manjih funkcija koje kao nejednakosti zadajemo *GeoGebri* sa logičkim operatorom konjunkcije, kako bi domen bio određen za celu funkciju (klasa „deljenja” za uslov postavlja da je imenilac različit od 0, klasa „ln” zahteva da argument bude veći od 0, klasa „koren” ima za uslov da je funkcija pod korenom veća ili jednaka od 0).

#### 3.2. Nule i znak

Skup svih vrednosti  $x$  iz oblasti definisanosti funkcije  $f$  za koje je  $f(x) = 0$  su nule funkcije. Na grafiku funkcije nule su tačke na  $x$ -osi u kojima grafik funkcije seče ili dodiruje  $x$ -osu. Znak funkcije određuje se tako što se za nejednakosti  $f(x) > 0$  i  $f(x) < 0$  odrede skupovi argumenta  $x$  za koje su date nejednakosti zadovoljene.

## Realizacija u GeoGebri

Nule funkcije  $f$  se računaju komandom  $B = \{Intersection[0*x, f(x), minX, maxX]\}$ . Ova komanda nalazi svaki presek funkcije  $f$  sa  $x$ -osom i nađene tačke vraća u listu B. Korišćena je lista tačaka zato što unapred nije poznat broj nula funkcije, a potrebno je sve tačke podesiti na debljinu

5. Takođe, na grafiku se ne vidi funkcija  $f$ , što je postignuto isključivanjem opcije *Prikaži objekat* (*Show object*) za datu funkciju. Za računanje znaka funkcije kreće se od minimalne vrednosti za koju je funkcija definisana i ide do maksimalne, ukoliko je funkcija u datoj trenutnoj tački pozitivna, ispisuje se znak “+”, ukoliko je negativna “-“. Ovo se postiže komandom

*Sequence[If[f(k) > 0, Text["+", (k, 0.3)], If[f(k) < 0, Text["-", (k, 0.3)]]], k, minX, maxX, 0.4].*

### 3.3. Asimptote

- Prava  $y = n$  je desna (leva) horizontalna asimptota funkcije  $f$  ako važi

$$\lim_{x \rightarrow \infty} f(x) = n \quad \left( \lim_{x \rightarrow -\infty} f(x) = n \right).$$

- Prava  $x = x_0$  je vertikalna asimptota funkcije  $f$  ako važi

$$\lim_{x \rightarrow x_{0+}} f(x) = \pm\infty \quad \text{ili} \quad \lim_{x \rightarrow x_{0-}} f(x) = \pm\infty.$$

- Prava  $y = kx + n$  je desna (leva) kosa asimptota funkcije  $f$  ako važi

$$\lim_{x \rightarrow \pm\infty} (f(x) - (kx + n)) = 0.$$

Parametre  $k$  i  $n$  određujemo sa  $k = \lim_{x \rightarrow \pm\infty} \frac{f(x)}{x}$  i  $n = \lim_{x \rightarrow \pm\infty} (f(x) - kx)$ .

*Realizacija u GeoGebri*

U programskom alatu *GeoGebra* asimptote se određuju komandom  $B = \{Asimptota[f]\}$ .

### 3.4. Izvod funkcije, lokalni ekstremumi i monotonost funkcije

Neka je funkcija  $f$  definisana u nekoj okolini tačke  $a$ . Ako postoji konačna ili beskonačna granična vrednost  $f'(a) = \lim_{x \rightarrow a} \frac{f(x) - f(a)}{x - a}$ , onda kažemo da je  $f'(a)$  prvi izvod funkcije  $f$  u tački  $a$ . Funkcija  $f$  na skupu  $E$  ima u tački  $a \in E$  lokalni maksimum (lokalni minimum) jednak  $f(a)$  ako postoji okolina tačke  $a$  takva da za svako  $x$  u datoj okolini važi nejednakost  $f(x) \leq f(a)$  ( $f(x) \geq f(a)$ ). Ako funkcija  $f$  ima izvod u svakoj tački  $x \in (a, b)$  i ako je  $f'(x) > 0$  ( $f'(x) < 0$ ) za svako  $x \in (a, b)$ , onda je funkcija monotono rastuća (monotono opadajuća) na intervalu  $(a, b)$ .

*Realizacija u GeoGebri*

Prvi izvod funkcije  $f$  računamo pomoću jedne od komandi *Izvod[f]*, *Derivative[f]* ili  $f'$ . Za izračunavanje lokalnih ekstremuma funkcije prvo smo računali prvi izvod funkcije  $f$  pomoću komande  $g = f'$ . Zatim smo računali i drugi izvod  $h$  koji nam je potreban za ispitivanje da li se radi o lokalnom maksimumu ili lokalnom minimumu. Funkciju, prvi i drugi izvod sakrivamo sa grafika, dok ostavljamo tačke lokalnih ekstremuma. Najzad, ekstremume računamo kao dve liste, pošto maksimume želimo da prikazemo plavom, a minimume crvenom bojom

*Sequence[If[h(x(Element[B, k])) < 0, (x(Element[B, k]), f(x(Element[B, k])))], k, 1, n],*

*Sequence[If[h(x(Element[B, k])) > 0, (x(Element[B, k]), f(x(Element[B, k])))], k, 1, n].*

Za određivanje intervala monotonosti funkcije, slično kao kod znaka funkcije, prolazimo kroz celu funkciju (kroz tačke za koje je definisana od minimalne do maksimalne), i za sve vrednosti argumenta  $x$  za koje je  $f'(x) < 0$ , crtamo vektor od tačke  $(x, 0)$  do tačke  $(x, -1)$ , a za sve vrednosti argumenta  $x$  za koje je  $f'(x) > 0$ , crtamo vektor od tačke  $(x, 0)$  do tačke  $(x, 1)$ . Ovo se postiže komandom

*Sequence[If[g(k) > 0, Vector[(k, 0), (k, 1)], Vector[If[g(k) < 0, Vector[(k, 0), (k, -1)]]]], k, -50, 50, 0.2].*

### 3.4. Prevojne tačke, konveksnost i konkavnost funkcije

Neka je funkcija  $f$  dva puta diferencijabilna na intervalu  $(a, b)$ . Tada je funkcija konveksna na intervalu  $(a, b)$  ako i samo ako je  $f''(x) \geq 0$  za svako  $x \in (a, b)$ , a konkavna na datom intervalu ako i samo ako je  $f''(x) \leq 0$  za svako  $x \in (a, b)$ . Tačka  $(x_0, f(x_0))$  je prevojna tačka funkcije  $f$ , ako je funkcija  $f$  konkavna na nekom intervalu  $(c, x_0) \subseteq (a, b)$  i konveksna na nekom intervalu  $(x_0, d) \subseteq (a, b)$ , ili obrnuto, videti [4].

#### Realizacija u GeoGebri

Za ispitivanje konveksnosti funkcije prvo smo, slično kao i za monotonost, odredili vrednost drugog izvoda  $h$  funkcije  $f$  u svim tačkama u kojima je definisan, i u zavisnosti od toga da li je manji ili veći od nule zaključili da se radi o konveksnosti, odnosno konkavnosti. Ovo se postiže komandom

*Sequence[If[h(k) < 0, Vector[(k, 0), (k, 1)], Vector[If[h(k) > 0, Vector[(k, 0), (k, -1)]]]], k, -50, 50, 0.2].*

Za izračunavanje prevojnih tačaka određujemo tačke u kojima drugi izvod seče  $x$ -osu, pomoću komande  $B = \{Intersection[0 * x, f''(x), -50, 50]\}$ . Zatim prolazimo kroz celu listu dobijenih presečnih tačaka, i za svaku određujemo vrednost funkcije  $f$  u datoj tački, koristeći komande  $n = Length[B]$ , *Sequence[(x(Element[B,k]), f(x(Element[B,k]))), k, 1, n, 1].*

Debljinu tačaka postavljamo na 5, a skrivamo sve ostale objekte sa grafika.

## 4. Apleti za simboličku i numeričku integraciju

Korišćenjem alata *GeoGebra* realizovan je niz interaktivnih veb stranica za prikaz i rešavanje nekih problema i primera iz simboličke integracije. U primerima su prikazani simbolička integracija određenog i neodređenog integrala, Rimanove sume, primena integracije u nalaženju dužine luka krive zadate u eksplicitnom ili parametarskom obliku, zatim primena integracije u nalaženju površine između dve krive kao i za izračunavanje zapremine i površine omotača rotacionog tela. Stranice su bazirane na veb apletima izvezenim iz *GeoGebra* koji su povezani sa HTML kodom stranice, koji generiše interfejs i prateći tekst, preko Javascript komandi. Iskorišćeno je više različitih opcija *GeoGebra* za prikaz grafika funkcija i crtanje krivih. U apletima smo imali potrebu da koristimo i prikazemo eksplicitno, implicitno kao i parametarski zadate funkcije. U primeru računanja zapremine i površine omotača rotacionog tela bilo je potrebno u 2D grafičkom prikazu *GeoGebra* kreirati odgovarajuću ilustraciju. Baze rotacionog tela su nacrtane preko implicitno zadatih elipsi dok je omotač tela nacrtan refleksijom segmenta osnovne funkcije oko  $x$ -ose.

### 4.1. Simbolička integracija

Prvi primer iz integracije je simbolička integracija određenog i neodređenog integrala realne funkcije jedne promenljive. Rezultat rada apleta je simbolički izračunat neodređen integral i simbolički izračunat određen integral prema Njutn-Lajbnicovoj formuli. U apletu su prikazani i grafici funkcije  $f$  i jedne njene primitivne funkcije  $F$ , i označena je površina koju predstavlja određeni integral. Korisnik unosi funkciju koju integralimo  $f$  i granice određenog integrala  $a$  i  $b$ .

## 4.2. Rimanove sume

Ovde prikazujemo gornju i donju Rimanovu sumu, kao i numeričku vrednost određenog integrala kome ta suma odgovara. Korisnik unosi funkciju  $f$ , granice  $a$  i  $b$ , i broj segmenata sumiranja  $n$ . Broj segmenata je takođe moguće kontrolisati klizačem na grafičkoj radnoj površini, koji je napravljen pomoću alatke *Slider*. Za izračunavanje suma koristili smo ugrađene funkcije *GeoGebra UpperSum[ $f$ ]* i *LowerSum[ $f$ ]* koje u potpunosti odgovoraju gornjoj i donjoj Rimanovoj sumi. Pri realizaciji apleta za Rimanove sume došli smo do nekih interesantnih zapažanja. Gornja i donja Rimanova suma izuzetno sporo konvergiraju ka tačnoj numeričkoj vrednosti određenog integrala. Za tačnost na tri decimale je bilo potrebno sumirati skoro 10,000 elemenata, što usporava aplet do neupotrebljivosti. Ovo lepo ilustruje nepraktičnost rešavanja problema numeričke integracije grubom silom i potrebu za efikasnijim rešenjima poput Simpsonovog pravila ili trapeznih suma koji dolaze do zadovoljavajuće tačnosti sa desetak iteracija.

## 4.3. Dužina luka eksplicitno i parametarski zadate krive

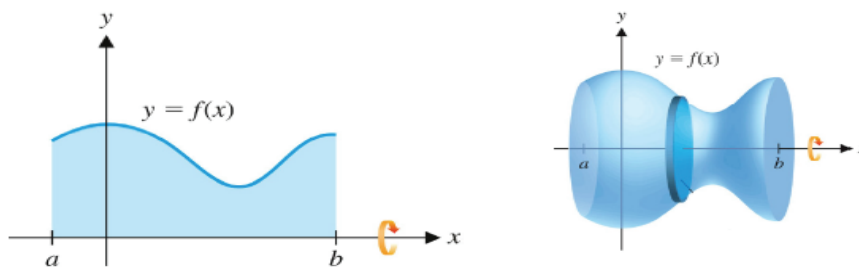
Nalaženje dužine luka krive smo realizovali kroz dva apleta. Prvi aplet prikazuje primenu određenog integrala za nalaženje dužine luka krive zadate funkcijom u eksplicitnom obliku  $y = f(x)$ . Tu dužinu nalazimo kao vrednost integrala  $L = \int_a^b \sqrt{1 + (f'(x))^2} dx$ . Izvod i integral potrebni u zadatoj formuli su realizovani CAS komandama *Derivative[ $f$ ]* i *Integral[ $f$ ]*. Da bi smo označili segment krive od  $a$  do  $b$  koristili smo *GeoGebra* naredbu *Function[ $f, a, b$ ]*, koja će za eksplicitno datu funkciju nacrtati njen isečak na zatom segmentu. Drugi aplet nalazi dužinu luka parametarski zadate krive na segment, gde parametar  $t$  uzima vrednosti od  $a$  do  $b$ :  $x = x(t)$ ,  $y = y(t)$ ,  $a \leq t \leq b$ . Dužina te krive se nalazi kao vrednost integrala  $L = \int_a^b \sqrt{x^2(t) + y^2(t)} dt$ . Za generisanje i iscrtavanje parametarske krive korišćena je *GeoGebra* funkcija *Curve[ $x, y, t, a, b$ ]*.

## 4.4. Površina figure određene sa dve krive

Nalazimo površinu figure koju kreiraju dve krive. Ova vrednost je određena razlikom određenih integrala gornje i donje funkcije na željenom intervalu  $P = \int_a^b |f(x) - g(x)| dx$ . Korisnik treba da unese obe funkcije u eksplicitnom obliku, kao i zadati interval.

## 4.5. Površina i zapremina omotača rotacionog tela

Ovde prikazujemo primenu određenog integrala za nalaženje površine i zapremine omotača tela generisanog rotacijom krive oko  $x$ -ose. Ako grafik krive  $y = f(x)$ ,  $a \leq x \leq b$  rotiramo oko  $x$ -ose, on opisuje jedno rotaciono telo. Zapremina ovako nastalog tela je  $V = \pi \int_a^b f^2(x) dx$ , a površina omotača je  $S = 2\pi \int_a^b f(x) \sqrt{1 + (f'(x))^2} dx$ . Radi preciznosti i podrške za veći broj klasa funkcija sva simbolička integracija je rađena u CAS-u. Bazu rotacionog tela prikazujemo dvema elipsama.



Slika 3. Rotacija krive oko  $x$ -ose

#### 4.6. Neki komentari o CAS-u u *GeoGebri*

Kako smo radili sa izvodima i integralima funkcija, neki od ovih apleta se oslanjaju na Computer Algebra System (CAS) koji *GeoGebra* nudi. CAS omogućava *GeoGebri* da reši probleme simboličkog diferenciranja i integracije koje inače ne bi mogla. U toku izrade rada *GeoGebra* je bila u fazi konstantnog razvoja i bilo je potrebno raditi sa “daily build“ beta verzijama. Na početku izrade prezentacije *GeoGebra* je imala dve mogućnosti za CAS. Za lokalnu upotrebu je tu bila Maxima (<http://maxima.sourceforge.net/>), dok je zbog prevelike veličine ovog paketa za aplete korišćen MathPiper (<http://www.mathpiper.org/>). Ovo je dovelo do situacija da se mnoge kompleksnije mogućnosti CAS nisu mogle koristiti u *GeoGebri* veb apletima. Rešenje je došlo sa vremenom kako je razvojni tim *GeoGebra* u novim verzijama prešao na jedinstveni CAS za sve primene MPReduce (<http://dev.geogebra.org/trac/wiki/MPReduce>) baziran na sistemu REDUCE (<http://reduce-algebra.sourceforge.net/>). Interesantno je primetiti da osnovni “final build“ *GeoGebra* 4.0 i dalje ne nudi integrisani CAS koji je ostao rezervisan za beta verziju 4.2.

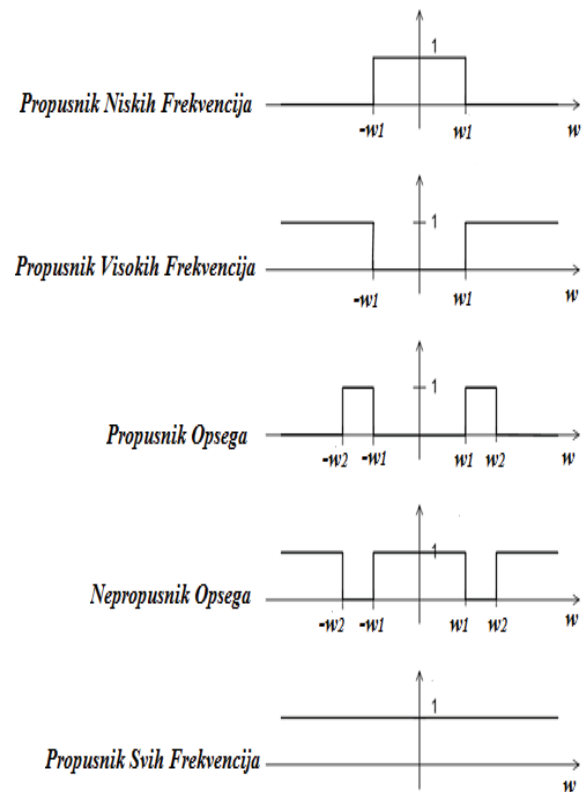
#### 5. Filtri

Dokazano je da je *GeoGebra* odličan alat za primenu u matematici, kako u nastavi tako i u učenju. Međutim, ljubitelji ovog softvera pokazuju da se *GeoGebra* može primeniti i na neke druge oblasti pa je tako i kad je elektrotehnika u pitanju. Obrada signala je oblast elektrotehnike koja se bavi različitim operacijama nad signalima, analizom signala i merenjem promena vremenskih i prostornih karakteristika. Signali sadrže informacije o ponašanju i osobinama neke pojave i u zavisnosti od toga koji deo informacije želimo da dobijemo, primenjujemo odgovarajuće obrade signala. Termin „signal“, između ostalog, obuhvata audio, video, muzički, medicinski, radarski, sonarni, komunikacioni signal, govor ili sliku. Filtriranje je jedan od procesa koji se koristi u obradi signala. Filtar je proces kojim se uklanjaju nepoželjne komponente signala. Funkcija filtra može u potpunosti ili delimično da potisne neke delove signala. To znači da se filtrom mogu ukloniti neke komponente signala dok se ostale propuštaju kroz filtara i na takav način se umanjuju smetnje u signalu i šum. Filtri mogu biti analogni ili digitalni, sa diskretnim ili kontinualnim vremenom, linearni ili nelinearni, vremenski promenljivi ili nepromenljivi, aktivni ili pasivni, sa konačnim (FIR) ili beskonačnim (IIR) impulsnim odzivom. U obradi signala, filtri koji se često koriste su linearni sa kontinualnim vremenom. Ovi filtri su dizajnirani tako da mogu da uklone samo određene frekvencije iz signala, a ostale da propuste. Frekvencijski odziv linearnih filtara se može podeliti u nekoliko različitih grupa, pa postoji više tipova tih filtara. Oni definišu za dati filtara koje frekvencije će da propusti (propusni opseg) i koje će da odbaci (nepropusni opseg):

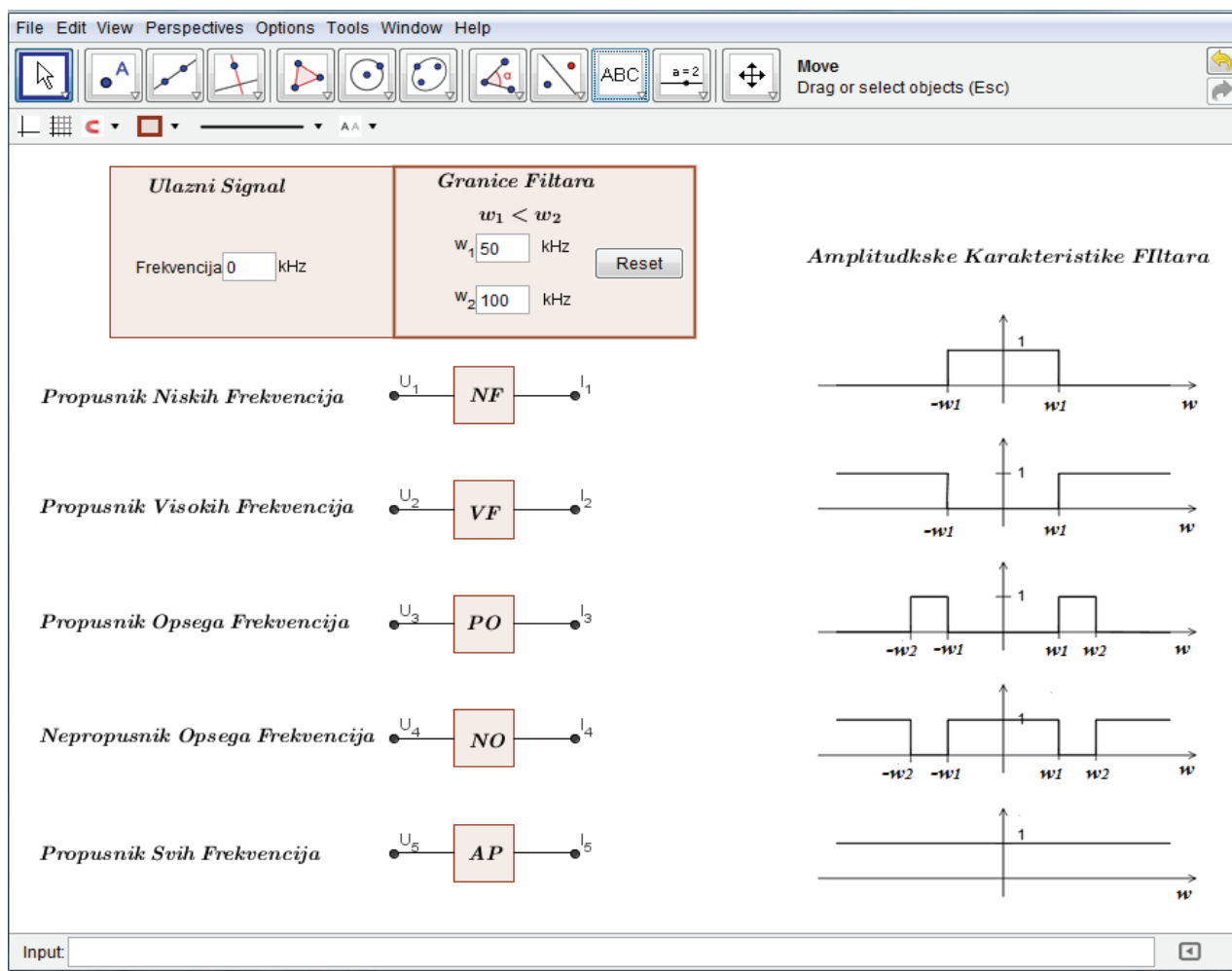
- propusnik niskih frekvencija (NF) – propušta niske, a odbacuje visoke frekvencije u odnosu na određenu graničnu vrednost  $w_1$ ;
- propusnik visokih frekvencija (VF) – propušta visoke, a odbacuje niske frekvencije u odnosu na određenu graničnu vrednost  $w_1$ ;
- propusnik opsega (PO) – propušta samo frekvencije u određenom opsegu ( $w_1, w_2$ );
- nepropusnik opsega (NO) – odbacuje samo frekvencije u određenom opsegu ( $w_1, w_2$ );

- propusnik svih frekvencija (AP – all pass) – propušta sve frekvencije ali se menja faza signala.

U obradi signala, simbol filtra je najčešće kvadrat sa dve linije na sredini koje predstavljaju ulazi i izlazi signala filtra. U *GeoGebri* postoji alat kojim se to može jednostavno konstruisati. Uz pomoć alata *Mnogougao (Polygon Tool)* može se kreirati pet kvadrata za pet tipova filtera, u okviru *Površine za crtanje (Graphics View)* prostora *GeoGebra* interfejsa). Kako je potrebno da linije koje predstavljaju ulazne i izlazne signale postavimo na sredinu kvadrata koristili smo alat *Središte ili Centar (Midpoint or Center Tool)* za pronalaženje tačaka na sredini u odnosu na stranice kvadrata i alat *Duž određena dvema tačkama (Segment between Two Points Tool)* za spajanje tih tačaka u linije. Za dobijanje odgovarajućih simbola filtera sve nepotrebne tačke ćemo postaviti da budu nevidljive. Cilj ovog dela rada je da pokaže da li određeni filter propušta neku frekvenciju ili ne. Za početak je potrebno kreirati tri *Tekstualna polja (Input Box)* koja će biti linkovana na karakteristične frekvencije filtera i uz pomoć kojih će moći da se menja njihova vrednost. Karakteristične frekvencije su  $w_1$  i  $w_2$  koje predstavljaju granice propusnih opsega filtera i kojima su početne vrednosti 50kHz i 100kHz, respektivno. Svaki filter će imati iste granične vrednosti, ali različite propusne opsege u zavisnosti od njegovih amplitudskih karakteristika. Treća karakteristična frekvencija je *Freq* sa početnom vrednošću 0kHz. Ta frekvencija je test frekvencija koja će se puštati kroz filtre i proveravati da li prolazi kroz njih ili ne. Menjanjem vrednosti ovih frekvencija možemo da testiramo različite frekvencije za različite propusne opsege. Potrebno je na neki način da vidimo rezultat rada filtera. Jedan od načina je ispis „*PROPUSTENO!*” na izlazu svakog filtra koji propušta test frekvenciju. Za to se koriste liste i tekstualna polja. Najpre se kreira lista *list1*, koja sadrži samo jedan član „*PROPUSTENO!*”. Potom se kreira pet tekstulanih polja, po jedno za izlaz svakog filtra, uz pomoć alata *Ubaci tekstualno polje (Insert Text Tool)*. Svako ovo polje će ispisivati vrednosti izlaza filtera definisanih sa  $EXIT_i$  ( $1 \leq i \leq 5$ ). Vrednost izlaza filtera dobija se rešavanjem *if* naredbe, definisane za svaki izlaz pojedinačno. Naredba ispituje da li je test frekvencija u odgovarajućem opsegu, ako jeste uzima se vrednost liste *list1*. Na primer, u slučaju Propusnika Niskih Frekvencija, odgovarajuća naredba će biti oblika  $iff[-w_1 \leq Freq \wedge (Freq \leq w_1), list1]$  i ako je uslov ispunjen na izlazu ovog filtra biće ispisano „*PROPUSTENO!*”, u suprotnom se neće ispisati ništa. Analogno ovome definišu se *if* naredbe za ostale filtre. Samim tim, omogućeno je testiranje neke frekvencije za zadate propusne opsege automatski za sve filtre istovremeno. Jedina dva uslova za rad filtera su da je *Freq* različito od 0 i da je granična vrednost  $w_1$  manja od granične vrednosti  $w_2$ . Pogledati referencu [5].



Slika 4. Amplitudske karakteristike linearnih filtera



Slika 5. Apolet, primena GeoGebre na linearne filtre

## Literatura:

- [1] **J. Hill, J. A. Brannen.** Brilljantno HTML5 i CSS3. *CET Computer Equipment and trade*, 2011. Beograd
- [2] **J. Hohenwarter, M. Hohenwarter.** Introduction to GeoGebra, [www.geogebra.org](http://www.geogebra.org), 2011.
- [3] **R. Lawrence, G. Bernard.** Theory and Application of Digital Signal Processing. *Englewood Cliffs, NJ: Prentice-Hall 1975*, New Jersey.
- [4] **M. Merkle.** Matematička analiza, teorija i hiljadu zadataka za student tehnike, II izdanje. *Akadska misao*, 2008. Beograd
- [5] **M. Nenezić, B. Malešević.** Some interactive visualisations in electronics signals using program GeoGebra. *MoNGeometrija*, 2012. Novi Sad pp. 145-163.
- [6] **Đ. Takači, G. Stankov, M. Rakić.** On the role of GeoGebra in examining functions. *International GeoGebra Conference for Southeast Europe*, 2011. Novi Sad pp.18-26.







**УНИВЕРЗИТЕТ У БЕОГРАДУ  
МАТЕМАТИЧКИ ФАКУЛТЕТ**

**ЗБИРКА АПСТРАКАТА  
ТРЕЋЕГ СИМПОЗИЈУМА „МАТЕМАТИКА И ПРИМЕНЕ”  
НАЦИОНАЛНОГ СКУПА СА МЕЂУНАРОДНИМ УЧЕШЋЕМ  
Београд, 25 – 26. мај 2012.**

**ОТВАРАЊЕ СКУПА**

проф. др Миодраг Матељевић, декан Математичког факултета,  
председник САНУ академик Никола Хајдин,  
представници Универзитета у Београду,  
др Александра Дреџун, директорка Центра за промоцију науке

**ОТВАРАЊЕ ДИГИТАЛНОГ ЛЕГАТА „МИЛУТИН МИЛАНКОВИЋ“**

проф. др Надежда Пејовић, секретар Удружења „Милутин Миланковић“,  
др Саша Малков, Универзитет у Београду, Математички факултет

**ПРЕДСТАВЉАЊЕ КЊИГЕ „МИКА АЛАС, БЕЛЕШКЕ О ЖИВОТУ ВЕЛИКОГ  
МАТЕМАТИЧАРА МИХАИЛА ПЕТРОВИЋА“**

написали Милутин Миланковић и Јеленко Михаиловић,  
приредио Владо Милићевић,  
књигу је представио проф. др Жарко Мијајловић, Универзитет у Београду, Математички  
факултет

**ПРЕДСТАВЉАЊЕ ИЗЛОЖБЕ „МИЛУТИН МИЛАНКОВИЋ“**

др Славко Максимовић, председник удружења „Милутин Миланковић“

## I СЕКЦИЈА: МАТЕМАТИКА И ПРИМЕНЕ ДАНАС

проф. др Градимир Миловановић, академик, Математички институт САНУ  
CONSTRUCTIVE THEORY OF ORTHOGONAL POLYNOMIALS, SOFTWARE, AND  
APPLICATIONS

### **Abstract:**

*In this paper we give the basic notions and results from the so-called constructive theory of orthogonal polynomials, which was developed last thirty years. This theory has enabled many new and unexpected applications in numerical analysis, approximation theory, other mathematical areas, as well as in many areas of computational and applied sciences. Also, we give an overview of the existing software and give some interesting applications.*

**Keywords:** *orthogonal polynomials, numerical analysis, approximation theory*

проф. др Бошко С. Јовановић, Универзитет у Београду, Математички факултет  
ГРАНИЧНИ ПРОБЛЕМИ ЗА ПАРЦИЈАЛНЕ ДИФЕРЕНЦИЈАЛНЕ ЈЕДНАЧИНЕ  
РАЗЛОМЉЕНОГ РЕДА И ЊИХОВО НУМЕРИЧКО РЕШАВАЊЕ

### **Анстракт:**

*Парцијалне диференцијалне једначине разломљеног реда привлаче у последње време све већу пажњу, како у теоријским истраживањима, тако и у моделовању најразноврснијих појава и процеса у природи и друштвеним односима. Такве једначине се користе за описивање широких класа физичких и хемијских процеса (у аморфним полупроводницима, вискоеластичним медијима, материјалима с фракталном геометријом итд.), као и приликом математичког моделовања економских, биолошких и социјалних феномена.*

*Пошто се у дефиницији извода разломљеног реда појављују интегрални изводи, јасно је да су такви изводи нелокални оператори – за разлику од класичних извода целобројног реда. Ова чињеница се може интерпретирати као специфичан меморијски ефекат, због чега изводи разломљеног реда представљају погодан алат за моделовање процеса у материјалима са сличним особинама (вискоеластични материјали, полимери и сл.). С друге стране, ова особина извода разломљеног реда знатно отежава конструкцију брзих и прецизних нумеричких метода за решавање одговарајућих граничних проблема.*

*У овом раду ћемо представити неке примере парцијалних диференцијалних једначина разломљеног реда и указати на главне теоријске и нумеричке проблеме који се при томе појављују. Посебну пажњу ћемо обратити на неке трансмисионе проблеме повезане с парцијалним диференцијалним једначинама разломљеног реда.*

**Кључне речи:** *изводи и интегрални разломљеног реда, парцијалне диференцијалне једначине разломљеног реда, гранични проблем, трансмисиони проблем.*

проф. др Миодраг Матељевић, Универзитет у Београду, Математички факултет  
ЈЕДНАЧИНЕ ЕЛЕМЕНТАРНИХ КРЕТАЊА

**Апстракт:**

*Разматрамо примене математике на елементарна кретања. Један од циљева овог предавања је популаризација науке. Планирамо да у предавању која укључује следеће теме: осцилације, ротационо кретање, Лоренцове трансформације, Ојлерову теорему о ротацији и примену сферених изометрија на оцену капацитета, скицирамо везе између механике, физике и математике.*

**Кључне речи:** *примене математике, елементарна кретања, физика, механика*

Hans Hartmann, Lecturer at the Vienna and Leipzig Universities of Technology  
WHAT PROGRAMMERS CAN LEARN FROM MATHEMATICS

**Abstract:**

*There exists a common misconception about the presence of software bugs in today's programs. Commonly it is believed that software failures – in whatever way they are introduced into software - are a result of human errors.*

*Some results found in the 20th century show that this assumption is not correct. The presentation shows that the very much neglected discipline "program testing" has to be assessed using a different viewpoint when one considers the results of Gödel, Turing or Chaitin. These considerations influence a new way to teach software testers and will show ways to improve on software test management.*

*The presentation is based on several very successful experiences during the past 12 years and will also cover the pitfalls that need to be avoided.*

**Keywords:** *mathematics software-science testing*

др Зоран Ђорђевић, доц. др Александар Перовић, проф. др Александар Јовановић,  
Универзитет у Београду, Математички факултет, Група за интелигентне системе  
ИНТЕЛИГЕНТНИ ПОМОРСКИ СИСТЕМИ

**Апстракт:**

*Потреба да се контролише комплетан поморски саобраћај на свету са циљем да се спречи илегална трговина, кријумчарење дроге, оружја, поморска пиратерија и терористички напади, као и да се помогне у операцијама тражења и спашавања резултовала је увођењем Аутоматског Идентификационог Система АИС. Поред система АИС постоји и осматрачка мрежа сателита и радарских станица – сензора већег и мањег домета. Радарски сензори детектују све објекте на морској површини. Такође, наведено да се у овом домену користе и други разноврсни сензорски системи. Интелигентна фузија податка са радара, система АИС, других сензорских система, релационих база података пловила, која резултује предлозима акција или аутоматизованим акцијама, неопходна је за све намене у поморским системима. Најзначајнији корисници поменутих резултата су Ратне Морнарице и Обалске страже. Предложене акције идентификују прегледе сумњивих објеката и у екстремном случају дејство софицистичаним оружјем системима. Овде се предлаже модел интелигентног система за праћење поморске ситуације који одређује скуп задатака које овакав систем решава.*

**Кључне речи:** *аутоматски идентификациони систем – АИС, обрада радарске слике, фузија података са сензора, објектно оријентисане технологије пројектовања, фази релационе базе података, интелигентни системи*

**Апстракт:**

У свакодневном животу често наилазимо на проблем распоређивања (послова на машинама, радника по сменама, возила на рутама, и сл.). Проблеми паковања су веома слични и такође свеprisутни. Обе групе проблема спадају у категорију проблема комбинаторне оптимизације. Сви ови проблеми могу се прецизно математички формулисати и решавати познатим математичким алатима. Ови алати најгрубље се могу поделити на тачне и хеуристичке методе. На примеру распоређивања задатака на рачунарске мреже биће илустровано како се проблеми комбинаторне оптимизације могу математички описати као и неколико метода за решавање овог проблема.

**Кључне речи:** проблеми распоређивања, проблеми паковања, комбинаторна оптимизација, рачунарске мреже

Милена Чукић, Институт за Неурлогију, КЦС Београд; Медицинска школа Београд  
ФРАКТАЛНА КАРАКТЕРИЗАЦИЈА ПОВРШИНСКОГ ЕЛЕКТРОИМИОГРАМА  
ДОБИЈЕНОГ ТРАНСКРАНИЈАЛНОМ МАГНЕТНОМ СТИМУЛАЦИЈОМ

**Апстракт:**

Површински електромиограм (рЕМГ) је сложен нестационаран сигнал генерисан мишићном активношћу чије су карактеристике детерминисане низом фактора. Претходна истраживања су показала да се фрактална димензија (FD) рЕМГ може употребити да опише сложеност мишићне активности и ниво мишићне контракције. Такође, одређивањем фракталних димензија рЕМГ могу се анализирати карактеристике мишића у смислу величине и сложености покрета. Транскранијална магнетна стимулација (ТМС) је неинвазивна метода којом се могу побудити ексцитаторне и инхибиторне структуре коре великог мозга. Активација ексцитаторних структура доводи до моторног евоцираног потенцијала (МЕП) док активација инхибиторних структура доводи до настанка периода тишине (ПТ) у рЕМГ за време вољне активације мишића.

У овом раду нас је интересовало да ли се FD рЕМГ -а мења након примене ТМС-а. На групи од 10 здравих испитаника поредили смо FD рЕМГ -а пре ТМС-а и након ТМС-индукованог одговора (МЕП+ПТ). рЕМГ је регистрован са првог дорзалног интересеусног мишића десне руке. Од испитаника је тражено да одржавају исти ниво мишићне контракције пре и после ТМС-а. FD рЕМГ -а је рачуната по Хигуџи-јевом алгоритму за одређивање франтквалне димензије сигнала. Резултати показују да је дошло до статистички значајне разлике у средњем интензитету рЕМГ -а пре и после ТМС-а. Као контролна група снимљено је 5 испитаника (изабраних из прве групе) који нису добијали стимулус, већ су само одржавали исте нивое вољне контракције у циљном мишићу. Код тих серија није утврђена статистички значајна промена комплексности сигнала.

Из ранијих истраживања где смо поредили резултате спектралне анализе са фракталним налазима, поредивши два модалитета стимулуса (ТМС са периферном стимулацијом истог циљног мишића) утврдили смо да до промене долази само кад се та структура стимулише централно. Овај налаз на већој групи и у поређењу са контролама код којих није извршена стимулација, дошли смо до закључка да ТМС (једнопулсна) ипак уводи промену комплексности сигнала која је мерљива, можда нам пружајући путоказ за даље студије пластичности коре великог мозга.

проф. др Весна Јевремовић, мр Марко Обрадовић, Универзитет у Београду, Математички факултет

## ДА ЛИ ЈЕ БЕРТРАНОВ ПАРАДОКС ПАРАДОКС?

### **Анстракт:**

*Интуитивни приступ случајности код геометријских вероватноћа може довести до проблема. Најпознатији пример је Бертранов парадокс који више од 100 година изазива пажњу математичара који покашавају да га „разреше“. На предавању ће бити речи о неким покушајима решавања овог парадокса, као и његовог проширивања.*

**Кључне речи:** геометријске вероватноће, Бертранов парадокс

проф. др Милутин Обрадовић, Универзитет у Београду, Грађевински факултет;  
prof. dr Saminathan Ponnusamy, Indian Institute of Technology, Chennai, India.

## UNIVALENCE AND STARLIKENESS OF SECTIONS OF UNIVALENT FUNCTIONS

### **Abstract:**

*Let  $D := \{z \in \mathbb{C} : |z| < 1\}$  be the open unit disk in the complex plane  $\mathbb{C}$ . Let  $S$  denote the family of all functions  $f$  that are analytic and univalent in  $D$  with the normalization  $f(0) = 0 = f'(0) - 1$ . For*

*$f(z) = z + \sum_{k=2}^{\infty} a_k z^k$  in  $S$  and  $n \in \mathbb{N}$ , we write*

$$s_n(f, z) := z + \sum_{k=2}^n a_k z^k$$

*for the  $n$ -th partial sums or sections of  $f$ .*

*In this lecture, we determine conditions so that each section  $s_n(f, z)$  of  $f \in S$  is univalent or starlike in some disk  $|z| < r_n$ . Also we discuss the same problems for certain special classes of the class  $S$ .*

**Keywords:** univalent, starlike, section

др Катица Р. (Стевановић) Хедрих, Математички институт САНУ, Београд; Универзитет у Нишу, Машински факултет

## ТРИГЕР СПРЕГНУТИХ СИНГУЛАРИТЕТА

### **Анстракт:**

*Неколико теорема о тригеру спрегнутих сингуларитета је дефинисано и приказују примене на примерима хибридне фасцинантне динамике материјалне тачке, односно крутог тела, који изводе спрегнуте ротације око мимоилазних оса у гравитационом пољу. Указује се на феномене резонанције, динамичке апсорбције, параметарске резонанције, синхронизације, резонантног скока у условима принудне хибридне динамике хибридног система или хибридне динамике “једноставног” система. Истичу се и примери феноменолошког пресликавања и математичке аналогije.*

**Кључне речи:** нелинеарна динамика, тригер спрегнутих сингуларитета, кинетика релатиивног мировања, фазни портрет, стабилност, спрегнуте ротације.

др Владимир Буљак, доцент, Универзитет у Београду, Машински факултет  
Giulio Maier, Gabriella Bolzon

## PROPER ORTHOGONAL DECOMPOSITION IN MODEL REDUCTION FOR FAST SIMULATIONS

### **Abstract:**

*Model reduction techniques nowadays represent quite popular research topic. The results achieved within this research field are finding their applications in several engineering fields oriented to either real-time simulations of complex phenomena, or to the parameter characterization based on inverse analysis. Vast majority of developed model reduction techniques are dividing the computing work into two phases: an off-line phase that is done once-for-all, and usually contains “heavy” computations; and an on-line phase that makes the use of previously generated data and performs “light” computation providing the results in a real-time. The technique which will be presented within this communication is based on the combination of “Proper Orthogonal Decomposition” (POD) with “Radial Basis Functions” (RBF). Within the first phase of this method, a set of analysis with “full” numerical model (e.g. using finite element method or boundary element method) are performed varying some of the parameters that are of further interest. Thus generated responses, that are called “snapshots” in pertinent jargon, are collected within a single matrix – the snapshot matrix. Each snapshot represents a response of the same system when only a few parameters are changed, which suggests that there would be a strong correlation between the resulting data. This means that by performing a POD analysis over snapshot matrix a new reference basis can be obtained, in which a significant reduction of dimensionality without any practical loss of the accuracy can be achieved by preserving only first few directions. After the truncation is performed, snapshot matrix is expressed with matrix of “amplitudes” referring to the reduced space. In the subsequent phase interpolation by Radial Basis Functions is further employed on thus prepared data.*

*By combining these two mathematical techniques into a practical procedure it is possible to obtain a robust computational tool capable of computing responses of highly non-linear systems in a time shorter by several orders of magnitude with respect to traditionally used FEM procedures, without any practical loss of accuracy. The developed reduced basis models can successfully be used for real time simulations of different non linear processes providing the results in a real time. Within this communication some of recently achieved results of the application of this technique in material characterization and in real time simulation problems will be presented.*

**Keywords:** proper orthogonal decomposition, radial basis functions, numerical simulations, model reduction

## II СЕКЦИЈА: МАТЕМАТИКА И ИНФОРМАТИКА У ОБРАЗОВАЊУ

проф. др Ђурђица Такачи, Универзитет у Новом Саду, Природно-математички факултет  
РЕШАВАЊЕ ЗАДАТАКА ВИЗУАЛИЗАЦИЈОМ

### **Анстракт:**

*Приказује се дидактички приступ добијања решења задатака и проблема, коришћењем програмског пакета GeoGebra, са посебним освртом на његове динамичке особине. Анализира утицај визуализације на процес решавања проблема који се заснива на познатом фазном концепту Г. Полуа. Разматра се визуализација различитих проблема и њихових решења, почев од елементарних, (докази елементарних теорема), које решавају ученици основних школа, до примера из наших најновијих научних истраживања (фракциони изводи).*

**Кључне речи:** дидактички приступ решавању проблема, динамичка геометрија, фракциони изводи

мр Дарко Дракулић, Универзитет у Источном Сарајеву, Филозофски факултет Пале  
РАЗВОЈ ЕЛЕКТРОНСКИХ УЏБЕНИКА МАТЕМАТИКЕ У РЕПУБЛИЦИ СРПСКОЈ

### **Анстракт:**

*У раду ће бити описане активности на изради електронских уџбеника из математике за четврти и шести разред основних школа у Републици Српској. Израда ових уџбеника реализује се у оквиру билатералног пројекта о интерактивном електронском учењу математике чији су носиоци Природно-математички факултет Универзитета у Марибору и Филозофски факултет Универзитета у Источном Сарајеву.*

**Кључне речи:** е-учење, е-уџбеници

проф. др Миодраг Матељевић, Универзитет у Београду, Математички факултет,  
Нивес Јозић, Филозофски факултет у Сплиту,  
Марек Светлик, Универзитет у Београду, Математички факултет,  
ИСТРАЖИВАЊЕМ ДО МИНИМУМА ИЛИ МАКСИМУМА

### **Анстракт:**

*Проблеми оптимизације су честа појава у реалном животу. Систем образовања а посебно настава математике треба да оспособе ученике да решавају и такве проблеме, односно да у датом контексту одреде најповољније решење. Оптимизација се често састоји у томе да на основу фиксираних вредности неких величина одредимо минималну односно максималну вредност неке друге величине. Који садржаји су прикладни за развијање те компетенције те како их обрађивати да би се постигао жељени исход учења те развиле способности одлучивања? Рутинско решавање бројних задатака утврђује научене процедуре, вештину и брзину рачунања, али врло мало способност одлучивања.*

*Да би ученици развиле способност одређивања (нај)повољнијег решења треба бирати такве задатке у којима се ученик ставља у ситуацију да самостално истражује могућности, спроводи анализу и тиме развија интуицију за откривањем траженога, изводи закључке и поставља формалне тврдње те схвата потребу за потврђивањем истинитости уоченог. Тако нпр. уместо да дамо готово правило да је од свих троуглова једнаког обима једнакостранични троугао највеће површине, можемо поставити ученике у ситуацију да истраже и закључе како би од жице неке произвољне дужине обликовали троугао највеће површине.*

*У овом раду је циљ показати како се ученици могу водити кроз истраживачки процес у откривању оптималног решења и усмеравати на критичко размисљање и учење с разумевањем. Биће разматрани проблеми одређивања минимума и максимума површине или обима раванских фигура.*

**Кључне речи:** минимум и максимум површине и обима, изопериметријска неједнакост, истраживање у настави математике

Наталија Будински, професор математике, основна и средња школа са домом ученика „Петро Кузмјак“, Руски Крстур  
УЧЕЊЕ МАТЕМАТИЧКИХ ФУНКЦИЈА МОДЕЛИРАЊЕМ ПРОБЛЕМА ИЗ РЕАЛНОГ ЖИВОТА

**Анстракт:**

*Примена математичког моделирања се одвија по шема активности која почиње упознавањем ученика са ситуацијом из реалног живота, а завршава се математичким решењем. Моделирање приказује математику у примењивом светлу, а код ученика развија способност решавања различитих проблема. Ово предавање даје пример како применити моделирање на часовима и намењено је наставницима математике. Дају се примери учења математичких функција моделирањем проблема из реалног контекста и анализирају се предности моделирања у настави математике. Такође, ставља се акценат на употребу рачунара и едукативног софтвера GeoGebra у процесу моделирања у настави математике.*

**Кључне речи:** моделирање, проблеми из реалног контекста, математичке функције, GeoGebra

мр Јелена Хаџи Пурић, Универзитет у Београду, Математички факултет  
Невенка Спалевић, Јасминка Михаљинац, Математичка гимназија, Београд  
FROM TOYS TO COMPUTER PROGRAMMING

**Abstract:**

*This paper aims to be a contribution to the integration of multimedia applications and programming into school and preschool education system. Our "digital kids" are capable to use computers and Internet, but it may be valuable to teach kids how to code. Our survey is based on experiences and observations of children aged 4-8 years olds using an animated programming language, Scratch. Those courses were held in weekly session over the several months in each year, involving approximately 220 children and 4 educators. Illustrative user manual and educational material on the initial use of computers, computer networks and computer programming are included, as well as an informative survey on existing initial learning environments for children. Also, we have made an effort on localization of software package Scratch into Serbian language. Promoting computer literacy through programming, even at an elementary level, makes technology seem less unfamiliar and more accessible. The results showed that in general kids (K-4) could successfully learn basic concepts of computer science. Also, animated programming languages encourage kids to explore basic mathematical concepts.*

**Keywords:** introduction to programming, preschool education, initial programming in primary school

Александра Росић, спец. наставе математике, ОШ „Мирослав Антић“, Београд  
ПРИМЕНА IPAD-А У НАСТАВИ МАТЕМАТИКЕ

**Анстракт:**

*Сви ми који смо радили у пар школа суочили смо се са немогућношћу извођења наставе по активним принципима углавном због недостатка средстава. Да ли чекајући нека боља времена, паметне табле и iPad-ове заборављамо да нам ученици из школа излазе са ниским нивоом функционалне математичке писмености?*

*На неколико практичних примера даће се приказ коришћења iPad-а у настави математике, али и алтернативна решења како уз помоћ једноставних средстава обогатити наставу математике.*

**Кључне речи:** наставна средства, iPad у настави математике



Радомир А. Михајловић, New York Institute of Technology,  
Невенка Спалевић, Математичка гимназија, Београд  
TEACHING HIGH ACHIEVERS COMPUTER ARCHITECTURE AND OPERATING  
SYSTEMS

**Abstract:**

*The so called personal computing and Internet revolution has brought a powerful computing machine as a basic necessity into almost all homes world wide.*

**Keywords:** *gifted student, hih achiever, computer architecture, operating system*

доц. др Милена Богдановић, Универзитет у Нишу, Учитељски факултет у Врању  
РАЧУНАРСКА (ИНФОРМАТИЧКА) ПИСМЕНОСТ И ИНТЕРНЕТ – ПРЕДНОСТИ И  
ОПАСНОСТИ ЗА КОРИСНИКЕ

**Анстракт:**

*У раду се објашњава неопходност рачунарске (информатичке) писмености, почев од предшколских установа и основне школе, па све до високог образовања, у савременим светским тенденцијама у науци и свакодневном животу. Наравно, није заобиђен ни ИНТЕРНЕТ, као једна од најзначајнијих могућности савремене технике и технологије у свакодневном и академском животу. Предности велике мреже ресурса и слободног протока информација веће су од опасности злоупотребе Интернета. Али, противзаконито коришћење Интернета не може се толерисати. Борба против илегалног садржаја на глобалном нивоу, као што је дечија порнографија, мора бити загарантована. Сви постојећи закони морају бити, такође, примењени на Интернету. Рачунарска и Интернет писменост мора бити стимулисана како би се ојачало техничко разумевање важности софтвера и кода. Ово је потребно како би се оставио простор за могућност дефинисања будуће улоге Интернета и његовог положаја у грађанском друштву. Интернет писменост мора бити примарни циљ образовања у школама. Потребно је органозовати и курсеве за одрасле.*

**Кључне речи:** *интернет, сервиси интернета, веб странице, chat, блог, download, форум, веб галерија.*

мр Драган Матић, Природно-математички факултет, Универзитет у Бањалуци,  
проф. др Милан Божић, Универзитет у Београду, Математички факултет  
ПРИМЈЕНА ПРОБЛЕМА МАКСИМАЛНЕ ПОВЕЗАНЕ БАЛАНСИРАНЕ  
ПАРТИЦИЈЕ У ОБРАЗОВАЊУ

**Анстракт:**

*Проблеми комбинаторне оптимизације проналазе широку примјену у рјешавању разних организационих проблема. Познати НП тежак проблем максималне повезане балансиране партиције у графу (Maximally Balanced Connected Partition problem - MBСP) је искориштен за организовање подјеле наставних јединки једног курса на два дијела, тако да се очува повезаност јединки унутар сваког дијела, а да тежина градива у оба дијела буде приближно једнака. У раду је предложена једна подјела курса из Увода у теорију бројева. Такође, описане су неке методе рјешавања овог проблема.*

**Кључне речи:** *партиције графа, рачунари у образовању, организација курса*

Богољуб Маринковић, Математичко друштво „Архимедес“, Београд  
МАТЕМАТИЧКО ДРУШТВО „АРХИМЕДЕС“ БЕОГРАД – ДОПРИНОС  
ОБРАЗОВАЊУ И ПОДИЗАЊУ МАТЕМАТИЧКОГ ПОДМЛАТКА

**Апстракт:**

Свака наука свој прогресивни развој остварује, поред осталог, систематском и конти-нуираним бригом о свом подмлатку, како у институционалном, тако и ван-институционалном организовању подршке младима. У том смислу је Математичко друштво „Архимедес“ из Београда (до недавно: Клуб младих математичара "Архимедес"), у својој дугој пракси (38 година), постигло изванредне резултате и дало велики допринос унапређивању математичког образовања младих генерација, што је високо оцењено у стручној јавности у нашој земљи и иностранству. О томе сведоче бројни подаци из архиве овог специфичног стручног друштва и многа признања која су "Архимедесовци" добијали за показано високо умеће у математици и у организовању активности са љубитељима математике свих нивоа.

„Архимедес“ је поодавно постао прави специјализовани центар за даровите младе математичаре и друге љубитеље математике и рачунарства свих узраста. До сада је регистровано око 28.400 чланова (1660 одраслих и 26.700 ученика и студената). Хиљаде ученика и стотине наставника сарађују са "Архимедесом" и учествују у његовим активностима. Бројне и обимне активности чине осмишљен систем: популаризација математике и науке уопште, непосредни рад са ученицима-љубитељима математике и рачунарства (стална школа која ради током целе школске године, летње и зимске школе-кампови, дописна школа, београдска математичка секција, рачунарски курсеви); математичка такмичења: Математички турнир – екипна математичка олимпијада основних и средњих школа Републике Србије, масовно такмичење „Мислишиа“, дописна математичка олимпијада, интернет математичка олимпијада, Међународни математички турнир градова (организатор за Београд, координатор за Србију); стручни рад са наставницима (стручне трибине, специјализовани и други семинари), специјализована библиотека, стварање и издавање приручне литературе за ученике и наставнике.

**Кључне речи:** Архимедес, математички подмладак, популаризација математике

Петар Огризовић, Образовни систем „Руђер Бошковић“, Београд  
ОБЛАСТ „МЕРЕЊЕ“ У НАСТАВИ МАТЕМАТИКЕ У ОСНОВНОЈ ШКОЛИ

**Апстракт:**

У овом излагању анализиран је задатак „Разговор преко интернета“ (ПИСА, 2003. година) кроз резултате тестирања 176 ученика Основне школе и Гимназије „Руђер Бошковић“ у Београду.

Делове овог задатка, иначе намењеног петнаестогодишњацима, успешно решавају и ученици млађих разреда основне школе. С тим у вези, разматране су лекције из области „Мерење“, која постоји у Образовним стандардима за крај обавезног образовања (Национални просветни савет, 2009), али је нема у наставним плановима за старије разреде основне школе. Питање је, дакле, како ученици, на међународним тестовима, решавају задатке којима се мало или нимало баве у редовној настави.

Биће дато и неколико примера и предлога како се мерење може детаљније обрадити кроз друге наставне области.

**Кључне речи:** мерење, образовни стандарди, ПИСА, наставни планови

мр Наталија Јеленковић, гимназијски професор математике и информатике, Београд  
ПРИМЕНА МАТЕМАТИКЕ У КАРТОГРАФИЈИ

**Апстракт:**

Саопштење садржи прво кратки историјски развој примене математике у картографији кроз истицање основних метода и поступака које директно користе математику, затим, визуелно приказивање старих карата и атласа за истицање тачности, шире презентовање коришћењем примера и модела, опис неких занимљивости које истичу улогу и примену математике као науке у картографији данас, као и приказ астрономских мапа, војних карата, картографских задатака кроз садашњост, прошлост и будућност.

**Кључне речи:** математичка картографија, картографски задаци, картографске пројекције, топографске карте, мрежа триангулације, рачунарска картографија, државни координатни систем, дигитализација карата.

Драган Петровић, професор математике, Гимназија Крушевац  
ПРИМЕНА ГРАФИКА ФУНКЦИЈЕ У РЕШАВАЊУ ЈЕДНАЧИНА

**Апстракт:**

Већина ученика средње школе график функције користи само ако се у задатку каже „Нацртај график функције...“. Мали број њих је оспособљен да график функције користи и приликом решавања једначина, посебно ако изрази нису истородни као на пример комбинација полинома и логаритамских, експоненцијалних или тригонометријских израза. И одређивање колико, на пример, има реалних решења једначина трећег степена за њих је велики проблем. Упоредивање решења квадратне једначине са заданим бројем или бројевима такође се може решавати уз коришћење графика квадратне функције. Циљ овога је да се ученици оспособе да користе график функције и ако се у задатку то експлицитно не каже.

**Кључне речи:** једначине, график функције

Мирослав Гојић, мр Горан Лазовић, проф. др Слободан Радојевић, Универзитет у Београду,  
Машински факултет

СПЕЦИЈАЛИЗАЦИЈА МАШИНСКИХ ИНЖЕЊЕРА У КОРИШТЕЊУ  
САСТАВНИЦА МОДЕЛИРАНИХ СТАБЛОМ

**Апстракт:**

Саставница је један од основних документа који прати производњу. Саставница као документ у производњи даје основне информације о производу у читавом процесу производње истог. Образовање и оспособљавање машинских инжењера за кориштење саставница моделираних стаблом отвара могућност унапређења производње. У саставнице се могу додавати и додатне информације које олакшавају праћење саме производње тако да исту чине предвидивом.

**Кључне речи:** саставница, производња, организација, стабло, структура, базе података, планирање.

### III СЕКЦИЈА: НАУЧНОИСТРАЖИВАЧКИ И СТРУЧНИ РАД СТУДЕНАТА

Сандра Хоцић, сарадник у настави, студент докторских студија, Универзитет у Београду, Математички факултет

#### ПРОБЛЕМ СОПСТВЕНИХ ВРЕДНОСТИ СА НЕЛОКАЛНИМ РОБИН-ДИРИХЛЕОВИМ ГРАНИЧНИМ УСЛОВИМА

**Апстракт:**

*Штурм-Лиувилове једначине се појављују у проблемима примењене математике. На пример, оне представљају вибрациона стања различитих система, као што су вибрације жице, или енергетске сопствене функције квантно-механичких осцилатора. То су гранични проблеми обичних диференцијалних једначина, који се још називају и проблемима сопствених вредности. Овде је изложен и решен систем са тзв. Нелокалним Робин-Дирихлеовим граничним условима са примерима који илуструју претходна теоријска разматрања.*

**Кључне речи:** Штурм-Лиувилова једначина, гранични проблем

Љубица Матић, асистент, Висока технолошка школа, Шабац, студент докторских студија на Математичком факултету, Универзитет у Београду

#### О ЈЕДНОМ ТИПУ ТАЛАСНЕ ЈЕДНАЧИНЕ; ПРИМЕР РЕШАВАЊА МУЛТИГРИД МЕТОДОМ

**Апстракт:**

*Разматраћемо осцилације две танке жице које су учвршћене на једном свом крају, док им је супротан крај слободан. Специфичност задатка се огледа у чињеници да се решење посматра у два дисјунктна, строго раздвојена интервала. Физичка појава осцилације танке жице је објашњена, добро дефинисана и поставља се питање како јој доделити одговарајући математички модел (који би требало да садржи и променљиве које описују еластичност жица, утицај спољашњих сила...). Након тога, модел је трансформисан до облика погодног за примену мултигрид методе. Мултигрид је итеративни метод базиран на решавању система на „грубљој“ мрежи и каснијом поправком иницијалног решења. Ефикасност методе зависи од избора и начина комбиновања његових основних елемената- итеративне методе којом се систем решава на нивоима, оператора рестрикције и пројекције помоћу којих се врши прелаз са једне мреже на другу.*

**Кључне речи:** таласна једначина, мултигрид метода

Вања Николић, сарадник у настави, Универзитет у Београду, Факултет организационих наука

#### ДВОМРЕЖНА МЕТОДА КОНАЧНИХ ЕЛЕМЕНАТА

**Апстракт:**

*У раду је приказана техника дискретизације методом коначних елемената базирана на два простора коначних елемената, једном који одговара грубљој и једном који одговара финијој мрежи. Разматран је проблем решавања нераздвојеног система линеарних парцијалних диференцијалних једначина, као и проблем сопствених вредности за линеарне парцијалне диференцијалне операторе. Примењујући двомрежну методу полазимо проблем решавамо на грубљој мрежи, док на финијој мрежи решавамо знатно једноставнији проблем (Пуасонову једначину, раздвојен систем парцијалних једначина, итд.) при чему уз одговарајући избор грубље мреже решење задржава оптималну тачност. Нумеричким експериментима демонстрирана је ефикасност методе.*

**Кључне речи:** метода коначних елемената, две мреже, систем парцијалних диференцијалних једначина, проблем сопствених вредности

дипл. фарм. Тијана Ракић, Универзитет у Београду, Фармацеутски факултет, Александар Ђенић, студент докторских студија, др Мирослав Марић, др Зорица Станимировић, Математички факултет, Универзитет у Београду, др Биљана Јанчић Стојановић, Фармацеутски факултет, Универзитет у Београду,  
**РАЗВОЈ НОВИХ МОДЕЛА ПРЕДВИЂАЊА РЕТЕНЦИОНОГ ПОНАШАЊА У ТЕЧНОЈ ХРОМАТОГРАФИЈИ ХИДРОФИЛНИХ ИНТЕРАКЦИЈА**

**Апстракт:**

*У овом истраживању развијени су модели предвиђања ретенционог понашања пет антидепресива у HILIC систему. Предложени модели засновани су на апроксимацијама помоћу квадратне форме другог реда, подељених разлика, и коначних разлика функције више променљивих. Циљ истраживања је поређење ефикасности различитих модела и оптимизација раздвајања применом функције хроматографског одговора.*

**Кључне речи:** хроматографија, ретенционо понашање, апроксимација

Марина Пекић, Сања Рибич, Универзитет у Београду, Фармацеутски факултет, Катедра за аналитику лекова, ментори: дипл. фарм. Тијана Ракић, др Биљана Јанчић Стојановић  
**ФУНКЦИЈЕ ХРОМАТОГРАФСКОГ ОДГОВОРА У РЕШАВАЊУ ОПТИМИЗАЦИОНИХ ПРОБЛЕМА У ТЕЧНОЈ ХРОМАТОГРАФИЈИ ХИДРОФИЛНИХ ИНТЕРАКЦИЈА**

**Апстракт:**

*Функције хроматографског одговора представљају математичко решење које омогућава објективну процену квалитета хроматограма и допушта истовремену оптимизацију више различитих хроматографских циљева. Циљ је процена ефикасности четири различите функције хроматографског одговора на симулираним хроматограмима и експериментално добијеним хроматограмима након анализе бета агониста и блокатора течном хроматографијом хидрофилних интеракција (HILIC – енг. Hydrophilic Interaction Liquid Chromatography). Пет симулираних хроматограма креирано је у Microsoft Excel програму. Осамнаест експериментално добијених хроматограма генерисано је након анализе пет бета агониста и блокатора под различитим условима у HILIC систему (стационарна фаза: BETASIL Силица-100 (100 mm x 4.6 mm, 5 µm величине честица), мобилна фаза: ацетонитрил: водени раствор амонијум-ацетата, чији је рН подешен глицеролном сирћетном киселином). Састав мобилне фазе вариран је према плану централног композиционог дизајна. Четири функције хроматографског одговора примењене су у процени симулираних хроматограма. Анализирани су различити сепарациони параметри укључени у функције и њихова способност да истовремено процењују квалитет раздвајања свих присутних пикова. Испитани су приступи процени укупне дужине трајања хроматографске анализе. Мерна је способност функција да поставе адекватан баланс између процене квалитета раздвајања и укупне дужине трајања анализе. Функције су тестиране на експериментално добијеним хроматограмима. Примењене су значајне разлике у вредновању квалитета хроматограма од стране различитих функција. Показано је да функције хроматографског одговора представљају поуздан и објективан приступ у оптимизацији раздвајања супстанци у HILIC систему. Проучене су предности и мане различитих функција и дефинисане су мере опреза при одабиру функције избора за дати оптимизациони проблем. Функције хроматографског одговора, HILIC, бета агонист и блокатори.*

**Кључне речи:** функције хроматографског одговора, HILIC, бета агонист и блокатори

Стефан Мишковић, сарадник у настави, студент докторских студија, Универзитет у Београду, Математички факултет  
ХИБРИДНИ ГЕНЕТСКИ АЛГОРИТАМ ЗА ЈЕДНОСТЕПЕНИ ЛОКАЦИЈСКИ ПРОБЛЕМ ОГРАНИЧЕНИХ КАПАЦИТЕТА

**Апстракт:**

У раду је приказан хибридни генетски алгоритам за једноступени локацијски проблем ограничених капацитета. За дати скуп терминала и потенцијалних локација за концентраторе, потребно је одредити који ће концентратори бити успостављени тако да сума цена придруживања терминала концентраторима и цена инсталације концентратора буде најмања могућа. Притом је потребно водити рачуна да капацитети концентратора одговарају потражњама терминала. Хибридни генетски алгоритам користи елитистичку стратегију, фино градирану турнирску селекцију, једнопозиционо укрштање и просту мутацију. За израчунавање функције циља је коришћен CPLEX. У свакој генерацији се примењују и две помоћне хеуристике локалне претраге, инверзија и транспозиција. Алгоритам је тестиран на ORLIB инстанцама и за сваку тест инстанцу је достигнут оптималан резултат.

**Кључне речи:** локацијски проблеми, генетски алгоритам, локална претрага

Оливера Јанковић, сарадник у настави, Универзитет у Крагујевцу, Економски факултет  
CYCLE-CANCELING ALGORITHM ALGORITHM ЗА РЕШАВАЊЕ ПРОБЛЕМА ПРОТОКА СА МИНИМАЛНОМ ЦЕНОМ

**Апстракт:**

У пракси, проблем протока са минималном ценом се често среће у скоро свим гранама индустрије, јер за решавање највећег броја практичних проблема, одлучујући значај имају трошкови неопходни за реализацију посматраних протока. У раду је показано да је овај проблем еквивалентан транспортном проблему, а затим је описан „Cycle – canceling algorithm“ који налази оптимално решење у псеудополиномијалном времену. Његов рад је илустрован на кратком примеру.

**Кључне речи:** cycle – canceling algorithm, проблем протока са минималном ценом, услов оптималности, резидуална мрежа

Милош Станковић, студент мастер студија, Универзитет у Београду, Математички факултет  
ХЕУРИСТИКА ТАБУ ПРЕТРАЖИВАЊА ЗА РЕШАВАЊЕ GAP-А

**Апстракт:**

У раду је разматран GAP (Generalized Assignment Problem), познати НП-тешки проблем комбинаторне оптимизације, чије решавање је од великог практичног значаја. GAP подразумева придруживање скупа послова скупу агената који их извршавају. Сваки агент има ограничене ресурсе и сваки посао се мора доделити тачно једном агенту заузимајући одређену количину његових ресурса. За решавање GAP -а у раду се примењује хеуристика табу претраживања. Алгоритам табу претраживања се заснива на адаптивној меморији која у процесу претраге има улогу у превазилажењу локалних оптимума и приближавању глобалном оптимуму. Најзначајније карактеристике овог алгоритма су једноставност и флексибилност. У раду је детаљно приказана имплементација предложеног алгоритма и његових компоненти као и анализа добијених решења.

**Кључне речи:** табу претраживање, локално претраживање, GAP, generalized assignment problem, метахеуристика, комбинаторна оптимизација

Јасна Рајачић, студент мастер студија, Универзитет у Београду, Математички факултет  
РЕШАВАЊЕ ЈЕДНОДИМЕНЗИОНОГ ВРР ПОМОЋУ ANT COLONY  
OPTIMIZATION

**Апстракт:**

У овом раду је разматран једnodимензиони *Bin Packing Problem* и његово решавање метахеуристико базираном на мрављим колонијама (*Ant Colony Optimization-ACO*). *ACO* је релативно нова метахеуристика која се успешно користи за решавање проблема комбинаторне оптимизације. Како је познато да овај приступ *VRP*-у боље резултате даје у комбинацији са *local search*-ом, искоришћена је ова идеја у циљу што ефикаснијег решавања проблема. Размотрен је алгоритам и упоређени су резултати са другим алгоритмима који су до сада давали добре резултате при решавању овог проблема.

**Кључне речи:** једnodимензиони *bin packing problem*, *ant colony optimization*, метахеуристика

Маја Ђукић, студент докторских студија, Универзитет у Београду, Математички факултет  
ХИБРИДНИ ГЕНЕТСКИ АЛГОРИТАМ ЗА РЕШАВАЊЕ ХАБ ЛОКАЦИЈСКОГ  
ПРОБЛЕМА НЕОГРАНИЧЕНИХ КАПАЦИТЕТА СА ВИШЕСТРУКИМ  
АЛОКАЦИЈАМА

**Апстракт:**

У овом раду је описан генетски алгоритам за решавање хаб локацијског проблема неограничених капацитета са вишеструким алокацијама. Овај НП-тежак проблем има значајну примену у пракси. Хаб локацијски проблеми се доста користе у модерним транспортним и телекомуникацијским системима. Најчешће се примењују за дизајнирање друмских и железничких система, поштанских система, система брзе испоруке и слично. У циљу побољшања ефикасности предложеног генетског алгоритма, примењена је хибридизација генетског алгоритма са хеуристиком локалног претраживања, па су тако настале методе које су веома успешне и при решавању проблема великих димензија. У раду су дата решења и за проблеме великих димензија за које оптимална решења нису позната. Генетски алгоритам је тестиран на инстанцама чије оптимално решење није познато, али дао је резултате за које се може претпоставити да су квалитетни.

**Кључне речи:** генетски алгоритми, хаб локацијски проблеми, комбинаторна оптимизација, метахеуристике

Бојан Бањац, Гордана Димитријевић, Смиљана Обрадовић, Никола Ајдуковић, доц. др  
Бранко Малешевић - Универзитет у Београду, Електротехнички факултет  
Милан Чампара, мастер студент, Универзитет ЕТН Цирих, Швајцарска  
ПРИМЕНА ГРЕБНЕРОВИХ БАЗА НА ПРОБЛЕМ ИНВЕРЗНЕ КИНЕМАТИКЕ У  
РОБОТИЦИ

**Апстракт:**

У оквиру рада је представљена примена Гребнерових база полиномијалних идеала на проблем инверзне кинематике у роботизи. Апликација развијена у оквиру рада визуелно демонстрира на који начин се ова област математике може применити и на постојеће индустријске роботе.

**Кључне речи:** Гребнерове базе полиномијалних идеала, роботика

Милена Кресоја, истраживач-сарадник, Универзитет у Новом Саду, Природно-математички факултет

## СКРИВЕНИ МОДЕЛИ МАРКОВА КАО МЕТОДА ЗА ГЕНЕРАЦИЈУ СЦЕНАРИЈА

### **Апстракт:**

*Када је реч о проблемима финансијске оптимизације, основна питања која се постављају везана су за генерацију сценарија будућих реализација временских серија. Геометријско Брауново кретање и ARCH-GARCH модели важе за најпознатије генераторе сценарија, али оба модела су прихваћена са одређеним резервама. Основне кртике ових модела су следеће. Геометријско Брауново кретање претпоставља да су и дрефт и волатилност константне величине, док ARCH-GARCH модели допуштају да се волатилност мења у времену, али задржавају дрефт константним. Како је данас општеприхваћена чињеница да су и дрефт и волатилност стохастичке величине, у овом раду предлаже се приступ генерацији сценарија базираном на моделима скривених ланаца Маркова. Финансијске временске серије се моделирају помоћу скривених модела Маркова, са геометријским Брауновим кретањем у сваком стању дозвољавајући параметрима геометријског Брауновог кретања (дрифту и волатилности) да се мењају кроз различита стања и стога објасне важне карактеристике временских серија.*

**Кључне речи:** геометријско Брауново кретање, скривени модели Маркова, генерација сценарија

Богдан Обадић, студент, Универзитет у Београду, Математички факултет

## ПСЕУДО-СЛУЧАЈНОСТИ У ТЕОРИЈИ ВЕРОВАТНОЋА

### **Апстракт:**

*Циљ овог научног рада је класификација генератора псеудо-случајних бројева ради њихове лакше примене у симулацијама случајних догађаја у теорији вероватноће. Како ниједан генератор псеудо-случајних бројева није савршен, из простог разлога што је детерминистичке природе, оваква класификација нам даје увид у предности и мане генератора. Само добрим одабиром генератора, у зависности од проблема пред којим се налазимо, можемо постићи најбље резултате. Такође, биће нам много лакше да се одлучимо између квалитета неке оцене и брзине оцењивања вероватноћа. Може се констатовати да услови у којима се неки случајни догађаји одигравају јесу неприступачни за мерење, превише скупи, или пак непрактични. Због тога је симулација случајних догађаја некада неизбежна, а извор псеудо-случајности на детерминистичкој машини није увек погодан за рад. Брз развој информационих технологија које се преплићу са другим научним дисциплинама дају све већи допринос у избору бољих и ефикаснијих извора псеудо-случајности, где због њихових уских примена за конкретну намену морамо имати увид у све, не бисмо ли одабрали нама најбољи.*

**Кључне речи:** случајност, псеудо-случајност, генератори, линеарни конгруентни генератор, Мерсен, Мерсенов твистер, симулација, вероватноћа, експеримент



Ђорђе Баралић, истраживач-сарадник, Математички институт САНУ, Београд  
Игор Спасојевић, ученик, Математичка гимназија, Београд  
ГДЕ СЕ САСТАЈУ ГЕОМЕТРИЈА И ПОЛИНОМИ?

**Апстракт:**

У овом раду су примењене теореме елементарне алгебарске геометрије на класичне теореме пројективне геометрије, Паскалову и Папосову теорему. Објашњена је дубока веза између ових теорема у духу једноставне теореме за кубике. На овај начин су добијена и нова тврђења у духу ових теорема за мистични осмоугао уписан у конику. Ова тврђења се могу добити као последица Безуове теореме, али овде су она доказана много једноставнијим техникама. Разматране су и даље генерализације ових теорема и везе са другим отвореним проблемима у математици.

**Кључне речи:** мистични шестоугао и осмоугао, конике, кубике, алгебарска геометрија  
метахеуристика

Матија Брашњевић, Александар Грујић Ђајић, Владица Васиљевић, Марија Ненезић, Ивана Јововић, Тамара Коледин, Бранко Малешевић – Универзитет у Београду, Електротехнички факултет

НЕКИ ЕЛЕМЕНТИ МАТЕМАТИЧКЕ АНАЛИЗЕ СА ПРИМЕНАМА  
РЕАЛИЗОВАНИ У ПРОГРАМСКОМ ПАКЕТУ GeoGebra

**Апстракт:**

Коришћењем алата GeoGebra реализован је низ интерактивних web страница за приказ решавања неких проблема и примера из симболичке и нумеричке интеграције, граничних вредности функција и диференцијалног рачуна. Неки од ових аплета се ослањају Computer Algebra System (CAS) који GeoGebra нуди. У делу који се бави интеграцијом приказани су симболичка интеграција одређеног и неодређеног интеграла, нумеричка интеграција, Риманове суме као и примена интеграције за налажење дужине лука криве, површине између две криве и израчунавање запремине и површине омотача ротационог тела. Дат је аплет за детаљно испитивање тока и цртање графика функција, као и аплет за одређивања граничних вредности функција.

**Кључне речи:** GeoGebra, диференцијални рачун, граничне вредности функција, симболичка и нумеричка интеграција

Сања Томић, Мајда Смоле, студенти студијског програма Астрономија, Универзитет у Београду, Математички факултет

ОТКРИЋЕ ОСЦИЛАЦИЈЕ ЗВЕЗДЕ SIGMA CYGNI

**Апстракт:**

Мењање сјаја код звезда је познат феномен. Ми смо имале прилику да проучавамо временску серију радијалних брзина одређених из спектра звезде sigma Cygni и откријемо осцилацију која има период од 1.59h.





Академик Никола Хајдин, председник САНУ свечано отвара скуп



Проф. др Миодраг Матељевић, дописни члан САНУ и декан Математичког факултета поздравља учеснике и госте



Мр Александра Дреџун, директорка Центра за промоцију науке на свечаном отварању скупа



Излагање проф. др Бошка Јовановића, редовног професора и шефа Катедре за Нумеричку математику и оптимизацију Математичког факултета



Проф. др Надежда Пејовић, редовни професор Математичког факултета представља дигитални легат о Милутину Миланковићу



Проф. др Жарко Мијаиловић, редовни професор Математичког факултета представља књигу о Михаилу Петровићу





Учесници и гости на отварању скупа



Учесници и гости секције „Математика и примене-данас“



Детаљ са излагања проф. др Миодрага Матељевића



Учесници и председавајући секције  
„Математика и информатика у образовању“



Радна атмосфера на секцији „Математика и  
информатика у образовању“



Питања и дискусија



Учесници и гости



## Индекс аутора

- Ајдуковић, Никола 135  
Бањац, Бојан 135  
Баралић, Ђорђе  
Богдановић, Милена  
Божић, Милан 61  
Volzon, Gabriella  
Брашњевић, Марија 163  
Будински, Наталија 55  
Буљак, Владимир  
Васиљевић, Владица 163  
Гојић, Мирослав  
Грујић Ђајић, Александар 163  
Давидовић, Татјана 15  
Димитријевић, Гордана 135  
Дракулић, Дарко 27  
Ђенић, Александар 91  
Ђорђевић, Зоран 5  
Ђукић, Маја 127  
Јанковић, Оливера 111  
Јанчић Стојановић, Биљана 91,97  
Јевремовић, Весна  
Јеленковић, Наталија  
Јовановић, Александар 5  
Јовановић, Бошко С.  
Јововић, Ивана 163  
Јозић, Нивес 35  
Коледин, Тамара 163  
Кресоја, Милена 145  
Лазовић, Горан  
Maier, Giulio  
Малешевић, Бранко 135,163  
Малков Саша 1  
Маринковић, Богољуб  
Марић, Мирослав 91  
Матељевић, Миодраг 35  
Матић, Драган 61  
Матић, Љубица  
Миловановић, Градимир  
Мијајловић Жарко 1  
Михајловић, Радомир А.  
Михаљинац, Јасминка  
Митић Ненад 1  
Мишковић, Стефан 105  
Ненезић, Марија 163  
Николић, Вања 83  
Обадић, Богдан 154  
Обрадовић, Марко  
Обрадовић, Смиљана 135  
Огризовић, Петар 69  
Пејовић Надежда 1  
Пекић, Марина 97  
Перовић, Александар 5  
Петровић, Драган  
Радојевић, Слободан  
Рајачић, Јасна  
Ракић, Тијана 91, 97  
Рибич, Сања 97  
Росић, Александра  
Светлик, Марек 35  
Смоле, Мајда  
Спалевић, Невенка  
Спасојевић, Игор  
Станимировић, Зорица 91  
Станковић, Милош 117  
Такачи, Ђурђица  
Томић, Сања  
Hartmann, Hans  
Хаџи Пурић, Јелена  
Хедрих (Стефановић), Катица Р.  
Хоџић, Сандра 75  
Чампара, Милан 135  
Чукић, Милена