

MONOGRAFIA
POKONFERENCYJNA

SCIENCE,
RESEARCH, DEVELOPMENT #23

TECHNICS AND TECHNOLOGY.

Rotterdam (The Netherlands)

29.11.2019 - 30.11.2019

U.D.C. 004+62+54+66+082

B.B.C. 94

Z 40

Zbiór artykułów naukowych recenzowanych.

(1) Z 40 Zbiór artykułów naukowych z Konferencji Międzynarodowej Naukowo-Praktycznej (on-line) zorganizowanej dla pracowników naukowych uczelni, jednostek naukowo-badawczych oraz badawczych z państw obszaru byłego Związku Radzieckiego oraz byłej Jugosławii.

(30.11.2019) - Warszawa, 2019. - 96 str.

ISBN: 978-83-66401-25-9

Wydawca: Sp. z o.o. «Diamond trading tour»

Adres wydawcy i redakcji: 00-728 Warszawa, ul. S. Kierbedzia, 4 lok.103

e-mail: info@conferenc.pl

Wszelkie prawa autorskie zastrzeżone. Powielanie i kopiowanie materiałów bez zgody autora jest zakazane. Wszelkie prawa do artykułów z konferencji należą do ich autorów.

W artykułach naukowych zachowano oryginalną pisownię.

Wszystkie artykuły naukowe są recenzowane przez dwóch członków Komitetu Naukowego.

Wszelkie prawa, w tym do rozpowszechniania i powielania materiałów opublikowanych w formie elektronicznej w monografii należą Sp. z o.o. «Diamond trading tour».

W przypadku cytowań obowiązkowe jest odniesienie się do monografii.

Publikacja elektroniczna.

«Diamond trading tour» ©

Warszawa 2019

ISBN: 978-83-66401-25-9

Redaktor naukowy:

W. Okulicz-Kozaryn, dr. hab, MBA, Institute of Law, Administration and Economics of Pedagogical University of Cracow, Poland; The International Scientific Association of Economists and Jurists «Consilium», Switzerland.

KOMITET NAUKOWY:

W. Okulicz-Kozaryn (Przewodniczący), dr. hab, MBA, Institute of Law, Administration and Economics of Pedagogical University of Cracow, Poland; The International Scientific Association of Economists and Jurists «Consilium», Switzerland;

С. Беленцов, д.п.н., профессор, Юго-Западный государственный университет, Россия;

Z. Ćekerevac, Dr., full professor, «Union - Nikola Tesla» University Belgrade, Serbia;

Р. Латыпов, д.т.н., профессор, Московский государственный машиностроительный университет (МАМИ), Россия;

И. Лемешевский, д.э.н., профессор, Белорусский государственный университет, Беларусь;

Е. Чекунова, д.п.н., профессор, Южно-Российский институт-филиал Российской академии народного хозяйства и государственной службы, Россия.

KOMITET ORGANIZACYJNY:

A. Murza (Przewodniczący), MBA, Ukraina;

A. Горохов, к.т.н., доцент, Юго-Западный государственный университет, Россия;

A. Kasprzyk, Dr, PWSZ im. prof. S. Tarnowskiego w Tarnobrzegu, Polska;

A. Malovychko, dr, EU Business University, Berlin – London – Paris - Poznań, EU;

S. Seregina, independent trainer and consultant, Netherlands;

M. Stych, dr, Uniwersytet Pedagogiczny im. Komisji Edukacji Narodowej w Krakowie, Polska;

A. Tsimayeu, PhD, associate Professor, Belarusian State Agricultural Academy, Belarus.

I. Bulakh PhD of Architecture, Associate Professor Department of Design of the Architectural Environment, Kiev National University of Construction and Architecture

Recenzenci:

L. Nechaeva, PhD, Instytut PNPU im. K.D. Ushinskogo, Ukraina;

М. Ордынская, профессор, Южный федеральный университет, Россия.

ВЕБ-ДОДАТОК ДЛЯ ВИБОРУ JAVASCRIPT-БІБЛІОТЕК ВІЗУАЛІЗАЦІЇ ДАНИХ Криласов В.	6
ГЕНЕРАТОРИ ПСЕВДОВИПАДКОВИХ ЧИСЕЛ Шелемєтьєв Е.О.	11
РОЗРОБКА ВЕБ-ДОДАТКУ З ФУНКЦІЯМИ МЕСЕНДЖЕРА ТА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ УПРАВЛІННЯ ПРОЕКТАМИ Марченко А.	16
CRITICAL SECTIONS Hulliev N. B.	20
USING OF JUNIT FRAMEWORK FOR AUTOMATION UNIT TESTING IN THE CONTEXT OF TEST-DRIVEN DEVELOPMENT Abalmasova D.Y.	23
OPENMP PARALLEL PROCESSING Avdieiev O. D.	26
RXJS LIBRARY USAGE FOR REACTIVE PROGRAMMING IN JAVASCRIPT Mylokhin D.O.	28
THE DESCRIPTION OF “WEB PORTAL FOR INTERACTION OF DIFFERENT CATEGORIES OF PARTICIPANTS IN THE EDUCATIONAL PROCESS” FUNCTIONS Nesmiian D.	31
АНАЛІТИЧНИЙ ОГЛЯД МЕТОДІВ АУТЕНТИФІКАЦІЇ Тучкова М.С.	34
РОЗРОБКА ПРИСТРОЮ ЗБЕРІГАННЯ ПЕРСОНАЛЬНИХ ЕЛЕКТРОННИХ КЛЮЧІВ Тучкова М.С.	39
PROCESSING OF OATS INTO GROATS AND FLAKES Sots S., Kustov I., Kuzmenko Y.	44
GRAPHIC PROCESSORS PROGRAMMING WITH MC# LANGUAGE Teslenko D.	47
COMPUTER MODELING OF FINANCIAL MARKETS THROUGH CAPITAL ASSET PRICING MODEL Liutova K., Khovrat A.	51

PRINCIPLE OF WORKING OF MUSIC STREAMING SERVICES Tynianov A. D.	55
ARTIFICIAL DIETS OF THE BRACONIDAE FAMILY REPRESENTATIVES BY REPRODUCING IN VITRO METHOD Kimsanboyev X., Rustamov.A.	57
CHALLENGES OF PARALLEL PROGRAMMING Kyrylenko O.	60
LOGICS OF MULTI-STAGE DECISION-MAKING FOR ROBOTS Kuchkov B.	63
FORECASTING THE MARKET PRICE OF SOFTWARE USING NEURAL NETWORKS Tykhomyrov V.	67
HIGH LOCAL MICROWAVE DIAGNOSTICS OF MICROELECTRONICS MATERIALS USING SCANNING MICROWAVE MICROSCOPY Shcherban I.M., Slusarenko O.A.	71
РАЗРАБОТКА ШАГАЮЩЕЙ СЕЯЛКИ ДЛЯ КРУГОВЫХ ЛЕСНЫХ ПИТОМНИКОВ Бырдина С. С., Бырдин П. В.	75
APPROXIMATING OF SIGNALS. THE FINITE DIFFERENCE METHOD Saichyshyna N.	80
РОЛЬ ПРОСТОРУ У ВІРТУАЛЬНОЇ РЕАЛЬНОСТІ Храменок Л. О.	83
СИНТЕЗ И АНАЛИЗ КООРДИНАЦИОННЫХ СОЕДИНЕНИЙ ИОДИДОВ НИКЕЛЯ (II), ЦИНКА И КАДМИЯ С ИЗОНИКОТИНОИЛГИДРАЗОНОМ ПАРА-АЦЕТИЛАМИНО БЕНЗАЛЬДЕГИДОМ (ИНГПААБА-L4) Чантурия М.М., Цинцадзе М.Г., Гулбани Д.В.	87
A DESIGN OF A NEURAL NETWORK FOR IMAGES CLASSIFICATION USING MATLAB AND JAVA Dudka. D. A.	92

ВЕБ-ДОДАТОК ДЛЯ ВИБОРУ JAVASCRIPT-БІБЛІОТЕК ВІЗУАЛІЗАЦІЇ ДАНИХ

Владислав Криласов

магістрант,

фізико-математичний факультет, Центральноукраїнський державний педагогічний університет імені Володимира Винниченка

Ключові слова: веб-додаток, javascript-бібліотека, візуалізація даних.

Keywords: web application, javascript library, data visualization.

Сучасний етап розвитку інформаційних технологій характеризується повсякчасним та повсемісним використанням візуалізованих даних. Стрімке проникнення інформаційних технологій в більшість сфер життєдіяльності людей та перенавантаження інформаційними потоками вимагають від сучасного суспільства розробки нових технологій, зміни способів подання інформації і даних в візуально-доступному для сприйняття вигляді.

Доцільність використання візуалізації зумовлена необхідністю компактного подання даних у вигляді, який буде найбільш зручним для їх сприйняття, розуміння, засвоєння і запам'ятовування.

Візуалізація даних є найшвидшим способом донести її до інших. Більш буквально, візуалізація – це процес відображення інформації у візуальній формі. При цьому ми створюємо правила, інтерпретуємо дані і визначаємо основні параметри в якості візуальних властивостей [7].

Інструментарій візуалізації даних дуже різноманітний. Подамо короткий перелік інструментів, придатних для обробки та візуалізації даних:

- **електронні таблиці** – LibreOffice, Excel чи Google Docs;
- **середовище для статистичного програмування** – R (r-project.org) чи Pandas (pandas.pydata.org);
- **геоінформаційні системи** (GIS) – Quantum GIS, ArcGIS, GRASS;
- **бібліотеки візуалізацій** – d3.js (github.com/d3), Prefuse (prefuse.org), Flare (flare.prefuse.org);
- **інструменти для обробки даних** – Google Refine, Datawrangler;
- **середовище для візуалізацій без використання програмування** – ManyEyes, Tableau Public [7].

Кількість доступних на сьогодні друкованих та електронних джерел, щодо інструментів візуалізації наочно демонструє, що інтерес до питання візуалізації даних утримується на досить високому рівні. Об'єктом нашого дослідження є JavaScript-бібліотеки візуалізації даних.

Ознайомлення з існуючими джерелами ми проводили за такими напрямками. По-перше різноманітні огляди і рейтинги найбільш популярних JavaScript-бібліотек візуалізації даних. Наприклад: 14 Popular JavaScript Libraries for Data Visualization in 2019

Schemes Generator

Вітаємо!
Генератор схем дозволить Вам за 3 простих кроки створити свою унікальну інтерактивну схему!

1 Оберіть назву для схеми

Назва схеми

Підготуйте CSV файл із даними для схеми за шаблоном. Файл може складатися із обов'язкових (name, description) та опціональних полів (link, filter1, filter2...). Поля-фільтри можуть мати декілька значень які розділяються комами.

ОБРАТИ ФАЙЛ З ДАНИМИ

	A	B	C	D	E	F
1	name	description	link	vendor	license	type
2	Windows	Windows		Microsoft	proprietary	desktop,mobile,server
3	Mac OS	Mac OS		Apple	proprietary	desktop
4	Linux	Linux		Community	free	desktop,mobile,server

Для візуалізації сутностей додайте зображення-іконку для кожної з них. Кількість зображень повинна співпадати із кількістю сутностей. Зображення співставляються до сутностей за іменем. Ім'я кожного зображення - це порядковий номер сутності. Бажано, щоб зображення були квадратної форми та відносно невеликого розміру (до 100x100 пікселів). Більші розширення будуть стиснуті.

ОБРАТИ ЗОБРАЖЕННЯ

Упорядкування: Палка

Бібліотека "Зображення"

Створити папку

01.jpg
02.jpg
03.jpg
04.jpg
05.jpg

Ім'я файлу: '05.jpg' '01.jpg' '02.jpg' '03.jpg'

Натисніть файл ("ctrl+glt")

Відкрити Скасувати

Рисунок 1. Головна сторінка веб-додатку «Schemes Generator»

[5], 11 Javascript Data Visualization Libraries for 2019 [4], 10 лучших JavaScript библиотек для визуализации данных на графиках и диаграммах [1].

Також, ми виокремили й дослідили джерела, які містять каталоги візуалізації даних. Найбільш повний перелік доступних каталогів візуалізації даних подано у статті [3]. Каталоги зазвичай містять інформацію про тип візуалізації, її опис, а також вказується перелік інструментів, якими її можна реалізувати.

Ще одне джерело, яке заслуговує уваги – сторінка Вікіпедії, що містить таблицю порівняння JavaScript-бібліотек візуалізації даних [6]. В таблиці наведено порівняння більшості існуючих JavaScript-бібліотек за такими параметрами: ліцензія, підтримувані типи діаграм, інтерактивність,

технологія рендерингу, прив'язка даних.

Провівши аналіз доступних матеріалів ми встановили, що у більшості джерел звертають увагу в основному на загальний опис та перелік підтримуваних типів візуального подання. Наявність великої кількості JavaScript-бібліотек візуалізації даних приводить до труднощів вибору розробником потрібного інструменту візуалізації. Тому ми вирішили створити власний інструмент для вибору JavaScript-бібліотек візуалізації даних.

Такий інструмент вирішено було створити у вигляді веб-додатку, що дозволить обирати бібліотеки, не лише аналізуючи кількість підтримуваних типів візуалізації, а й з врахуванням інших важливих для розробників програмного забезпечення параметрів. До

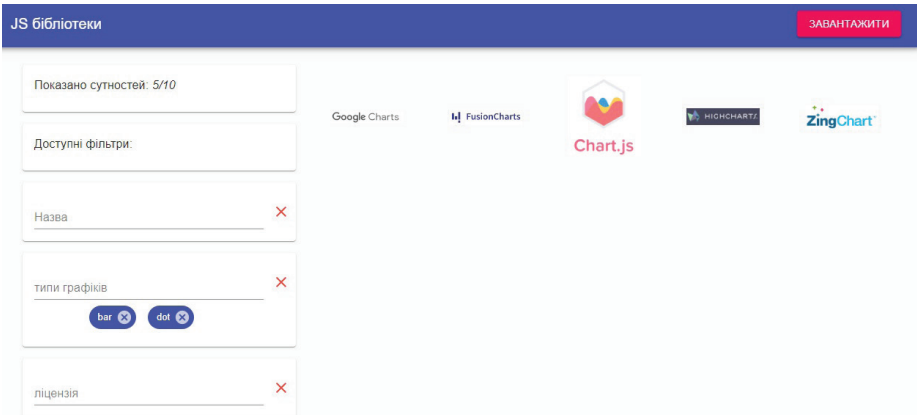


Рисунок 2. Схема згенерована додатком «Schemes Generator»

таких параметрів ми віднесли: ліцензію, інтеграцію, технологію рендерингу, формати імпорту та експорту даних, інтерактивність та анімацію, особливості технічної підтримки, присутність підтримки картографії. Також було вирішено додати короткий опис бібліотеки та посилання на офіційний сайт чи інший подібний ресурс.

Під час роботи над додатком виникла ідея, розробити універсальний веб-додаток для візуалізації певної предметної області за різними категоріями, параметрами, а вже потім засобами створеного додатку реалізувати вибір JavaScript-бібліотек візуалізації даних. Отже, було спроектовано та реалізовано веб-додаток «Schemes Generator». Додаток розташований за тимчасовою адресою <https://schemes-generator.herokuapp.com>, а його зовнішній вигляд показано на рисунку 1.

Головна сторінка додатку «Schemes Generator» містить інструкцію для користувача та майстер створення схеми. Як видно з наведеної інструкції

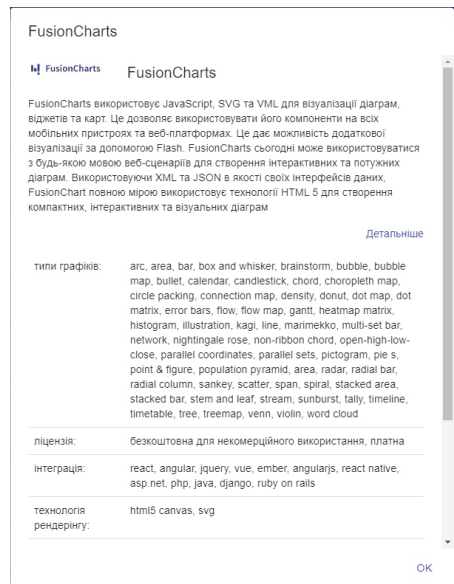


Рисунок 3. Вікно з описом бібліотеки FusionCharts

для створення схеми потрібно провести попередню підготовку. Імпорт даних в додаток відбувається в форматі csv. Для зручності користувачів ми описали обов'язкові та необов'язкові поля, які він може містити, а також

надали користувачеві шаблон такого csv-файлу. Наступним обов'язковим кроком є підготовка зображень для кожної із сутностей. Зображення необхідно пронумерувати відповідно до переліку сутностей в csv-файлі.

Після того, як csv-файл з сутностями та відповідні зображення готові, можна приступати до створення схеми: ввести назву, завантажити csv-файл та групу зображень. Якщо всі складові завантажено стане активною кнопка «Згенерувати». Після її натиснення розпочнеться процес створення нової схеми. Якщо файли підготовлені правильно, то в новій вкладці відкриється новостворена схема (рис. 2). Зліва вона міститиме перелік фільтрів. Інша частина сторінки показуватиме перелік JavaScript-бібліотек візуалізації даних відібраних згідно з фільтрами. Бібліотеки показуються у вигляді зображень з гіперпосиланням, при натисненні на яке відкривається вікно з описом JavaScript-бібліотеки та її властивостями. Також це вікно містить гіперпосилання на веб-ресурс, якщо це було передбачено csv-файлом (рис. 3).

Вгорі сторінки наявна кнопка «Завантажити» (рис. 2), яка дозволяє зберегти створену схему у вигляді zip-архіву з html-сторінкою та супутніми файлами. Збережену веб-сторінку можна використовувати локально або ж розташувати на будь-якому хостингу.

Таким чином, у результаті дослідження проведено огляд сучасних JavaScript-бібліотек візуалізації даних та виділено ключові характеристики,

які повинен враховувати розробник при виборі JavaScript-бібліотек. Для вибору JavaScript-бібліотек реалізовано універсальний веб-додаток «Schemes Generator» і за його допомогою створено схему «JS бібліотеки». Використовуючи фільтри, які автоматично створюються на основі полів csv-файлу, розробник може сформувати список JavaScript-бібліотек та отримати детальну інформацію про кожну з них, а також перейти на офіційний сайт для їх скачування. Створена схема доступна онлайн [2]. Також її можна зберегти у вигляді html-сторінки. В подальшому планується розширити перелік JavaScript-бібліотек візуалізації даних та урізноманітнити фільтри.

Бібліографія

1. 10 лучших JavaScript библиотек для визуализации данных на графиках и диаграммах: [Электронный ресурс] // TechMedia Издательство интернет-проектов, 2006-2019 – Режим доступа: <https://habr.com/ru/post/457946/> (Дата звернення: 25.05.2019)
2. Веб додаток «Schemes Generator»: [Электронный ресурс]. Криласов Влад, 2018-2019. – Режим доступа: <https://schemes-generator.herokuapp.com/schemes/71302e41?fbclid=IwAR0MGjAHliurWx-Wcq9GlsGwak7iT4t4sReXCG9DKh3BE-G-T6tNGh1r64X4> Дата звернення: 25.11.2019)
3. Освіта: Каталоги візуалізації даних: [Электронный ресурс] // Текстура. Блог про візуалізацію даних – Режим доступа: <https://textura.in.ua/posts/ed/dataviz-catalogue/> (Дата звернення: 12.04.2019)
4. Javascript Data Visualization Libraries for 2019: [Электронный ресурс] // A Medium Corporation – Режим доступа: <https://blog.bitsrc.io/11-javascript-charts->

- and-data-visualization-libraries-for-2018-f01a283a5727 Дата звернення: 28.05.2019)
5. Popular JavaScript Libraries for Data Visualization in 2019: [Електронний ресурс] // Monerail sp. z o.o. – Режим доступу: <https://www.monerail.com/blog/javascript-libraries-data-visualization> (Дата звернення: 15.06.2019)
 6. Comparison of JavaScript charting libraries: [Електронний ресурс] // Wikipedia, the free encyclopedia – Режим доступу: https://en.wikipedia.org/wiki/Comparison_of_JavaScript_charting_libraries (Дата звернення: 27.05.2019)
 7. Scott Murray Interactive Data Visualization for the Web / Murray Scott – Sebastopol, O'Reilly Media, 2013. – 268 p.

ГЕНЕРАТОРИ ПСЕВДОВИПАДКОВИХ ЧИСЕЛ

Шелемєтьєв Е.О.

студент групи ПЗПІ-18-6

Харківський національний університет радіоелектроніки

Анотація: Доповідь присвячена огляду найвідоміших алгоритмів генерації псевдовипадкових чисел, їх сильним та слабким сторонам. Проведене дослідження дозволяє встановити критерії оцінювання генераторів при їх виборі для застосування на практиці. Були перелічені статистичні тести для оцінки різноманітних генераторів псевдовипадкових чисел.

Ключові слова: генератор псевдовипадкових чисел; метод середини квадратів; лінійний конгруентний метод; реєстр зсуву з лінійним зворотним зв'язком; критерій NIST.

У сучасному світі дуже важливо передбачати тенденцію числових випадкових даних, але також існує велика кількість задач, які вимагають створення, або генерації, випадкових величин: від лотерей та відеоігор до шифрування даних та статистичного моделювання.

Здається, що тут може бути складного? Спробуйте самі назвати десять випадкових чисел чи навіть сто. Для пересічної людини – це є складною задачею, тому це завдання доручають електронним пристроям.

Перед тим, як продовжити, давайте згадаємо, яку величину ми називаємо випадковою. За класичним визначенням випадковою називають величину, котра в результаті випробування прийме одне і тільки одне можливе значення, наперед невідоме і залежне від випадкових причин, які заздалегідь не можуть бути враховані [1]. Чи може людина або арифметико-логічний пристрій генерувати насправді випадкові числа? Очевидно, що ні, бо вони виробляють величини у відповідності до своїх вподобань, настрою, алгоритмів чи попередніх згенерованих значень. Тобто в даному випадку наявні причини, які можуть бути враховані. Людина чи арифметичний пристрій можуть лише імітувати генерацію незалежних та рівномірно розподілених величин, через що їх можна назвати генераторами псевдовипадкових чисел (ГПВЧ) [2].

Ще в 1949 Джон фон Нейман, батько архітектури сучасних комп'ютерів, запропонував дуже простий алгоритм генерації псевдовипадкових чисел – *метод середини квадратів*. Він формулюється так: беремо початкове число, скажімо, чотирьох розрядне, підносимо його до квадрату, беремо з середини чотири розряди, які стають новим початковим числом. Наприклад:

5678 \rightarrow 32239684 \rightarrow 2396 \rightarrow 05740816 \rightarrow 7408 \rightarrow ...

На жаль, даний алгоритм має багато недоліків: в залежності від початкового значення можна дуже швидко досягти циклічного повтору чисел; в загальному випадку алгоритм покриває тільки 60% усіх значень [3].

Метод середини квадратів на практиці може використовуватися як хеш-функція. Також варто зазначити, що в ЕОМ алгоритм обчислюється на основі двійкової системи числення за допомогою арифметичних операцій множення та зсуву [4].

Найпростішим алгоритмом ГПВЧ, який використовується у більшості програмних додатків, є *лінійний конгруентний метод*, що має наступний загальний вигляд:

$$X_{n+1} = a * X_n + b \text{ mod } m , \quad (\text{формула 1})$$

де a – множник, b – приріст, m – модуль, X_0 – заздалегідь задане (початкове) число (seed), mod – операція отримання залишку від ділення, числа X_1, X_2, \dots, X_n утворюють послідовність згенерованих значень. Числа a, b, m та X_0 називають параметрами генератора [5].

Наприклад, для $a = 13, X_0 = 1, b = 3, m = 16$ маємо наступну послідовність згенерованих значень: 0, 3, 10, 5, 4, 7, 14, 9, 8, 11, 2, 13, 12, 15, 6, 1, 0, 3 Як можна побачити, рівномірно розподілена послідовність має період 16 чисел (m). Тобто після того, як хоча б одне згенероване число повториться, послідовність також почне повторюватися. Для заданого m цей період є найбільшим можливим.

Маємо зовсім протилежну ситуацію, якщо змінити значення a на 2. Отримаємо сукупність чисел така: 5, 13, 13, Маємо усього два унікальних значення.

Для того, щоб в лінійному конгруентному методі досягти значення періоду, рівному m необхідне виконання умов наступної теореми [6]:

a та m є взаємно простими;

a рівне (конгруентне) за модулем p одиниці, для будь-якого p , що є простим множником m ;

a рівне (конгруентне) за модулем 4 одиниці, якщо 4 є дільником m .

Даний алгоритм є дуже швидким і використовується для моделювання випадкових значень в статистиці, але є надзвичайно передбачуваним, що не дозволяє його використання в криптографії [7].

Більш безпечним є *регістр зсуву з лінійним зворотним зв'язком*, який працює на основі вираховування результуючого біту (біт зворотнього зв'язку) у відповідності до лінійної булевої функції від значень усіх або декількох бітів регістра, що називається функцією зворотнього зв'язку [8].

Нехай задано регістр з довжиною p , тобто регістр має p комірок, проіндексованих від 0 до $p - 1$, коефіцієнти a_1, a_2, \dots, a_{p-1} , що приймають значення 0 та 1, які визначають рекурентну формулу:

$$X_{n+p} = a_{p-1} * X_{n+p-1} \oplus a_{p-2} * X_{n+p-2} \oplus \dots \oplus a_1 * X_{n+1} \oplus X_n , \quad (\text{формула 2})$$

де X_i – вихідна послідовність бітів, \oplus – булева операція суми по модулю 2 (XOR).

Сам алгоритм виглядає наступним чином:

Вміст комірки $p - 1$ формує черговий біт вихідної послідовності.

Вираховується значення функції зворотнього зв'язку за формулою (2).

Зміст кожної i -ої комірки переміщується в наступну комірку з індексом $i + 1$ операція бігового зсуву).

В комірку з індексом 0 заноситься значення, вираховане на кроці 2.

Найбільший можливий період вихідної послідовності (M -послідовності) дорівнює $2^p - 1$.

На практиці регістр зсуву з лінійним зворотним зв'язком найефективніше використовувати на апаратному рівні, але існують досить продуктивні його реалізації на низькорівневих мовах програмування [9].

Усі вищеперелічені генератори псевдовипадкових чисел є основою для інших генераторів, які включають їх різноманітні комбінації та додаткові операції над вихідними послідовностями (генератор «стоп-пішов», вихор Мерсенна, XPD/KPD, A5, проріджуючий генератор, каскад Голмана, генератор Гефа та інші).

Але як обрати генератор псевдовипадкових чисел? Потрібно мати на увазі безліч властивостей: період, рівномірність розподілу величин, незалежність нових значень від попередніх та ефективність програмної або апаратної реалізації. Щоб перевірити генератор на всі ці параметри існують безліч тестів, які можна розділити на дві групи: емпіричні та теоретичні. Якщо останні вимагають знання лише алгоритму роботи генератора, то емпіричні тести вираховують його якість тільки використовуючи вихідну послідовність псевдовипадкових значень.

Найбільш використовуваними емпіричними тестами є критерій χ^2 -квadrat (χ^2 -критерій) та критерій Колмогорова-Смирнова [10]. Зупинимося на першому.

Перевірка критерію χ^2 полягає у виконанні наступних кроків:

Отримуємо достатню кількість згенерованих значень від a до b .

Підраховуємо кількість значень n_j , що потрапили в кожен з k інтервалів $(a_j, b_j]$ проміжку $(a, b]$.

Підраховуємо статистику за формулою:

$$\chi^2 = \sum_{j=1}^k \frac{(n_j - E_j)^2}{E_j}, \quad (\text{формула 3})$$

де E_j – очікуване число потраплянь в j -ий інтервал.

Визначаємо, чи знаходиться вираховане значення в довірчому інтегралі, за спеціальною таблицею [8].

Варто мати на увазі, що χ^2 -критерій є основою для багатьох емпіричних тестів і є найбільш часто використовуваним для оцінки генераторів псевдовипадкових чисел.

Статистичний критерій NIST – це програмний пакет, розроблений Лабораторією інформаційних технологій, який включає 15 тестів, призначених для пере-

вірки двійкових послідовностей, згенерованих за допомогою генераторів псевдовипадкових чисел [11]. Розглянемо кожен з тестів окремо.

Frequency (Monobits) Test (частотний побітовий тест) – виявляє дисбаланс у кількості нулів та одиниць послідовності.

Test For Frequency Within A Block (блочний тест на частоту) – ділить послідовність на блоки та виявляє дисбаланс у кількості нулів та одиниць.

Runs Test (тест на серійність) – неперервні серії однакових бітів порівнюються з очікуваним розподілом для випадкових послідовностей.

Test For The Longest Run Of Ones In A Block (тест на найдовшу серійність) – аналогічно попередньому, але порівнюються найдовші послідовності однакових бітів.

Random Binary Matrix Rank Test (матрично-ранговий тест) – розрахунок рангів підматриць, побудованих на основі вхідної послідовності.

Discrete Fourier Transform (Spectral) Test (спектральний тест) – виявляє повторювані блоки чи послідовності.

Non-Overlapping (Aperiodic) Template Matching Test (тест з неперіодичними шаблонами, що не перетинаються) – показує кількість заздалегідь визначених бітових шаблонів в послідовності.

Overlapping (Periodic) Template Matching Test (тест на періодичні шаблони, що перетинаються) – аналогічно попередньому, але тепер шаблони шукаються і в попередньо знайдених бітових шаблонах.

Maurer's Universal Statistical Test (універсальний статистичний тест) – показує кількість біт між двома шаблонами.

Linear Complexity Test (тест лінійної складності) – оцінює довжину регістр зсуву з лінійним зворотним зв'язком.

Serial Test (тест на періодичність) – підрахунок усіх можливих перетинів шаблонів заданої довжини на вхідній послідовності бітів.

Approximate Entropy Test (тест приблизності ентропії) – порівнює частоти перекриття двох послідовних блоків вхідної послідовності з частотами перекриття аналогічних блоків у випадковій послідовності.

Cumulative Sum (Cusum) Test (тест кумулятивних сум) – перевірка на максимальне відхилення від нуля кумулятивних сум вхідних цифр послідовності.

Random Excursions Test (тест на довільні відхилення) – підрахунок кількості циклів, що мають k заходів при довільному обході кумулятивної суми.

Random Excursions Variant Test (альтернативний тест на довільні відхилення) – підрахунок відхилень від очікуваного числа при заході в довільному обході кумулятивної суми.

Можна задатися питанням: чому існує так багато тестів? Але відповідь досить проста – усі генератори псевдовипадкових чисел мають свої слабкості та сильні сторони, які один тест не може ніяк охопити або навіть точно сказати про

те, наскільки гарним є даний генератор взагалі. Наприклад, в нас є послідовність чисел: 010101...010101, яка ідеально пройде частотний тест, але повністю провалить серійний тест [10].

Генератори псевдовипадкових чисел – дуже важлива складова нашого життя, яка зазнає активного розвитку останні півсторіччя. На неї покладаються криптографія, статистика, програмування та теорія ігор. Тому необхідно гарантувати існування надійних джерел випадкових чисел, які зможуть задовольнити потреби людей.

Список літератури

1. Гмурман В.Е. Теория вероятностей и математическая статистика: Учеб. пособие для вузов. М.: 2003. – 479 с.
2. Random Number Generation: [Електронний ресурс] ResearchGate. Режим доступу: <https://www.researchgate.net/publication/>
3. Random_Number_Generation (Дата звернення: 16.11.2019).
4. The Middle Square Method (Generating Random Sequences VIII): [Електронний ресурс] Harder, Better, Faster, Stronger. Режим доступу: <https://hbfs.wordpress.com/2017/11/21/the-middle-square-method-generating-random-sequences-viii/> (Дата звернення: 16.11.2019).
5. Хеширование: [Електронний ресурс] RSDN. Режим доступу: <https://rsdn.org/article/alg/bintree/hash.xml#ETF> (Дата звернення 16.11.2019).
6. Chung-Chih Li, Bo Sun. Using Linear Congruential Generators for Cryptographic Purposes: [Електронний ресурс]. Режим доступу: http://zaic101.ru/files/728/using_linear_congruential_generators_for_cryptographic_purposes.pdf (Дата звернення: 16.11.2019).
7. Hull T.E. and Dobell A.R. Random Number Generators: [Електронний ресурс]. Режим доступу: https://dspace.library.uvic.ca:8443/bitstream/handle/3142/Random_Number_Generators.pdf?sequence (Дата звернення: 17.11.2019).
8. James Roper. Cracking Random Number Generators – Part 1: [Електронний ресурс]. all that jazz. Режим доступу: https://jazzy.id.au/2010/09/20/cracking_random_number_generators_part_1.html (Дата звернення 17.11.2019).
10. Слеповичев И.И. Генераторы псевдослучайных чисел: [Електронний ресурс]. Режим доступу: https://www.sgu.ru/sites/default/files/textdocsfiles/2018/09/slepovichev_i.i._generatory_psevdosluchaynyh_chisel_2017.pdf (Дата звернення 17.11.2019).
12. Шустрый 128-битный LFSR (MMX required): [Електронний ресурс]. habr. Режим доступу: <https://habr.com/ru/post/140095/> (Дата звернення: 17.11.2019).
13. Dan Viebighauser. Testing Random Number Generators: [Електронний ресурс]. Режим доступу: <http://www-users.math.umn.edu/~garrett/students/reu/prNGs.pdf> (Дата звернення: 17.11.2019).
15. Random Bit Generation: [Електронний ресурс]. NIST. Режим доступу: <https://csrc.nist.gov/Projects/Random-Bit-Generation/Documentation-and-Software/Guide-to-the-Statistical-Tests> (Дата звернення: 17.11.2019).

РОЗРОБКА ВЕБ-ДОДАТКУ З ФУНКЦІЯМИ МЕСЕНДЖЕРА ТА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ УПРАВЛІННЯ ПРОЕКТАМИ

Марченко Альона

магістрант, фізико-математичний факультет,
Центральноукраїнський державний педагогічний університет
імені Володимира Винниченка

Ключові слова: веб-додаток, месенджер, управління проектами.

Keywords: web application, messenger, project management.

У сучасному світі все більшої ваги набуває програмне забезпечення. Відповідно і розробка програмного забезпечення виділилася в окрему галузь економіки провідних держав світу. Показники цієї галузі мають позитивну динаміку. Економічні дані про розвиток компаній малого та середнього бізнесу в сфері розробки програмного забезпечення в Україні підтверджують це. Досягнення галузі ІТ багато в чому завдячують новизні ринку розробки програмного забезпечення.

Активний розвиток галузі розробки програмного забезпечення призводить до наявності певних недоліків, найбільш частими із яких є:

- недотримання розробниками планів і графіків робіт;
- перевищення розмірів встановлених бюджетних обмежень;
- недосконала робота створюваного програмного продукту;
- невдало організований менеджмент розробки програмного забезпечення [2].

На думку авторів [3] специфічними для галузі розробки на замовлення програмного забезпечення, є такі по-

милки: помилки в оцінці трудомісткості і термінів виконання робіт, відсутність або неповнота вимог замовника, зміна вимог на всіх етапах розробки, невисока якість розробленого продукту, втрата компанією-розробником накопиченої експертизи, проблеми управління розробкою.

Одним із шляхів розв'язування проблем пов'язаних з управлінням розробкою є використання різноманітних сервісів task- і тайм-менеджменту, для управління і планування проектів, командної роботи, побудови онлайн діаграм і т.п. [1]. Наприклад, першочергові задачі таких сервісів: швидко призначити завдання виконавцям, планувати спринти, трекати виконання і статуси, візуалізувати процеси і ділитися результатами з замовниками.

Провівши огляд сучасних джерел ми склали цілий список різноманітних сервісів, які б дозволили працювати над проектами в кілька разів ефективніше. Повний список сервісів (а їх було близько 50) наводити не будемо, вкажемо лише основні з них:

Jira – потужний онлайн-сервіс, що дозволяє командам-розробникам пла-

The screenshot displays two administrative panels. The top panel, titled 'Users', shows a list of three users: an Administrator with 2 posts, and two Authors with 0 posts. The bottom panel, titled 'Project', shows two published projects: 'Разработка туристического сайта' and 'Разработка Промо сайту', both published recently.

Username	Name	Email	Role	Posts
[Redacted]	—	[Redacted]@gmail.com	Administrator	2
mar4enko	Alena Mar4enko	mar4enko.lena0522@gmail.com	Author	0
[Redacted]	—	[Redacted]@gmail.com	Author	0

Title	Date
Разработка туристического сайта	Published 33 mins ago
Разработка Промо сайту	Published 2019/10/29
Title	Date

Рисунок 1. Створення користувачів і проектів

нувати проекти, призначати виконавців завдань, планувати спринти і збирати завдання в беклог, виставляти пріоритети і дедлайни. В цілому дуже гарний сервіс, але він виявився занадто складним і дуже багатофункціональним, наприклад має дуже велику кількість налаштувань, дуже важко і довго доводиться в них розбиратися, часто важко просто знайти відповідь на питання, що цікавить, також немає можливості призначення кількох виконавців для одного завдання

GanttPro – діаграма Ганта, що дозволяє планувати і управляти проектами онлайн, візуалізувати процеси, створювати завдання і призначати їх учасникам, виставляти дедлайни і відсоток завершення окремих завдань і проекту в цілому, додавати віхи, ділитися створеним графіком Ганта з ко-

мандою і з клієнтами з правом перегляду та експортувати його. Серед мінусів виділили відсутність мобільної версії і мобільного додатку, а також відсутність кастомізувати звіти.

Basecamp – один з найбільш популярних онлайн-сервісів для спільної роботи над проектами, що дозволяє ділитися документами, вести обговорення з командою, створювати to-do листи і додавати коментарі до завдань, вислати і приймати електронну пошту. Але відсутні можливості для тактичного і стратегічного планування, неможливо оцінити повну картину по проекту.

Trello – популярний онлайн-сервіс, що дозволяє систематизувати ваші завдання, to-do листи, обговорення та ідеї на одній дошці. Серед основних недоліків відсутність можливості спілкування.

Список запитів

На перевірці
 Тестування
 Відкрито
 Очікування
 Зроблено

Новий запит

Назва

Message

Створити

Виправити помилку у відображенні нижньої навігаційної панелі #51

Відкрито - 30.11.2019 17:24
Оновлено - 30.11.2019 17:24
Клік повідомлень - 1

При зменшенні розмірів вікна виникає помилка у відображенні нижнього меню - воно виходить за межі відповідної області
Send - 30.11.2019 17:24

Додати повідомлення

Додати

Змінити статус

Відкрито

На перевірці

Тестування

Відкрито

Очікування

Зроблено

Додати перевірку правильності введення товарів #50

Відкрито - 21.11.2019 6:52
Оновлено - 30.11.2019 17:27
Клік повідомлень - 4

Потрбно додати перевірку правильності введення товарів від виробників з бази даних нашого підприємства
Send - 21.11.2019 6:52

надайте список товарів який ви вводите
Send - 30.11.2019 17:24

У мене при введенні все працює, уточніть що саме не так!
Send - 30.11.2019 17:25

Коли вводиш товари через кнопку импорт не всі товари завантажуються. Зазвичай десь 10 -20 товарів втрачаються із 100
Send - 30.11.2019 17:27

Рисунок 2. Робоча область проекту

Slack – web-сервіс для створення чатів і окремих груп для ведення обговорень з командою, клієнтами або користувачами. Серед недоліків виділимо такі: основні особливості розкриті і захвані в дроп-даунах, немає можливості кастомізувати інтерфейс, відсутня можливість відключити повідомлення.

Відмітимо також важливу особливість, що для потреб які постають перед розробниками програмного забезпечення, немає єдиного сервісу, необхідно суміщувати декілька з них в залежності від призначення – месенджери, управління проектами, списки задач тощо.

Саме тому ми вирішили створити сервіс, який би суміщував функції месенджера та програм управління проектами.

Спроектований сервіс має клієнт-серверну архітектуру. В якості сервера виступає веб-сервер Nginx з встановленою CMS WordPress. Для баз даних задіяно MySQL-сервер. Клієнтська частина FrontEnd реалізована засобами фреймворку Angular. Додаток передбачає участь трьох типів користувачів: адміністратора (менеджер проектів, керівник), клієнтів (замовників) і виконавців (дизайнер, розробник, тестувальник). Доступ до

адміністративної частини має лише адміністратор.

Основна задача адміністратора створення користувачів та проектів (рис. 1). Також при створенні нового або при редагуванні існуючого проекту адміністратор вмикає видимість і доступ до проекту.

Робота з веб-додатком клієнтів і виконавців розпочинається з авторизації, після якої стане доступним список проектів на які їх призначено. Після вибору проекту відкриється вікно що містить опис проекту, фільтр запитів, область запитів. В області запитів, показуються існуючі запити, відповідно до ввімкненого фільтру та є можливість створити новий запит. Зовнішній вигляд робочої області проекту показано на рисунку 2.

Запити складаються з заголовку виділеного певним кольором та області повідомлень. Кожен запит має динамічний статус. Можливі статуси показані в випадяючому списку (рис 2.). Розкриємо значення статусів: відкрито – новий запит; тестування – встановлюється клієнтом коли він тестує зроблену роботу; на перевірці – встановлює виконавець, про те що треба провести перевірку; очікування – якщо все перевірено, але потрібне доопрацювання чи наприклад очікується оплата; зроблено – запит закрито, завдання виконане. Встановлення статусу запиту впливає на його

відображення відповідно до налаштувань фільтру.

Отже, у результаті дослідження проведено огляд сучасних сервісів, для підвищення ефективності процесу розробки програмного забезпечення. Встановлено, що не має єдиного сервісу, який може забезпечити потреби в управлінні проектами і месенджера. Тому було реалізовано універсальний клієнт-серверний веб-додаток який забезпечив комунікацію між менеджером, клієнтом і розробником, дозволив зберігати всі обговорення щодо проекту в одному місці. Також до додатку додані деякі функції управління проектами. Кожен запит окрім обговорення містить статус, що дозволяє управляти запитами за допомогою фільтрів. Подальші дослідження будуть спрямовані на вдосконалення запитів, додавання файлів до повідомлень, зв'язок з іншими месенджерами.

Бібліографія

1. 43 полезных сервиса для управления проектами. Без эпитетов: [Електронний ресурс] // – TechMedia Издательство интернет-проектов, 2006-2019 – Режим доступу: <https://habr.com/ru/post/276873/> (Дата звернення: 25.09.2019)
2. Менеджмент разработки программного обеспечения: учеб. пособие / А. В. Благов. – Самара: Изд-во Самар. гос. аэрокосм. ун-та, 2014. – 84 с.
3. Управління IT-проектами. [Книга 1. Стандарти, моделі та методи управління проектами]: [підручник]. – Львів: «Новий Світ-2000», 2013. – 550 с.

CRITICAL SECTIONS

Huliiev N. B.

Scientific leader – senior lecturer of Software Engineering department Oliinik O.V. Kharkiv National University of Radioelectronics (61166, Kharkiv, Lenin Avenue, 14, faculty of computer science, e-mail: nural.huliiev@nure.ua

Keywords: critical sections, parallel programming, blocking.

Consider using the simplest idea to solve a competition problem. It is easy to see as a source of our mistake that the simplest money-transferring operation actually breaks down into several operations, and there is always a chance of interference between them of some other flow. In this case, it is said that the original operation is not atomic.

Now, when two threads are going to execute the code of the critical section at the same time, the one that started the first one executes all of its code completely before the second starts its execution (the second thread will wait until the first one completes the code of the critical section).

Consider the properties that a critical section should have.

- Mutual exclusion: A critical section code can only execute one thread at a time.

- Progress: If multiple threads are simultaneously requested to log in to a critical section, then one of them must be logged in (they cannot all block each other).

- Restrictions on Standby: The process of trying to enter a critical section will sooner or later be mandatory.

Only one process can be in a critical area.

Active standby mutual exclusion:

- Interruption prohibition (CLI, STI, non-masked interrupt running);

- Lock variables. If the processes are the same, then they can cooperate. If different, then we may have problems called spin blocking (it all depends on speed);

- Strict duty. To collaborate create a method of strict alternation. Strict rotation involves not only zero-to-zero checks, but also checking for an additional variable that determines whose queue;

- Peterson algorithm. Allows you to set strict alternation for n-processes. Implementation: If a process is in a critical area, it blocks access to the critical area and sets all other processes to a value – may enter. When leaving the critical area, the process sets values – they cannot enter;

- TSL command. The essence of the work is the following: read from a variable value and enter it in the register, with the variable set 1.

In all variants of implementation of mutual exclusion, when expected, checks are performed, which means that CPU time is consumed. Therefore, this expectation is called active.

Consider the work of processes. Generalizing the case of n processes requires

n variables. However, there are only two states of interest to us: either a particular process is in its critical section or none is present. Regardless of the number of processes, in order to distinguish between these two states, one variable is sufficient.

Let lock be a logical variable that indicates whether a process is in a critical section, ie `lock == (in1 || in2)`. Using lock instead of in1 and in2, you can implement logon and logout protocols as follows:

```
# critical sections based on blocking
bool lock = false;
process CS1 {
    while (true) {
        await (! Lock) lock = true; #
input critical section;
        lock = false; # Entrance non-
critical section;
    }
}
process CS2 {
    while (true) {
        while (true) {
            await (! Lock) lock =
true; # input critical section;
            lock = false; # Entrance
non-critical section;
        }
    }
}
```

The advantage of the latter program is that it can be used to solve a critical section problem for any number of processes. They will all share the lock variable and execute the same protocols. The use of variable lock instead of in1 and in2 is very important, since almost all machines, especially multiprocessors, have special instructions for performing

conditional indivisible actions. Here is an instruction called «test and set – TS», as an argument gets a lock variable and returns a boolean value. In indivisible action, the TS statement reads and stores the value of the variable lock, sets it to true, and then returns the stored previous value of the lock variable. The result of the TS instruction is described by the following function:

```
bool TS (bool lock) {
    bool initial = lock; # save the
original value
    lock = true; # set the lock
    return initial; # return the origi-
nal value
}
```

Using the TS instruction, it is possible to implement a large module version of the program for the critical section problem. Conditional indivisible actions are replaced by cycles. The cycles do not end until the lock variable becomes false, that is, the TS statement returns false. The following solution works for any number of processes.

Using a blocking variable is usually called a spin lock because the process continually repeats the cycle while waiting for the lock to be unlocked.

```
# Critical «check-install» sections
bool lock = false; # collective variable
process CS [i = 1 to n] {
    while (true) {
        while (TS (lock)) skip; # login
protocol critical section;
        lock = false; # exit log non-
critical section;
    }
}
```

Mutual exclusion is ensured: if several processes attempt to enter the critical section, only one of them will first change the value of, therefore, only one of the processes successfully completes its input protocol and enters the critical section. The lack of interlocking implies that if both processes are in their input protocols, then the lock is false, and therefore, one of the processes will enter its critical section. Unwanted delays do not occur because, if both processes go beyond their critical sections, the lock is false and, therefore, one of the processes can successfully enter the critical section if the other executes a non-critical section or is completed.

On the other hand, the performance of the sign-in property is not guaranteed. If a strong planning strategy is used in a strong sense, then the process attempts to enter the critical section will succeed, since the lock variable will often end up being «false». In the weakest sense of the most commonly used scheduling strategy, the process may become permanently

stuck in the log. However, this can only happen if other processes consistently enter their critical sections, which in practice should not be. Therefore, this high-resolution solution satisfies the condition of a fair planning strategy.

Bibliography

1. Levin, M.P. Parallel programming using OpenMP / M.P. Levin. – M.: Internet University of Information Technology, Binom. Laboratory of Knowledge, 2008.
2. Стан змагання. Проблема критичних ділянок. Взаємне виключення з активним очікуванням. Аналіз підходів до рішення проблеми:
3. <https://sites.google.com/site/ekzamenzos/bilet-4>.
4. Критичні секції та блокування: https://studopedia.net/3_75931_kritichni-sektsii-ta-blokuvannya.html
5. Эндрюс Г. Р. Основы многопоточного, параллельного и распределенного программирования. – М.: Издательский дом «Вильямс», 2003. – 512 с.
6. Методичний посібник. Частина II: <https://dspace.uzhnu.edu.ua/jspui/bitstream/lib/21330/1/Методичний%20посібник.%20Частина%20II.pdf>.

USING OF JUNIT FRAMEWORK FOR AUTOMATION UNIT TESTING IN THE CONTEXT OF TEST-DRIVEN DEVELOPMENT

Abalmasova D.Y.

Bachelor's Degree student Kharkiv National University of Radio Electronics

Keywords: automation, unit testing, JUnit, test-driven development

In this thesis consider the problem of automation unit testing using the JUnit framework. Before proceeding to the JUnit framework and test-driven development it is important to consider concepts like testing, automation and unit testing. Testing is a significant part of the system development life cycle. It ensures that almost all of the bugs are found and eliminated, and that the product meets the expected requirements. Nevertheless, some test are too labour-consuming when perform them manually even though they are seem easy enough. In such cases, automated testing comes to the rescue. Automated testing is a technique in which a tester writes test scripts and uses the appropriate software to run and monitor the execution of test suites. The purpose of an automation is to iron out defects and problems with software, development process and with business by increasing the testing efficiency.

Automated testing has many advantages. First of all, automation testing can be done remotely and in any time of the day. The tester can run test scripts before leaving the office and get results of the testing at the morning when come back. Therefore, automation saves a lot of time. Also there are no need in a big QA team for automation. It is enough just

have one or couple experienced automation engineers that will write all test and then run them. This saves money on human resources. Moreover, the automation test can be run on hundreds of devices simultaneously what is impossible to do manually.

Automation is also applicable when talking about unit testing. It is possible to carry out unit testing manually but is in most cases automated. Unit testing is aimed to check if the separate module of a software works properly. The developer writes and executes the unit tests.

It is also important to understand that the unit is any isolated part of the code like a class, function, interface, procedure etc. which behavior can be checked apart other modules of the project. Unit test can be also considered as a project functionality documentation. Other developers that do not know the specific of the project can read unit tests to understand how the system works A unit test can be considered good if created using the principles of easing of writing, readability, reliability and it should not be like an integration test. The question often arises of what is the difference between unit and integration testing. Unlike unit testing, when doing the integration testing, tester considers the different parts of software as a

one group and tests if these modules work correctly together.

Unit testing is a part of the test-driven software development process. TDD assumes that the developer first creates negative unit tests and then writes code in order to fix the application until the test passes. The main idea of failing tests is to force the developers to take into account all possible errors, inputs and outputs. Test-driven development implies that a unit test is able to run fast, run separately and use live data. Therefore, unit tests should be narrowly focused and should not check many different things at once [1].

A unit test usually includes three stages. These stages are also called AAA pattern that means Arrange, Act and Assert. The unit test is successful if the actual result in all three phases corresponds to the expected result. Unit tests written using the AAA syntax will help to determine exactly where to look for errors when testing fails.

The AAA pattern organizes and clarifies test code by dividing it into the following functional segments:

- the Arrange part initializes objects and specifies the value of the data that is transferred to the test case. E.g. create an empty cart;
- the Act part calls a test case with ordered parameters. E.g. Add a product to the shopping cart;
- the Assert part checks that the test case returns the expected result. E.g. the quantity of the products in the cart is increased by one.

Often, developers use JUnit to create unit tests. JUnit is combination of Java

and Unit words. Therefore, JUnit is a framework for creating and running unit tests. In addition, it is a member of the xUnit. The main idea of JUnit is set up the test kits for a segment of code that firstly should be tested and only then fulfilled. This makes the program code more stable, as well as the efficiency of the developer, while reducing the debugging time. JUnit is a perfect open source early bug finder, which is best for TDD environment.

It is not hard to integrate JUnit framework with different integrated development environments and build systems. It is installed by default in such common IDEs as Eclipse or IntelliJ. For build systems like Gradle or Maven the installation of JUnit is performed via build.gradle or pom.xml.

JUnit consists of assertions and annotations that are build in the code while testing. Assertions are methods that provides the conditions of assertions in tests. Annotation is an indicator that shows the compiler what to do with the code that goes after this annotation. JUnit annotations are very simple and at the same time important when writing Junits. With a help of annotations JUnit understand what to do with the processing part of code.

The most common annotations are @Before, @Test, @After, @BeforeClass and @AfterClass. “Before” annotation is used to execute code before every test. It is used to initialize methods before running the test. “Test” annotation includes the code for the initial test. It is ruined after the “Before” code has been processed.

“After” annotation indicates the code runs after the testing code has been carried out. This helps to destroy the variables and clean the memory. The “BeforeClass” annotation is static. It runs only once and points out the code runs before running all tests. The “AfterClass” annotation specifies that the code located under this annotation is performed after all the tests have been already executed [2].

JUnit includes such significant features as Fixtures, Test suites, Test runners and JUnit classes. Fixture is a fixed objects state that can be used as basis for executing tests. The test fixture goal is to guarantee that there is familiar and stable environment that allows running tests in a way that returns the same results. It contains setUp method that carries out before each test call and tearDown method that starts after each test method. A test suite combines several test cases to execute them together. Developer can use “RunWith” and “Suite” annotations to perform the test suite. Test runner executes the test cases. JUnit classes are essential classes that programmer uses for writing and checking JUnits. For example, class Assert that keeps a kit of assert methods, class TestResult which contains methods for collecting the results of executed test cases and class TestCase that sets a fixture.

The usual flow of using the JUnit includes the creating a java test class, adding a test method to the test class, adding the appropriate annotations, implementing the test condition, creating a TestRunner java class, executing the test case. After all these steps the failures and success results are returned. The next steps are compiling

the Test case and Test Runner classes with javac, running the Test Runner, which executes the test case, described in the Test Case class and the final point is verifying the output. There are three ways to run JUnits: directly from the command line, from the IDE and with a help of build systems. It is recommended to keep unit tests is a separate project or source folder from the project code [3].

Thus, there is a list of the main but not all advantages of JUnit framework. First is that it is included almost in all integrated development environments and build automation tools, so there is no need to spend time on the additional setup. JUnit is not overloaded with excessive functionality and provides easy in use features for creating and editing the test cases. It enables to write test cases on development stage that gives the possibility to test and detect issues early. Moreover, JUnit with its automated repeatable test cases helps to ensure that the software still works correctly each time the changes are deployed to the code. In sum, JUnit, thanks to its ease of use and flexibility, is the most preferred unit testing framework if the project has been written in Java programming language using test-driven software development process principles.

References

1. Gulati S., Sharma R.. Java Unit Testing with JUnit 5: Test Driven Development with JUnit 5. Appress, 2017. – 168 p.
2. Link J., Fronlich P. Unit Testing in Java: How Tests Drive the Code Morgan Kaufmann, 2003. – 376 p.
3. Langr J., Hunt A., Thomas D.. Pragmatic Unit Testing in Java 8 with JUnit. Pragmatic Bookshelf, 2015. – 236 p.

OPENMP PARALLEL PROCESSING

Avdieiev O. D.

Scientific leader – senior lecturer of Software Engineering department Oliinik O.V. Kharkiv National University of Radioelectronics
(61166, Kharkiv, Lenin Avenue, 14, faculty of computer science,
Phone (057)702-14-65), e-mail: oleksii.avdieiev@nure.ua

Keywords: OpenMp, compiler, thread, interface, parallelization, mechanism, language, data, region.

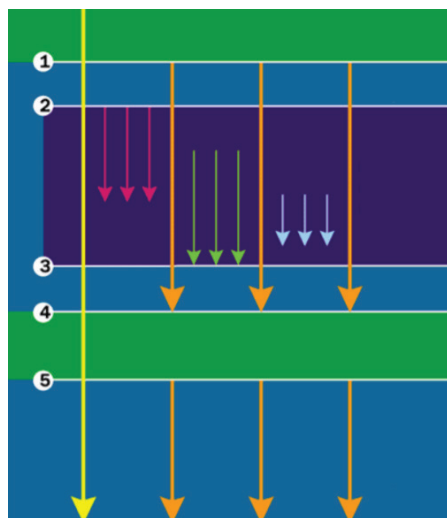
OpenMP (Open Multi-Processing) is a set of compiler directives, library procedures, and environment variables that are designed to program multi-threaded applications on multi-processor systems with shared memory (SMP systems).

OpenMP can be thought of as a high-level add-on for Pthreads (or similar thread libraries). We list the advantages that OpenMP gives the developer. With the idea of “incremental parallelization,” OpenMP is ideal for developers who want quickly parallelize their computing programs with large parallel loops. The developer does not create a new parallel program, but simply sequentially adds OpenMP directives to the program text. At the same time, OpenMP is a fairly flexible mechanism that provides the developer with great control over the behavior of the application. It is assumed that the OpenMP-program on a single-processor platform can be used as a serial program, that is, there is no need to support serial and parallel versions. OpenMP directives are simply ignored by the serial compiler, and stubs can be substituted for calling OpenMP procedures, the text of which is given in the specifications. One of the advantages of OpenMP, its developers

consider the support of the so-called “orphan” (torn off) directives, that is, the synchronization and work distribution directives may not directly enter the lexical context of the parallel domain.

An OpenMP application starts with a single thread – the main one. The application may contain parallel regions, entering which the main thread creates groups of threads (including the main thread). At the end of the parallel region, the thread groups stop and the execution of the main thread continues. Other parallel regions can be nested in a parallel region, in which each stream of the initial region becomes the main one for its group of flows. Nested regions may in turn include regions of a deeper nesting level. The parallel processing in OpenMP is illustrated in the picture 1. The leftmost arrow represents the main thread that runs alone until it reaches the first parallel region at point 1. At this point, the main thread creates a group of threads, and now they all run simultaneously in the parallel region.

At point 2, three of these four flows, reaching a nested parallel region, create new groups of flows. The original main and flows that created new groups become the owners of their groups (main



“OpenMP parallel sections” **Picture 1**

in these groups). Also, threads can create new groups at different times or not meet a nested parallel region at all. At point 3, the nested parallel region ends. Each thread of a nested parallel region synchronizes its state with other flows in this region, but synchronization of different regions between themselves is not

performed. At point 4, the first parallel region ends, and at point 5, a new one begins. The local data of each stream between the parallel regions is saved.

These are the basics of the OpenMP runtime model. Based on the foregoing, we can conclude that OpenMP is a mechanism for writing parallel programs for systems with shared memory, which consists of a set of compiler directives and library functions and makes it easy to create multi-threaded applications in C / C ++.

List of references:

1. Levin, M.P. Parallel programming using OpenMP / M.P. Levin. – M.: Internet University of Information Technology, Binom. Laboratory of Knowledge, 2008.
2. Barbara, Chapman Using OpenMP – Portable Shared Memory Parallel Programming / Barbara Chapman. – Moscow: World, 2007.
3. Antonov, A. S. Parallel programming technologies MPI and OpenMP / A. S. Antonov. – Moscow: St. Petersburg. [et al.]: Peter, 2012.

RXJS LIBRARY USAGE FOR REACTIVE PROGRAMMING IN JAVASCRIPT

Mylokhin D.O.

Bachelor's Degree student Kharkiv National University of Radio Electronics

Keywords: RxJS, reactive programming, observer, observable, operator, subject, asynchrony.

In this thesis consider the problem of using the RxJS library for reactive programming. First need to understand what reactive programming is. Reactive programming – programming with asynchronous data streams. It is necessary in order to model any subject area with the help of reactions and calculations. Different data models can be used for this. Reactively programmer can bind properties of objects, table cells, structures such as trees, graphs, etc. The main idea is that changes can be automatically propagated through the data flow.

Reactivity is needed when it comes to huge amounts of data, as well as multi-user mode. In this case, there is a need for asynchronous processing to make the system responsive and fast. Asynchrony can become very problematic and make it difficult to understand and maintain code, and reactive programming, in turn, simplifies working with asynchronous threads.

Reactive programming has many advantages and here is a short list of them:

- helps avoid callback issues;
- makes code cleaner and more readable;
- helps to easily compose data streams;
- provides many operators that simplify the work;

- makes threading very easy;
- simplifies realization of back pressure.

No matter which library and language will be used for reactive programming, in any case it will be based on the Observer Design Pattern. This pattern allows components of an application to react to certain incoming events or streams of data efficiently. This is also known as callback-based or event-driven code. Thus, the sense of the Observer pattern is that the functions are triggered in response to incoming data.

There are few things that should be known when talking about Observer Design Pattern. First, is that the «listening» to the stream is called subscribing. Second, the functions we are defining are observers. And the last but not least, the stream is a sequence of ongoing events ordered in time and it represents the subject being observed.

Web applications today are event-driven. This means that they respond to external services and data sources. ReactiveX, also known as Reactive Extensions, helps to work with this. This library is implemented in many languages. But the most common is the implementation of Javascript – RxJS. RxJS is one of the most

popular libraries in web development today as it can be used in both server and client applications.

RxJS includes three key points: Observable, Observer and Schedulers.

Observables are data streams. They can be considered as a provider that processes and gives data to other components. It packs data that then can be transferred from one stream to another. Observable usually emits the data periodically or only once in its life cycle based on the configurations. This concept is useful to determine when the Observables should start emitting items. Cold Observable will only start emitting items once it has a subscriber. It is a good approach when previous items are no longer necessary to process. Hot Observables will start emitting items even if no observers have subscribed to it, this allows observers to enter into a stream as they subscribe and be able to gain previously emitted items.

Observers receive the data emitted by the observable by subscribing to it with the help of `subscribeOn()`. All registered observers receive the data each time the observable emits them. If observable throws an error, the observer will receive it in `OnError()` event. In addition, various operators help observer to give out some specific data based on particular events [2].

Schedulers are the component of Rx. It tells observable and observers, on which thread they should run and observe accordingly.

Talking about RxJS it is not possible not to mention operators. Around one hundred and twenty operators can be

found in RxJS. and comprehensive documentation on how to use them when programming reactively. An operator is a function that carries out a certain action. When working with RxJS developer may notice some familiar functions that persist in JavaScript. Examples of these functions can be filtering, sorting, counting and searching. They have the same idea and concept but the code is different. Such functions take a value, perform some actions with it, and then return it. The returned value can be an array, an object or anything that passed.

Often it is needed to execute several operations with the data set. Instead of creating all these functions and locate them in the application, there is one convenient style of its implementation.

Operator is not something that necessarily should be used. However, there are cases when operator can help to simplify the task. When need to display a list of records according to a specific criterion for example.

As it has been already mentioned above, there is a big number of operators in RxJS. Nevertheless, it is not complicated to find the necessary one. Almost all operators have names that describe what it does.

It is not required to know all of them. In practice, there are four or five common operators help to realize most of the goals. The rest can be found in the RxJS library API page that contains all of them.

Finally yet importantly, what should be mentioned in RxJS is Subjects. It is possible to subscribe multiple Observers to Subjects and at the same time can use

Subjects as and Observable themselves. Subjects are a mix of Observable and Observer because they inherit characteristics of both. Subjects are kind of intermediaries that facilitates communication between Observables and Observers. Thus, Subjects well suited for cases when need to implement a flexible solution using a mix of Observables and operators [1].

There are four types of Subjects in RxJS. First is a standard Subject. It has no initial value and no replay behavior. Second is BehaviorSubject. A Behaviour Subject gives off the most recent emanated item to any new subscribers, before sending out any following items that are send forth. Third ReplaySubject emits all early source Observable outputs to new subscribers, and also transfer newly emanated items to them. The last fourth AsyncSubject. It only emits the latest value to all its Observers when it finishes.

In conclusion, reactive programming helps when it comes to certain types of high-load or multi-user applications with

real-time data streaming such as games, social networks, audio apps and etc.

But do not use reactive programming when there is no live large amount of data and users that will interact with these data at the same time.

Reactive programming has become widespread as the Internet has grown into a huge system of dynamic event-driven applications.

The tools used to develop web applications now allow applications to easily respond to events in real time, without knowing anything about the service from which they are received. RxJS is the main object of this revolution.

Remember, it is not hard to learn how to code reactively. The most difficult part is learn how to think reactively.

References

1. Koutnik R.. Build Reactive Websites with RxJS: Master Observables and Wrangle Events. Pragmatic Bookshelf, 2018. – 196 p.
2. Daniels P.P., Atencio L.. RxJS in Action. Manning Publications, 2017. – 352 p.

THE DESCRIPTION OF “WEB PORTAL FOR INTERACTION OF DIFFERENT CATEGORIES OF PARTICIPANTS IN THE EDUCATIONAL PROCESS” FUNCTIONS



Nesmiian Dmytro

4th year student Computer Science faculty Kharkiv National University of Radioelectronics

Keywords: web-portal, students, education, entrants, educational institution, announcement.

The mentioned system “Web portal for interaction of different categories of participants in the educational process” will allow person, who will search information about an educational institution, to find information about certain educational institutions and activities that take place there. These persons may be students or entrants. Universities and schools serve as educational institutions. In the future, all interested institutions may be involved.

This program will be an alternative to existing systems for announcing important events in educational institutions. The drawbacks of competitors will be considered when the system will be developing. There are some issues that were found. Most official sites do not conveniently display information about the institution they represent [1]. In addition, many of them do not have easy access to information about events that will take place at the university [2]. At least two third-party sites should be used to find information about activities that students visit at these institutions, and they do not always display accurate information [3]. This process is difficult for users.

The event information will help people who are not part of this institution to get an idea of the institution. And with their help it will be possible to draw conclusions about the activity of students in educational institutions.

This project will allow people to simplify the search for information about educational institutions and activities conducted in these institutions using special form for this project called as “announcements” quickly and efficiently. Information can be provided by both students and the administration of these institutions. In the first case, the information will be presented as announcement of an event that was visited or planned to be attended by students or students of the institution. In case of administration, the information may also be represented as announcement or be placed on the information page of the institution, faculty or specialty page.

The web portal implements its own algorithm for adding announcements. The announcement display depends on the category selected. There are two categories in total: private and public. In case of selecting a private category, the announcement will be available only within

this group. A public announcement will be available in generally accessible categories (for example, “specialty”, “faculty”, “university”). Announcements will be ranked by date added. For a fee, the author can place the announcement in the area of “hot” announcements (those that appear above the rest) of the selected category.

Each announcement is equipped with a “+” button. Pressing this button user will increase the rating of interest in this event. Thanks to this rating, it will be possible to find out more information about the interest of users in certain events. In addition, the information obtained will help to analyze and to identify the main priorities the actions of entrants in time of choosing an educational institution. This information will allow universities to improve their admission campaigns in the years to come and to get those students who are most likely to study in certain specialties. Also, web portal will provide some addition information such as:

- the most active students;
- specialties that are very popular with students;
- Universities with the highest involvement in various activities.

For stable operation, the web portal must support following functions:

- registration of new users and granting them certain rights;
- viewing and editing personal information;
- removing users from the system if necessary;
- user authorization;

- registration of a new educational institution;
- viewing information about educational institutions registered in the system;
- changing information about the educational institution;
- removing information about the educational institution if necessary;
- adding information about new events as ads
- reviewing announcements about activities organized by the administration of the institution;
- review announcements for activities that students visit;
- changing the status of your ads if the event has already taken place;
- sorting the information using special filters and viewing universities’ students activity rating.

To implement the functionality of the application, the JavaScript programming language and Express framework, NodeJS technology must be used.

As a result of this work, the requirements for a future web portal were identified. Web portal will be developed for the interaction of different categories of participants in the educational process, which allows the administration of the educational institution to add information about the educational institution, and also allows students to add information about the activities. The information that will be given allow universities to improve their admission campaigns in the years to come and to get those students who are most likely to study in certain specialties.

Sources:

1. V. N. Karazin Kharkiv National University (Official Site): [Electronic resource]. – Access mode: <https://www.univer.kharkov.ua/en> (Date of appeal: 20.11.2019);
2. NURE | Kharkiv National University of Radio Electronics: [Electronic resource]. – Access mode: <https://nure.ua/en/> (Date of appeal: 20.11.2019);
3. Освіта в Україні та за кордоном – Освіта. UA: [Electronic resource]. – Access mode: <http://osvita.ua/> (Date of appeal: 20.11.2019).

АНАЛІТИЧНИЙ ОГЛЯД МЕТОДІВ АУТЕНТИФІКАЦІЇ

Тучкова М.С.

Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського»

Ключові слова: аутентифікація, строга аутентифікація, апаратний токен, біометрична аутентифікація.

Keywords: authentication, strict authentication, hardware token, biometric authentication.

Вступ

На сьогоднішній день цифрові канали зв'язку відіграють важливу роль в житті людини: зв'язок з клієнтами, комунікації між співробітниками, онлайн-банкінг тощо. Велика кількість інформації з обмеженим доступом переноситься, зберігається і обробляється в інформаційних системах. Аутентифікація є важливим фактором, від якого напряму залежить рівень безпеки конфіденційної інформації. Однією з багатьох загроз при використанні слабкої аутентифікації є масовий злом облікових записів користувачів, що призводить до викрадення або знищення конфіденційних даних.

Сучасна людина активно користується мережею Інтернет і може мати десятки, а то і сотні облікових записів для доступу на ті чи інші ресурси. Працюючи з сотнями акаунтів і десятками «підключених» мобільних пристроїв, кожен користувач потребує надійного, зручного та комфортного способу аутентифікації. Для захисту інформації з обмеженим доступом від несанкціонованого доступу та уникнення можли-

вих загроз необхідно комплексно підходити до аналізу методів аутентифікації.

Основна частина

Ідентифікація (англ. identification) – процедура розпізнавання користувача за його ідентифікатором (ім'ям). Ця функція виконується, коли користувач робить спробу увійти в мережу. Користувач повідомляє системі за її запитом свій ідентифікатор і система перевіряє в своїй базі даних його наявність.

Аутентифікація (англ. authentication) – процедура перевірки аутентичності заявленого користувача, процесу або пристрою. Ця перевірка дозволяє достовірно переконатися, що користувач (процес або пристрій) є саме тим, ким себе оголошує. При проведенні аутентифікації перевіряюча сторона переконується в достовірності тої сторони, що перевіряється, при цьому та сторона, що перевіряється, також активно бере участь в процесі обміну інформацією. Зазвичай користувач підтверджує свою ідентифікацію, вводячи в систему унікальну, невідому іншим

користувачам інформацію про себе (наприклад, пароль або сертифікат). Після ідентифікації та аутентифікації суб'єкта виконується його авторизація.

Авторизація (англ. authorization) – процедура надання суб'єкту певних повноважень і ресурсів в даній системі. Іншими словами, авторизація встановлює сферу його дії і доступні йому ресурси.

Адміністрування (англ. accounting) – реєстрація дій користувача в мережі, включаючи його спроби доступу до ресурсів[1].

Для коректної аутентифікації користувача необхідно, щоб користувач пред'явив аутентифікаційну інформацію – унікальну інформацію, якою володіє тільки користувач.

Виділяють три основні типи аутентифікаційної інформації:

1. Користувач, якого перевіряють, знає якусь унікальну інформацію.

Приклад: парольна аутентифікація.

2. Користувач має предмет з унікальними характеристиками або вмістом.

Приклад: смарт-карта, USB-токен тощо.

3. Аутентифікаційна інформація є невід'ємною частиною користувача.

Приклад: відбиток пальця або інші види аутентифікації користувача за його біометричними ознаками[2].

Процедура аутентифікації виконується в два послідовних етапи:

1. У користувача береться еталонний зразок аутентифікаційної інформації, наприклад, пропонується ввести

пароль (або даний зразок генерується випадковим чином і потім записується на смарт-карту користувача). Даний зразок зберігається у суб'єкта системи, котрий перевіряє аутентифікацію – модуля аутентифікації (наприклад, сервера, який виконує аутентифікацію користувачів). Срок дії еталону обмежений, тому по завершенні необхідно повторно надати зразок аутентифікаційної інформації.

2. Кожен раз при виконанні аутентифікації у користувача запитується аутентифікаційна інформація, яка порівнюється з еталоном. На основі даного порівняння робиться висновок про справжність користувача[1].

Таблиця відповідності ідентифікаторів і еталонних зразків зберігається в модулі аутентифікації. Еталон може зберігатися як в явному вигляді, так і в якості результату деякого перетворення отриманої інформації. Наприклад, при реєстрації користувача в системі з біометричної аутентифікації після запиту відбитка пальця виконується його згортка в кодове значення, що залежить від основних параметрів відбитка пальця. Згодом, при аутентифікації користувача виконується аналогічне перетворення, результат якого порівнюється з кодовим значенням. Зберігання відбитка (еталону) безпосередньо у вигляді зображення вимагало б значно більших обсягів пам'яті, а також ускладнило б порівняння зразків та збільшило час аутентифікації.

Аутентифікаційна інформація не повинна зберігатися у відкритому вигляді із міркувань безпеки – напри-

клад, еталонні зразки паролів зберігаються в модулі аутентифікації або в зашифрованому вигляді або у вигляді хеш-значень.

Аутентифікація може бути як односторонньою (сервер перевіряє користувача на предмет подальшого надання доступу або відмови в ньому), так і взаємною (взаємна перевірка достовірності учасників інформаційного обміну).

Для підвищення стійкості аутентифікації (запобігання можливості аутентифікуватися під чужим ім'ям) часто використовують кілька методів аутентифікації одночасно. Наприклад, одночасне використання смарт-карти та PIN-коду (пароль для доступу до інформації, що зберігається на смарт-карті); одночасне використання пароля та відбитка пальця. Аутентифікація на основі одночасного пред'явлення аутентифікаційної інформації двох видів називається двофакторною. Трьохфакторна аутентифікація визначається аналогічним чином.

Парольна аутентифікація. На сьогодні парольна аутентифікація є найбільш поширеною завдяки простоті використання. Проте вона має суттєві недоліки:

На відміну від випадково сформованих криптографічних ключів (які може містити унікальний предмет), паролі користувача досить легко підібрати через недбале ставлення більшості користувачів до формування пароля. Нерідко користувачі обирають легко передбачувані паролі, а саме: ім'я користувача або ім'я користувача, за-

писане в зворотному порядку; короткі паролі, які зламуються перебором всіх можливих варіантів;

Існують вільно доступні програми підбору паролів;

Пароль може бути отриманий шляхом застосування насильства до власника;

Пароль можна підглядіти або перехопити при введенні[1].

Аутентифікація за допомогою унікального предмета. Аутентифікація за допомогою унікального предмета забезпечує більш надійний захист, ніж парольна аутентифікація.

Предмети, що використовуються для аутентифікації, можна розділити на наступні групи:

«Пасивні» предмети, які містять аутентифікаційну інформацію (наприклад, випадково сгенерований пароль) і передають її в модуль аутентифікації за вимогою. Аутентифікаційна інформація може зберігатися в предметі як у відкритому вигляді (магнітні карти, смарт-карти з відкритою пам'яттю, Touch Memory), так і в захищеному вигляді (смарт-карти з захищеної пам'яттю, USB-токени). В останньому випадку необхідно ввести PIN-код для доступу до даних, що перетворює предмет в засіб двофакторної аутентифікації.

«Активні» предмети, які володіють достатніми обчислювальними ресурсами і здатні брати активну участь в процесі аутентифікації (прикладі: мікропроцесорні смарт-карти і USB-токени). Ця можливість особливо цікава при віддаленій аутентифікації

користувача, оскільки на основі таких предметів можна забезпечити сувору аутентифікацію. Під цим терміном мається на увазі такий вид аутентифікації, при якому секретна інформація, що дозволяє перевірити справжність користувача, не передається у відкритому вигляді[3].

Аутентифікація за допомогою унікальних предметів також має ряд недоліків:

Предмет може бути викрадений;

Необхідність використання додаткових програм та спеціального обладнання для роботи з предметами;

Біометрична аутентифікація. Біометрична аутентифікація заснована на унікальності ряду характеристик людини. Для аутентифікації використовуються наступні характеристики:

Відбитки пальців;

Візерунок райдужної оболонки ока і структура сітківки ока;

Риси обличчя;

Форма кисті руки;

Параметри голосу;

Схема кровоносних судин особи;

Форма і спосіб підпису;

В процесі біометричної аутентифікації еталонний і пред'явлений користувачем зразки порівнюють з деякою погрішністю, яка визначається і встановлюється заздалегідь. Похибка підбирається для встановлення оптимального співвідношення двох основних характеристик, які використовуються засобом біометричної аутентифікації:

FAR (False Accept Rate) – коефіцієнт помилкового прийняття (стороння

особа пройшла аутентифікацію під ім'ям легального користувача).

FRR (False Reject Rate) – коефіцієнт помилкової відмови (легальний користувач системи не пройшов аутентифікацію).

Обидві величини вимірюються у відсотках і мають бути мінімальні. Також величини є зворотньо залежними, тому аутентифікуючий модуль при використанні біометричної аутентифікації налаштовується індивідуально – залежно від використовуваної біометричної характеристики і вимог до якості захисту. Засіб біометричної аутентифікації має дозволяти налаштувати коефіцієнт FAR до величин порядку 0,01 – 0,001% при коефіцієнті FRR до 3 – 5%[3].

Залежно від використовуваної біометричної характеристики, засоби біометричної аутентифікації мають різні переваги і недоліки.

Загальний недолік біометричної аутентифікації – необхідність в обладнанні для зчитування біометричних характеристик, яке може бути досить дорогим.

Висновки

За результатами огляду та аналізу сучасних методів аутентифікації можна зробити висновок про те, що використання строгої аутентифікації з використанням унікального предмету є найбільш перспективним, а саме використання криптографічного апаратного токена, оскільки він надійно захищений PIN-кодом. PIN-код є фактором знання, що дозволяє використовувати двофакторну аутентифікацію

за допомогою криптографічних токенів і знижує ймовірність виникнення помилок першого і другого роду, тим самим підвищуючи захищеність інформаційних систем.

Література

1. Шаньгин В. Ф.. Информационная безопасность компьютерных систем и сетей.: Москва, 2011. С. 143-155.
2. Протоколы аутентификации (идентификации): [Электронный ресурс] // Лекции. – Режим доступа: <https://sites.google.com/site/anisimovkhv/learning/kripto/lecture/tema11> дата звернення: 02.11.2019).
3. Проскурин В. Г., Крутов С. В., Мацкевич И. В.. Программно-аппаратные средства обеспечения информационной безопасности. Защита в операционных системах: учеб. пособие для вузов. М.: Радио и связь, 2000

РОЗРОБКА ПРИСТРОЮ ЗБЕРІГАННЯ ПЕРСОНАЛЬНИХ ЕЛЕКТРОННИХ КЛЮЧІВ

Тучкова М.С.

Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського»

Ключові слова: паролі, NFC, аутентифікація, строга аутентифікація, апаратний токен.

Keywords: passwords, NFC, authentication, strict authentication, hardware token.

Вступ

На сьогоднішній день мобільні пристрої, зокрема смартфони та планшети, зберігають значну частину нашої конфіденційної інформації. Ці пристрої використовуються для доступу до електронної пошти, підключення до соціальних мереж і навіть керування банківськими рахунками.

Втрата телефона може серйозно підірвати вашу конфіденційність та призвести до крадіжки особистих даних. Навіть якщо хтось з знайомих використовує ваш пристрій, це може призвести до незручних і небажаних ситуацій.

Останні роки також набирають популярність програми-менеджери паролів, які дозволяють зберігати всю необхідну інформацію маючи один універсальний ключ. Але цей метод також має свої вагомні недоліки.

Працюючи з сотнями акаунтів і десятками «підключених» мобільних пристроїв, кожен користувач потребує надійного, зручного та комфортного засобу ідентифікації. Але необхідність тримати в голові довгі складні паролі

відходить у минуле, тому ідея створення універсального цифрового ключа є надзвичайно актуальною.

Основна частина

Пристрій зберігання електронних ключів має низку переваг на відміну від інших методів аутентифікації, серед них:

1. Зручність зберігання електронних ключів в єдиному захищеному пристрої. Сучасні рішення для захисту персональних даних з використанням PIN-кодів, паролів та графічних ключів є громіздкими і незручними, тому захист персональних даних за допомогою цифрового ключа є актуальним на сьогоднішній день.

2. Мінімізація впливу людського фактору. В той же час спрацьовує людський фактор: можуть виникати труднощі із запам'ятовуванням складних паролів, особливо, якщо їх багато. Іноді пароль може бути дуже простим, але використовуватися дуже рідко – як наслідок він просто буде забутий. Для часткового уникнення людського фактору паролі можуть бути збережені,

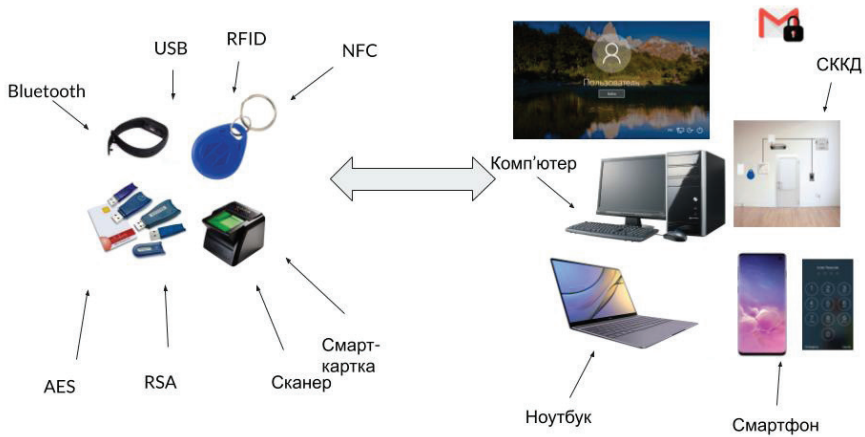


Рисунок 1. Засоби аутентифікації та область їх застосування

наприклад, записані на папері, збережені в електронній таблиці на комп'ютері або записній книжці телефона. Але чи є надійним таке місце збереження того чи іншого паролю? Навряд чи пересічна людина цим переймається, а у шахраїв навпаки з'явилося більше шансів заволодіти важливими даними.

3. Підвищений рівень захисту даних. А саме: використання виділеного передавача для зв'язку з пристроєм; NFC, оскільки радіус його дії до 10 см; використання сучасного виду шифрування (AES)[1];

Розглянемо більш детально технології та засоби, які можуть використовуватись для здійснення аутентифікації.

На рисунку 1 зображено пристрої, які є засобами аутентифікації та зберігають в собі паролі та засоби біометричної аутентифікації, наприклад, відбитки пальців, зображення райдужної оболонки ока або малюнку вен на долоні.

Такі пристрої можуть мати різний форм-фактор і бути реалізованими у вигляді брелоку, картки, браслету або кільця.

Зв'язок з іншими пристроями може здійснюватися за допомогою таких технологій передачі даних як Bluetooth, NFC, USB, RFID[3].

Захищеність даних забезпечується такими криптографічними алгоритмами як RSA та AES.

Область застосування аутентифікації:

1. Розблокування облікового запису на комп'ютері або ноутбуці;
2. Аутентифікація на сайтах;
3. Розблокування смартфона;
4. Доступ до приміщень[2];

Аналіз характеристик існуючих пристроїв зберігання даних для індивідуальних електронних ключів показав, що серед технологій передачі даних найбільшим попитом користуються Bluetooth, проте також є пристрої,



Рисунок 2. Сценарії роботи пристрою зберігання електронних ключів

що використовують USB та NFC. Для шифрування даних використовується AES 128-бітний та 256-бітний. Більшість пристроїв виготовлена у формі брелоку.

Проаналізовано основні технології передачі даних та шифрування інформації, що застосовуються в цифрових ключах.

NFC і Bluetooth – сучасні технології зв'язку малого радіусу дії. Суттєва перевага NFC над Bluetooth – коротший час встановлення з'єднання. Замість виконання інструкцій за погодженням для ідентифікації Bluetooth-пристрою, зв'язок між двома пристроями NFC встановлюється відразу (менш ніж за одну десятю секунди).

Максимальна швидкість передачі даних NFC (424 кбод) менше, ніж Bluetooth (24 Мбод). У NFC менший радіус дії (менше 20 см), що забезпечує більший ступінь безпеки і робить NFC

більш пристосованим для переповнених просторів, де встановлення між сигналом і фізичним пристроєм, який його передає (і як наслідок, його користувачем), інакше може не здійснитись. На відміну від Bluetooth, NFC сумісна з існуючими RFID-структурами[1].

Передача інформації за допомогою USB використовується швидше як допоміжний спосіб передачі даних (наприклад, для налаштування пристрою), адже USB пристрої дозволяють змінювати мікропрограму мікросхеми, що відповідає за взаємодію з комп'ютером. Зловмисник, провівши реверс-інжиніринг конкретного пристрою, може створити і записати в нього шкідливий код[1].

Пристрій зберігання електронних ключів може працювати за двома сценаріями (Рис.2).

Алгоритм дій для підключення до комп'ютеру для передачі паролів:

1. Підключаємося до комп'ютера або мобільного телефону;

2. Відкриваємо спеціальне програмне забезпечення (менеджер паролів) та авторизуємося (вводимо логін та пароль);

3. Вносимо акаунти – набори даних для необхідних авторизацій. Для сайту необхідно ввести доменне ім'я, логін та пароль. Для розблокування комп'ютера з ОС Windows необхідно піднести пристрій достатньо близько до комп'ютера та обрати відповідний варіант розблокування у вікні блокування робочого столу;

4. Дані в зашифрованому вигляді передаються на пристрій та зберігаються;

Алгоритм дій для вводу пароля з пристрою зберігання паролів на комп'ютер:

1. Підключаємося до комп'ютера або мобільного телефону;

2. Відкриваємо браузер та вводимо адресу сайту. У вікні авторизації вводимо логін;

3. Доменне ім'я сайту передається на пристрій;

4. Пристрій шукає відповідну інформацію. Якщо пароль знайдено, то він передається на підключений пристрій (комп'ютер або телефон) та автоматично підставляється у вікно паролю. Якщо дані відсутні, то користувач повинен самостійно ввести пароль;

Результуючий пристрій обмінюється інформацію з іншими пристроями за допомогою NFC. Також технологія NFC дозволяє емулювати RFID-мітки, таким чином браслет дозволяє взаємо-

діяти із СККД та використовуватися в якості пропуску на підприємстві[4].

Для окремого пристрою або групи пристроїв може існувати пристрій-адміністратор. Такий пристрій може зберігати в собі сукупний набір інформації із кількох дочірніх пристроїв і використовуватися в екстрених ситуаціях, коли людина із дочірнім пристроєм відсутня.

Пристрій має на корпусі сканер відбитку пальця, та спеціальний кольоровий індикатор. Індикатор спрацьовує при обміні паролями із підключеним пристроєм. Пристрій очікує на обмін паролем. Для підтвердження передачі паролю користувач повинен надати свій відбиток. Таким чином кожна операція обміну інформацією потребує підтвердження. Це додає додаткового захисту у випадку, якщо обмін інформацією ініціює зловмисник, при цьому власник пристрою жодних операцій для запиту паролю не проводив.

Висновки

Порівняльний аналіз розглянутих фізичних пристроїв зберігання даних для електронних ключів показав, що існуючі продукти можуть не задовольняти вимоги, що до них висуваються в залежності від середовища використання, тому потребують вдосконалення.

Проаналізовано основні технології передачі даних та шифрування інформації, які використовуються існуючими продуктами, визначено їх переваги та недоліки. Обрано технології для моделі, що найкращим чином відповідають для використання на підприємстві.

Розроблена модель включає опис взаємодії з іншими пристроями, сценарії роботи, визначена сукупність функцій, яка дозволяє безпечно використовувати пристрій на підприємстві. А також дозволяє забезпечити прийнятний рівень безпеки для сучасних компаній та уникнути негативного впливу людського фактору, знизити ризик несанкціонованого доступу до конфіденційної інформації.

Література

1. Шаньгин В. Ф.. Информационная безопасность компьютерных систем и сетей.: Москва, 2011. С. 143-155.
2. Hidezz. Руководство пользователя: [Електронний ресурс]. – Режим доступу: <https://update.hideez.com/update/HideezKey101-UserManual-RU.pdf> (дата звернення: 01.09.2019).
3. Everykey – No more Passwords, No more Keys: [Електронний ресурс]. – Режим доступу: <https://everykey.com/> (дата звернення: 05.11.2019).
4. Kickstarter. Corelike: [Електронний ресурс]. – Режим доступу: https://www.kickstarter.com/projects/corelike/have-more-time-to-enjoy-your-life-with-corelike?ref=discovery_category&term=security (дата звернення: 20.10.2019).

PROCESSING OF OATS INTO GROATS AND FLAKES

Sots S.

c. tech. sc. (ph.D.), docent,
Odessa national academy of food technologies

Kustov I.

c. tech. sc. (ph.D.), senior lecturer,
Odessa national academy of food technologies

Kuzmenko Y.

c. tech. sc. (ph.D.), assistant,
Odessa national academy of food technologies

Key words: oats, groats, flakes, modes of processing, scheme of processing.

Standard products of processing oats in Ukraine are not crushed oats groat of which during further processing produces flaked groats, flakes “Hercules”, “Pelyustkovi”. Separate oat products are flakes “Extra” and “Tolokno” – special prepared oats flour.

The technological processes of processing oats are amongst the most difficult in cereal production and the need for large production areas for its implementation.

For example in the production of not crushed oats groats cleanup of impurities, divided into two factions and specially prepared oats grain enters operational silos passes magnetic control and goes to dehulling stage which carried out on two systems separately for each faction. Modes of dehullers set so that provide a minimum amount of crushed groats at the maximum value of the dehulling coefficient on the first dehulling system. After each dehulling system carried out sorting of dehulling products using the groats separation stage. First at sieving machine (usu-

ally use dressing reel) use sieves \varnothing 2,0 mm removing husking bran then on the two systems of aspirators removed hulls and remainder of husking bran. Mixture of dehulling products which consists of dehulled and unde-hulled crops sent to groats separation stage which is carried out on paddy separators. The classic scheme provides two sequential passages of these machines. Overflow from first Paddy machine represents unde-hulled crops which are sent for second dehulling system, underflow – dehulled crops control on second paddy separator and then sent to pearling. Pearling conducted on one system. Usually, at this stage are used pearlier machines. Also allowed the possibility of exclusion from the technological scheme if dehulling conducted at hulling stone and air transport is used to transport products of processing. Under these conditions, more intensive processing of grain at hulling stone and friction in the walls of products pipe line provide a similar surface treatment as the pearling system. After pearling are carried sorting

products, which are formed at this stage. From the mixture remove husking bran, particles of crushed groat and small amount of hulls. Sorting of pearling products is carried out at plansifter. Overflow from sieve 2,5x20 mm referred to feed impurities, underflow of sieve \varnothing 2,0 mm removed a mixture of crushed kernels and husking bran. Oats groat obtains by underflow of sieve 2,5x20 mm. End product for control passed through two control systems of paddy separator and then controlling in one system of air separators. In applying the classical scheme provides the production of high, first and second grades of groats, the total yield of finished products is 45-55 %.

Oats are the raw material for the production of a wide range of flaked products. From it produce flaked groats, flakes "Hercules", "Pelyustkovi" and "Extra." For all types of flakes as raw material allowed using oats groats or oats grain. Each type of flakes has its production technological features.

For the production of flaked oat groats are used high or first grade of oats groat. In the first stage groats sent to water heat treatment which consists in its steaming at horizontal screw steamer with the vapor pressure 0,05-0,10 MPa. Steamed groats temper in special insulated bins for 20-30 min. Flaking of groats allowed to carry on flaking machine or roller mills with ribbed rollers. Thickness of flakes typically is 0.7-0.9 mm. Flaked groats sent to sorting, during which by underflow of sieve \varnothing 2,0 mm conduct removing of husking bran and particles of crushed groats, by overflow from this sieve obtain flaked groats which

controlling by passed through two control systems of air separators. For the production of flakes "Hercules" is used high grade of oats groat. The classical scheme includes the following steps: steaming, tempering, flaking, sorting and cooling of end products. At the beginning of the technological process carried out additional control of groats at two consecutive systems of paddy machines and one system of groats separation. At systems of paddy machines remove unprocessed grain which return to dehulling systems or if plant does not have the conditions for it processing sent to waste products. Control of husking bran and crushed groats conduct at groats separation machines. By overflow from sieve 2,5x20 mm obtain groats which sent for further processing. Water heat treatment in the production of flakes "Hercules" carried out a similar scheme and modes which provides for the production of flaked groats. Flaking stage carry on flaking machine with smooth rollers, thickness of flakes typically is 0.6 mm. Obtained flakes are dried at belt dryer to standardized humidity 12,0 % and control at two control systems of air separators. For the production of flakes "Pelyustkovi" is used high or first grade of oats groat. At the beginning of the technological process carried out additional control of groats which carried similarly of the process of production flakes "Hercules" and then groats additionally send to one pearling system. Surface treatment of groats reduces the content of mineral elements to 1.9 % which is regulated by standard. The mixture of pearling products at first sorts at dressing reel where by underflow of sieve № 080

remove husking bran, after this at groats separation machine remove crushed kernels and separates groats into two factions which sent to control at air separators systems. Next stages: water heat treatment, flaking, drying and control operations carried similarly of the process of production flakes "Hercules".

Analyzing the classical technology of processing oats in groats and flakes it can be concluded that they are outdated, contain large amounts of energy-intensive operations after which get a low yield of groats and flakes. Almost half of all refined products consists waste products (up 46 %). In modern conditions carrying out such a complex technological process with low yield of finished products is unprofitable.

References

1. Pomeranz, Y., & Shands, H. L. (1974). Food uses of barley. *Critical Reviews in Food Science & Nutrition*, 4(3), 377-394.
2. Kroshko, GD (1998). Rules for organizing and maintaining the technological process at the mill factories. K. : WIPOL.
3. Eliasson, A. C. (2006). *Carbohydrates in food*. CRC press.
4. Chu, Y. (Ed.). (2013). *Oats nutrition and technology*. John Wiley & Sons.
5. Hamaker, B. R. (Ed.). (2007). *Technology of functional cereal products*. Elsevier.
6. Sots, S.M. Evaluation of some technological factors of Ukrainian naked oats and barley affecting on its processing to food products - <http://pscipub.com/Journals/Data/JList/Scientia%20Agricolturaerae/2014/Volume%208/Issue%201/8.pdf>
7. Henry, R., & Kettlewell, P. (Eds.). (2012). *Cereal grain quality*. Springer Science & Business Media.

GRAPHIC PROCESSORS PROGRAMMING WITH MC# LANGUAGE

Teslenko Denys

Student of Kharkiv National University of Radio Electronics

The parallel programming language MC# is an extension of the object-oriented language C# and is designed to develop applications that run both on multi-core processors and on computing systems with distributed memory (clusters). In particular, he supports the development of applications for hybrid systems based on the GPU (graphical processor unit) – systems that include the main processor and accelerators based on Nvidia's GPUs. At the same time, application development is carried out exclusively using specific MC# language tools and does not require the use of additional tools or libraries, such as OpenMP, MPI, OpenCL, etc.

The extension of the MC# language to support the programming of graphic processors lies within the framework of a single parallel programming model adopted in this language. In particular, async methods that are designed to be executed on separate cores of a multicore processor, and movable methods that are designed to be executed on separate cluster nodes. The so-called GPU methods are added to the MC# language extension for GPUs – methods that are designed for execution on a GPU.

The general ideology of programming GPUs in the MC# language coincides with the ideology of the CUDA technol-

ogy, knowledge of which is assumed for programming GPUs in the MC# language. In particular, before calling the GPU method, the programmer, in accordance with the CUDA technology, must determine the configuration parameters of the GPU – specify the number of the graphic device, the size of the lattice and blocks of computational flows, etc. In programs in the MC# language, this is realized by creating an instance of an object of the GpuConfig class and setting its properties. An integral part of the MC# programming system, which is responsible for supporting GPUs, is the GPU.NET library, which is fully implemented in C# and includes:

- (1) JIT compiler for Nvidia GPUs,
- (2) A set of functions corresponding to the basic functions of the CUDA library.

Also, the MC# language compiler is responsible for supporting the GPU, which recognizes GPU methods in the program text and generates the code which is necessary to call the corresponding functions directly on the GPU.

Using the MC# language for programming GPUs greatly simplifies their use compared to the application of the basic CUDA technology. In particular, when programming in the MC# language, the programmer is freed from the need to ex-

PLICITLY program the data transfer from the main memory to the graphics device memory and vice versa – this task is solved by the MC# language compiler, which generates calls to the corresponding GPU.NET library functions that implement data copying. A similar mechanism, called Unified Memory, appeared only in the latest versions of the CUDA library.

SINCE all the components of the MC# programming system, including components supporting the GPU, are written in C#, the MC# programs can be run on both Windows and Linux, where, in the latter case, as an implementation of the platform. NET uses the freely available Mono software.

THE MC# programming system can be integrated into the Microsoft Visual Studio development system, which allows developing and executing MC# programs for the GPU as part of the latter. At the same time, as in the case of Windows, and Linux, it is assumed that the machine has an installed CUDA system.

THE GPUS supported in the MC# programming system are all types of Nvidia GPUs, including the latest Kepler K20 and K40 models.

An example of programming a GPU in MC#

THE basic structure of the program in the MC# language, intended for execution on the GPU, consists of:

(1) A description of the configuration of the GPU, in which, in particular, the number of (parallel) threads running on the GPU and the parameters for combining them into blocks and a grid are specified, and

(2) GPU-function (method), which will be executed within the framework of one computational thread on the GPU.

THE GPU method in the program is set by assigning it to the definition of the GPU modifier syntactically located in place of the return value type (see the example of the `vecadd` GPU function below). GPU configuration is determined by creating an object of the `GpuConfig` class and setting its parameters. Below is the full text of a simple MC# program designed to add two integer vectors using a GPU.

```

1 using System;
2 using GpuDotNet.Cuda;
3 public static class VectorAddition {
4     public static void Main ( String[] args )
5     {
6         int N = Convert.ToInt32 ( args [ 0 ] );
7         Console.WriteLine ( "N=" + N );
8         int[] A = new int [ N ];
9         int[] B = new int [ N ];
10        int[] C = new int [ N ];
11        for ( int i = 0; i < N; i++ ) {
12            A [ i ] = i;
13            B [ i ] = i + 1;
14        }
15        GpuConfig gpuconfig = new GpuConfig();
16        gpuconfig.SetBlockSize ( N );
17        gpuconfig.vecadd ( A, B, C );
18        for ( int i = 0; i < N; i++ )
19            Console.WriteLine ( C [ i ] );
20    }
21    public static gpu vecadd ( int[] A, int[] B, int[] C ) {
22        int i = ThreadIndex.X;
23        C [ i ] = A [ i ] + B [ i ];
24    }
25 }

```

IN this program, the initial vectors A and B, as well as the resulting vector C, have a length of N. This number is the size of the one-dimensional block of threads launched on the GPU, respectively, the *i*-th thread performs the addition of the *i*-th components A [i]

and B [i] of the original vectors. It is also assumed in this program that the length of the vectors N does not exceed the size of the threads block allowed for a particular graphic device on which the execution of this program is supposed.

Using this program as an example, we note some key features of MC# programs designed for execution on the GPU, which will be detailed in the following sections:

(1) For execution on a GPU, the use of the GPU.NET library included in the MC# programming system is required. The link to this library is set using the using statement:

using GpuDotNet.Cuda;

(2) The main methods setting the configuration parameters of the GPU are:

- SetDeviceNumber (graphic device number),
- SetGridSize (dimensions of the flow lattice),
- SetBlockSize (threads block sizes).

Some of the configuration parameters of the GPU have default values (for example, the number of the graphic device on the machine is 0 by default), and so the call to some configuration methods may be omitted.

(3) Gpu-must be written in accordance with the ideology of CUDA. In particular, it assumes the use of specific CUDA tools, such as ThreadIndex, BlockIndex, BlockSize, GridSize, SyncThreads, GetClock, etc.

(4) The GPU method is considered as an extension method (in C# terminology) of the GpuConfig class, and therefore it is called relative to some object of this class. In accordance with the limitations of C#, extension methods can only be called from static classes, and therefore GPU methods can only be located in classes declared by the user using the static modifier.

(5) The GPU method itself must also be declared static. as well as all functions

called from it (to which the GPU modifier is no longer applied).

(6) Since the current generation GPUs have their own memory different from the main processor memory, all arrays that are arguments of the GPU method are implicitly copied from the main memory to the GPU memory before the GPU method is called and copied back to main memory after the completion of this method.

(7) In the current implementation of MC#, a call to the GPU method is synchronous (unlike async and movable methods), i.e., the execution of the main computational thread from which the GPU method was called is blocked until until this method finishes its work on the GPU.

Conclusion

Modern supercomputers are no longer imaginable without universal accelerators like GPU or Xeon Phi. The specific architecture of the graphics processors, in particular, fast local memory, allows you to run on them some applications ten times faster than conventional processors. But the same specificity of the GPU architecture has complicated programming for them several times. Therefore, at the moment, research is very relevant, aimed at both improving existing GPU programming tools and searching for new paradigms for their use. Within the framework of the MC# project for GPUs, both types of such tasks are solved with the aim of making the MC# language a practical means of solving real problems on GPUs in areas such as image processing, financial mathematics, machine learning, etc.

References

1. D. Negrut, R. Serban, A. Li, A. Seidell, "Unified Memory in CUDA 6: A Brief Overview", Dr. Dobb's Journal, 2014, Sept., URL <http://www.drdobbs.com/parallel/unified-memory-in-cuda-6a-brief-overview/240169095> t 5.
2. R. Farber, "Easy GPU Parallellism with OpenACC", Dr. Dobb's Journal, 2012, June, URL <http://www.drdobbs.com/parallel/easy-gpu-parallelism-with-openacc/240001776> t 14.
3. Петров А.В., Сердюк Ю.П., «Система параллельного программирования МС#2.0», Вычислительные методы и программирование, 9 (2008).

COMPUTER MODELING OF FINANCIAL MARKETS THROUGH CAPITAL ASSET PRICING MODEL

Liutova K., Khovrat Artem

Students of Kharkiv National University of Radio Electronics
scientific director *Aleksey Nazarov*

Key words: computer simulation; financial model; mathematics; CAPM.

The modern financial market is characterized by significant complexity of the processes occurring in it. Risks are increasing, global markets are globalizing, volatility of currencies, interest rates, securities and commodity prices is increasing, and as a result, financial markets are becoming more unstable, complex, risky and unregulated.

Therefore, an important task for both the company and investors is to assess the cost of capital. But it is associated with many problems: often the result is subjective and therefore cannot be considered as a reliable guide.

In order to determine the return on investment of capital, analysts resort to the use of various **forecasting models**. However, standard modeling methods for analyzing processes occurring in financial markets, under the conditions described above, often give unsatisfactory results. This is caused by the gap between real economic realities and economic theory. One of the models that currently helps in solving the problem is the CAPM model. It is she who will be considered in this article.

The essence of the model.

The long-term asset valuation model (CAPM from English 'capital asset pricing

model' or the Russian abbreviation *МОЦА*) was developed by Harry Markowitz in the 50-s [1]. It is an idealized depiction of how financial markets value securities and thereby determine the expected return on investment. The meaning of this model is to demonstrate the close relationship between the rate of return and the risk of a financial instrument, translating it into an estimate of the expected return on equity. This model is based on several neoclassical economic principles as assumptions:

- Securities markets are very efficient and competitive.;
- Rational investors dominate.

Although in modern realities the postulates of the neoclassical school, especially rationality, lend themselves to widespread criticism, this model is still actively applied and works. Most likely this is due to the fact that the amount of transactions in the securities market is quite large, as is the general understanding of economic and financial theory among investors, which causes rational behavior.

The CAPM model is based on the following statement: the greater the risk, the greater the return. Thus, if we know the potential risk of a security, we can predict the rate of return. And vice versa,

```
def linear_regression(y, X):
    from numpy.linalg import inv
    X = np.array(X)
    y = np.array(y)
    if X.shape[0] != y.shape[0]:
        raise ValueError("Неподходящие размерности!")

    # оценка параметров модели
    XTX = np.dot(X.T, X)
    XTy = np.dot(X.T, y)
    w = np.dot(inv(XTX), XTy)

    # коэффициент детерминации
    y_pred = np.dot(X, w)
    R2 = ((y - y_pred)**2).sum() / ((y-y.mean())**2).sum()

    return w, R2
```

«Linear regression» picture 1

if we know the profitability, then we can calculate the risk. All calculations of this kind regarding profitability and risk are carried out precisely with the help of CAPM.

Capital asset pricing model in details.

The relationship of risk with profitability according to the valuation model of long-term assets is described as follows:

where: R – expected rate of return, R_{wr} – risk-free income, R_m – overall market profitability, β , α – special model coefficients, ε – random error [2].

Now let's look at each component of the formula separately, to clarify the subtleties in their definition, in addition to the random error, which is the standard for such models.

Expected rate of return.

This is the profit that you (or another investor) expect from a financial instrument. The expected rate of return may also describe your needs, not expectations. When you need a certain rate of return (the reasons may be different), then this rate will be your expected rate.

In fact, this indicator can be considered profit from a particular instrument.

Risk-free income.

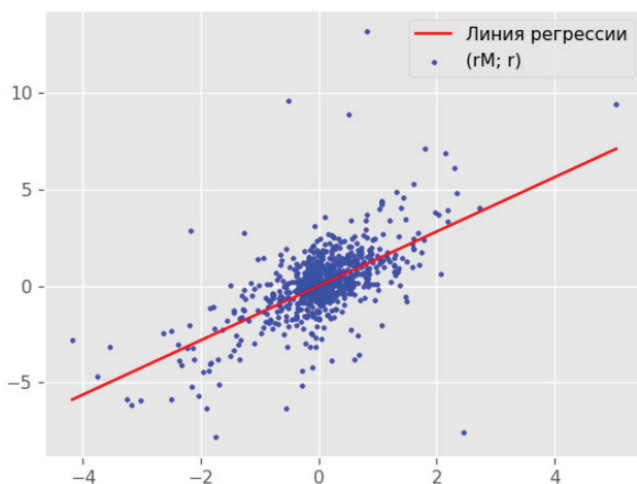
This is the part of the income that is embedded in all investment instruments. Risk-free income is usually measured at the rates of government bonds, since they practically do not contain risk. In the United States or countries of the European Union, this income is approximately 4-5%, for Ukraine this indicator ranges from 10%.

Total market profitability.

This is the rate of return for this market index. If we consider the example of the United States, then the S&P 500 index would be such, for Ukraine it is advisable to consider the PFTS index.

Special Odds.

One of the most important and difficult from the point of view of determining the parameters of the system are the coefficients α и β – previously called special factors that are responsible for measuring the risk of an instrument. These parameters can be found using linear re-



«Graphic illustration of the model» **picture 2**

gression. The following formula for estimating coefficients written in linear-algebraic form is used in the framework of this article:

∂z : \mathbf{w} – vector with target coefficients, \mathbf{X} – vector containing the values of the coefficients for α and β ; \mathbf{y} – dependent vector.

This way vector \mathbf{X} consists of two elements: 1 and $R_m - R_{wr}$. And vector \mathbf{y} of one: $R - R_{wr}$.

It is also important to analyze the determination coefficient:

Based on it, one can judge the adequacy of the results.

Example.

As an example of application and to test the performance of CAPM, consider Amazon shares (they are traded under the AMZN ticker), and take the S&P 500 as a market, as previously determined, traded under the SPY ticker. For a risk-free rate, take the yield on 3-month US Treasury bonds (have a ticker \wedge IRX).

Computer model.

The approach of transferring a mathematical model in symbolic form to a computer representation has recently become very relevant. Especially for the financial sector. Thus, it is possible to significantly simplify and speed up the process of finding the necessary parameters. That is why it was decided to use computer simulation for the task described above.

A suitable tool for this process can be considered the Python programming language. It allows you to quickly create and research models using a small number of libraries.: numpy, pandas, pandas_datareader. And also easily visualize them using tools of matplotlib.

Initially, to complete the task it was necessary to write a function that implements the linear regression indicated above (pict.1).

In the future, it is necessary to obtain data on the rate of return of the market

as a whole, Amazon and three-month government bonds. This process was carried out using the function `get_data_yahoo` from library `pandas_datareader`. She receives the necessary data for a specific ticker and a specified period (within the framework of this article, the period from 01/01/2016 to 12/31/2018 is considered).

After that, variables were formed X and Y according to the principle described above.

As a result of the program, the following values were obtained:

The value of the coefficient of determination indicates the adequacy of the data about special coefficients α and β . You can see the linear relationship between $R - R_{wr}$ and $R_m - R_{wr}$ (pict.2). Therefore, we can con-

clude that for the selected period of time, CAPM worked for Amazon.

Conclusion.

As part of this article, it was shown that the long-term asset valuation model (CAPM) can help determine the selection of shares in your investment portfolio, demonstrating a direct relationship between the risk of a security and its profitability. Using Amazon as an example, from 01/01/2016 to 12/31/2018, the application of this model was considered and the possibility of its use in the real world was confirmed.

Bibliography:

1. <https://ru.wikipedia.org/wiki/CAPM> (Дата обращения 12.11.2019)
2. <http://berg.com.ua/fundam/capm/> (Дата обращения 12.11.2019)

PRINCIPLE OF WORKING OF MUSIC STREAMING SERVICES

Tynianov A. D.

Student of the Department of Computer Engineering Kharkov National University of Radio Electronics

For the last ten years, music industry has changed a lot. CD/DVD discs has been gone, now no one is willing to buy a new album of favorite artist. Today, everybody uses music streaming services.

Leaders in music streaming services industry for today:

1. Spotify
2. Apple music
3. Amazon Music Unlimited
4. Google Play Music
5. Pandora

In this article will be explained how music streaming services work and how they make money from the example of world leader Spotify.

How Spotify makes money?

Spotify uses freemium business model. Company has two major source of income: paid subscriptions and advertisement [2].

There are 4 type of paid subscriptions:

1. Spotify free version. Free version of application with lots of limitation and in-app ads.

2. Spotify premium. For \$9.99 per month users get all features unavailable in free version, namely: ad free, listen offline shuffle play, unlimited skips, high audio quality and many other useful features.

Spotify family premium. Subscription plan special for families. Premium version of application for up to 6 persons

living at the same address. Cost – \$14.99 per month.

Spotify Premium for students. Spotify offers special offer for students – 50% discount for Premium subscription. Price – \$4.99 per month.

Advertisement is the second source of income. Spotify supports several types of ads: sponsored playlists, branded moments, sponsored sessions, audio and video takeovers, display, homepage takeovers, overlay, advertiser pages and branded playlists[3].

Spotify revenue for Q4 of 2018 was approximately \$1.5 billion. Of which, approximately 1.2 billion are obtained through the purchase of a premium subscription, and the rest through advertising, which proves that the main source of earnings of the company is the sale of a premium subscription[1, p.141].

How do musicians get paid from Spotify?

Why it's profitable for musician to work with Spotify? That's because the service pays artists handsomely. Every time, when someone listen to song, artist get paid. Royalty is individual for every artist and can be calculate using this equation:

$$\text{Artist payout} = \text{Spotify monthly revenue} \times \frac{\text{Artist's spotify streams}}{\text{total Spotify streams}} \times \sim 70\% \times \text{artist royalty rate} \quad (1)$$

Spotify takes 30% of an artist's revenue as a commission, and other 70% goes to label or rightsholder. Artists royalty rate is individual for each musician [1, p.210].

Based of the official reports, Spotify pays from \$0.0065 to \$0.0085 per one stream. Also, in February 2018, Spotify announced that it had paid approximately \$10 billions to artists, labels and rights-holders [1, p. 98].

So, it's very profitable for musicians to work with Spotify, not only because of big payouts, but also because it allows you to reach a much larger audience of listeners than, say, the sale of CD / DVD discs, thereby increasing profits. Other

streaming services have the same business model, but Spotify has remained the leader in the music industry for many years.

References

1. Spotify Teardown Inside the Black Box of Streaming Music / Maria Eriksson, Rasmus Fleischer, Anna Johansson, Pelle Snickars and Patrick Vonderau, London: 2019. 288 c.
2. Spotify Usage and Revenue Statistics (2018) // Business of App URL: <http://www.businessofapps.com/data/spotify-statistics/#3>
3. Spotify Business Model | How Does Spotify Make Money? // Feedough URL: <https://www.feedough.com/how-does-spotify-make-money/>

UDC: 937:635.64+632.2.7.78(54U)

ARTIFICIAL DIETS OF THE BRACONIDAE FAMILY REPRESENTATIVES BY REPRODUCING IN VITRO METHOD

Xojimurod Kimsanboyev, Atham Rustamov.

Tashkent State Agrarian University

Abstract: There is a huge role of bracon entomophagy in preventing plants biologically in our Republic. It is possible to achieve biologic efficiency which is higher than 90 % in struggling against cotton – plant moth with this parasite. However while developing them in the condition of laboratory a lot of issues are occurring. In order to populate bracon first of all wax moth is demanded to reproduce. Wheat flour, corn, sugar, margarine, honey productions are used very much, as well as excessively extra workforce is spent for this. Besides these, it takes 55–60 days to reproduce only a generation of a bracon.

Key words: bracon, entomophagy, in vitro, egg, larvae, pupa, imago.

In all over the countries only in China Li Li-Ying who is the professor at the institute of Guanchjou etymology and her apprentice Xie Zhong – neng made researches successfully by bracon *in vitro* way. Special two types of *Bracon hebetor* Say and *Bracon juglandis* Ashm parasites were selected to reproduce *Braconidae* family representatives *in vitro*. In biolaboratory it is required to populate in some equal quantity wax moth and mill flour moth in order to reproduce *in vitro* these parasites at any time of the year.

The main 3 nutritional components were prepared to reproduce *Bracon hebetor* Say's type *in vitro*. Then all the components of the diet are mixed by sentrafuga at 2000 sec. speed for 5 minutes and be transformed, excess oil and foam are removed. Before putting into diet, artificially made maggots that was prepared in advance, they are placed under the ultraviolet lamp for 15 minutes and checked carefully. Afterwards one side of the artificial

worms is thoroughly attached and with the help of syringe ready diet is put by two open sides and after second side has also been attached, it will be put into test-tube. Ready artificial maggots are preplaced in special thermostat at 26–28°C and 75–85 moisture is necessary. After passing 2–3 hours artificial maggots are checked one more time whether they are out of the diet or not. If such kind of situation occurs, they will be taken away. Then female genders of nutrient bracon that flew a day ago is replaced flying at 1:1 proportion. In order to select the most effective diet 5 types of the diet is prepared. 3 of them are chosen as the most effective in reproducing bracon parasite. Their ingredients are following;

Preparation of synthetic diets:

Wax or mill-flour's moth pupa hymolymphasis;

10 % dry cow's milk, as 1gr 10 ml dissolved in distilled water;

Chicken egg yolk.

Table 3.

Indicators of *Bracon hebetor* Say development in artificial diet at differently consumptional rates. (Laboratory experiments in percentage, 2019 y).

№	Consumption rates of the components of the diet, % egg			The degree of damaging with the descendants of the bracon, % larvae	Survival of various stages of development of the bracon generation, by days				Sex ratio (♂:♀) (in one artificial maggot)
					pupa	imago			
1	A			73,5	1,9+0.02	5,3+0.05	4,2+0.03	6,4	12:18
	A 1	A 2	A 3						
	45.+0.07	35+0.03	20+0.05						
2	B			85,8	2,2+0.03	5,1+0.09	4,4+0.04	8.5	6:24
	B 1	B 2	B 3						
	52+0.05	30+0.08	18+0.03						
3	C			92,5	2,3+0.04	5,4+0.07	4,8+0.05	4,3	14:16
	C 1	C 2	C 3						
	55+0.09	25+0.05	20+0.04						
C (Control)				98,5	2.2	5.1	5.3	12.5	3:7
Cotton-plant wormand waxmoth's maggots									

By the first diet (A) wax moth hemolymph (A₁) 45 %, chicken yolk (A₃) 35 %, natural milk (A₄) 20 %. By the second diet (B) wax moth hemolymph (B₁) 52 %, chicken yolk (B₃) 30 %, natural milk (B₄) 18 %. By the third diet (C) cotton-plant pupa hemolymph (C₁) 55 %, chicken yolk (C₃) 25 %, natural milk (C₄) 20 %. (table 3).

The type of *Bracon hebetor* Say is used to damage diets with bracon, this type is considered as a resistant towards extreme conditions. In the most convenient condition is separated for each of

the species to develop by damaging diets with bracon and in this manner thermostat is put. Hurting types of bracon with diets should consist of 26-28 °C, 75-85 % in relative humidity. It was hurted with bracon generations cotton-plant's pupa and wax moth maggots at the age of 4-5 in order to compare between bracon generations and maggots in all diets. Above mentioned versions have been done repeatedly more than thousand times for 3 years and improved artificially-created maggots as well as selected efficient diet.

According to it on the first diet (A) hurting degree with bracon generations is about 73,5 %, it took 11,4 days period of time from eggs to imago. The survival of imagos is approximately 6,4 days. This was the time it took for the larvae to emerge from eggs was 1,9 days, the larval period was 5,6 days, pupa's period was 5,2 days. The sexes which flew out of diet were at the number of proportion for male:females (♂: ♀) (developed in one artificial maggot) 12:18. To the next diet (B) the degree of damaging with the bracon generations was 92,15%. It took 2,2 days to emerge larvae from eggs, in diet larvae developed till 5,3 days then changed into pupa. It took approximately 11,7 days period of time that was spent from egg to imago. Pupa's period continued 5,5 days, the sexes of bracons which flew out artificial maggots were male: females proportion (♂: ♀) (developed in one artificial maggot) was 6:24.

The survival of imagos included about days.

Used literatures

1. B.A. Sulaymonov, X.X. Kimsanboev, R.A. Jumaev, A.A. Rustamov, A.R. Anorbaev, O.A. Sulaymonov. Preventing plants biologically (manual) T: «Uzbekistan» NMIU, 2015. 45p
2. X.X. Kimsanboev, B.A. Sulaymonov, A.R. Anorbaev, U.D. Ortikov, R.A. Jumaev, O.A. Sulaymonov. Development of the plant pests of parasite entomophages in agrobiocenosis. Tashkent "Uzbekistan" 2016 year -235 p
3. M.T. Arslonov, A.U. Sagdullaev, K. Xalilov. Preventing agricultural plants biologically. Tashkent-2010 P-80.
4. B.A. Sulaymonov, X.X. Kimsanboev, A.R. Anorbaev, R.A. Jumaev, A.A. Rustamov, S.K. Sobirov, Sh. SH. Bolki-boev. Phytophagous species in vegetable agro biocenosis and manage their quantity. Tashkent "Uzbekistan" 2018 year -34p
5. B.A. Sulaymonov, X.X. Kimsanboev, Sh.E. Esonboev. Orchard pests and using biological technique against them. T.: Extremum press, 2015.-144 p.

CHALLENGES OF PARALLEL PROGRAMMING

Kyrylenko Olena

Student of Kharkiv National University of Radioelectronics

Key words: complexity, MPI, multicomputers, parallel programming.

Parallel programming has been developing for more than 50 years (for computer standards it is quite a long period) and at the moment it is almost a mass profession. Moreover, very rapid development of various electronic technologies caused appearance of a huge number of commercial supercomputers on the market that are successfully used to solve many problems in science (for example, mathematical modeling) and industry (bank tasks, complex technical devices managing, etc.). As for a single processor computers, they have undergone rather greater changes than less, so their functional devices have started using parallelism in one way or another a long time ago.

Till these days a large number of parallel programming problems have not found a good solution in modern theories, languages, systems and technologies of parallel programming. For this reason, parallel implementation of the algorithms encounters significant difficulties. The development, debugging and maintenance of parallel programs are very difficult. There are common situations when a long and correctly working program suddenly gives erroneous results. Of course, such situations are present in sequential programming. However, in parallel programming, such problems are exacerbated by the additional need to program and control in-

terprocess communications correctly (pithy and in time), which is very difficult for a person to do (like any other actions in time). A parallel program is a system of interacting processes without rigidly fixed execution order, which is the reason of the non-determinism of calculations, which appears, for example, when at each program execution, its processes are executed, if summarized, in a different order and because of this, if an error occurs, it is difficult to repeat the error situation for its localization. Each operation of the wrong working program may produce a different result. It is extremely difficult to detect such errors. Therefore, in parallel programming languages, it is reasonable to try to prevent a large non-determinism or limit it with some strictly defined frameworks [1].

Parallel programs must have special dynamic properties (customizability for all available resources, portability in the multicomputer class, dynamic balance of loading, dynamic accounting of task properties), which are not relevant for sequential programs. As a result, the current system-wide software and programming systems turned out to be poorly suitable for creating parallel programs, and the complexity of parallel programming appeared more than enormous. When it comes to programs with dynamic properties, it is not a secret that

they can be debugged indefinitely. Definitely there are a considerable number of tasks that are relatively simple to program for execution on multicomputers, but still it's quite difficult to parallelize and program most of them. In addition, as practice shows, there is a quite small number of ready-made algorithms that will be suitable for a good parallel realization [2]. All of the above makes parallel programming an occupation only for people who are well educated in mathematics (especially discrete one) and properly trained in the necessary technologies, who also have knowledge of operating systems, programming systems, the architecture of modern processors, computers and multicomputers, and the ability to program.

At the present, parallel programming systems based on MPI (Message Passing Interface) are the most common. The idea of MPI is initially simple and obvious. It assumes a parallel program presentation in the form of many parallel processes (process programs are usually developed in the usual sequential programming language like C or Fortran) that interact with each other during execution to transfer data using the communication procedures, which make up the MPI library. Nevertheless, the proper implementation of MPI to provide inter-processor communications proved to be enough difficult. Such complexity is associated with the need to achieve high performance of programs, the need to use numerous multicomputer resources and, as a result, a wide variety of the algorithms for the implementation of

communication procedures depending on the data processing mode in the processor elements of the multicomputer. At the same time, data transfers between program processes should be carried out regardless of where and on which processors the interacting processes are actually located. Another source of complexity is the consideration of the data transfer between processes features in the implementation of communications. The better features of the equipment and the data processing mode in the processor elements are taken into account the better parallel program is. The less accounting is, the more general algorithms are used to implement communications and so the worse the quality of data transmission by time [3]. Programming using MPI turned out to be a deceptively simple thing. It is deceptive, because there are examples of elementary obvious parallel programs consisting of literally two to three dozens operators that contain hard-to-detect errors. And worst of all, these programs work correctly or plausibly (especially during tests), so errors are detected, as a rule, at the most inopportune moment.

In total, the most important thing is to remember that mistakes made in the early stages of creating parallel programs are irreparable in the subsequent ones. For example, a bad chosen algorithm for solving a problem cannot be fixed with good programming and, in particular, a sequential algorithm cannot be executed in parallel. Thus, maximum efforts to solve the problem should be applied at the very beginning. Besides, it is neces-

sary to bear in mind that unsuccessful programming of parallel executable algorithm can repeatedly worsen the quality of the solution of a problem compared to an average good program. Therefore, knowledge of parallel programming methods and understanding of the problem are so important.

Literature:

1. Korneev V.V. Parallel computing systems. – M., 1999. -p. 312.
2. V.E. Malyshkin, V. D. Korneev. Parallel programming of multicomputers. – Novosibirsk, 2006. -p. 439.
3. G.R. Andrews. Basics of multi-threaded parallel and distributed programming. – M.: Ed. Williams House, 2003. -p. 330.

LOGICS OF MULTI-STAGE DECISION-MAKING FOR ROBOTS

Kuchkov Boryslav

Student of the Department of Computer Engineering and Management Kharkov National University of Radio Electronics

Introduction and Problem statement

Deep Learning techniques have recently enabled researchers to successfully tackle low-level perception problems in a supervised learning fashion. In the field of Reinforcement Learning this has transferred into the ability to develop agents able to learn to act in high-dimensional input spaces. In particular, deep neural networks have been used to help reinforcement learning scale to environments with visual inputs, allowing them to learn policies in testbeds that previously were completely intractable. For instance, algorithms such as Deep Q-Network (DQN) have been shown to reach human-level performances on most of the classic ATARI 2600 games by learning a controller directly from raw pixels, and without any additional supervision beside the score. Most of the work spawned in this new area has however tackled environments where the state is fully observable, the reward function has no or low delay, and the action set is relatively small. To solve the great majority of real life problems agents must instead be able to handle partial observability, structured and complex dynamics, and noisy and high-dimensional control interfaces.

Google's Artificial Intelligence research group, DeepMind recently released a python API, pySC2 for the popular Real Time Strategy (RTS) computer game,

StarCraftII [1, 382]. After successfully conquering the boardgame, Go, with their AlphaGo program, DeepMind has set their sights on the next big challenge for AI systems in attempting to train these systems to learn how to compete with world champions in the incredibly complex world of StarCraft.

Real time strategies problem

Real-time strategy (RTS) games have historically been a domain of interest of the planning and decision making research communities. This type of games aims to simulate the control of multiple units in a military setting at different scales and level of complexity, usually in a fixed-size 2D map, in duel or in small teams. The goal of the player is to collect resources which can be used to expand their control on the map, create buildings and units to fight off enemy deployments, and ultimately destroy the opponents [1, 384]. One of the reasons the game is so hard is that playing it well requires thinking and acting on different levels of abstraction. The game requires resource collection management, build order scheduling, prioritizing technology development, exploration, micro-management of troops as well as overall strategy and ways of deducing and countering the adversary's strategy. Trying to build an AI that can do all this well is very very hard.

Purpose of the work

As current RL algorithms require deep learning with back propagation and SC2 has lots of states that can dramatically the learning is dramatically slowed down. It is needed to speed up existing algorithms to use with less powerful computers leveraging reasonable amount of resources. I tried to reproduce Deepmind's results on more limited hardware.

Learning environment

During the research PYSC2 environment was used. The StarCraft II API allows programmatic control of StarCraft II. The API can be used to start a game, get observations, take actions, and review replays. This API is provided in a limited headless build that runs on Linux especially for machine learning and distributed use cases. Python wrapper was used along with the environment. PySC2 defines an action and observation specification, that can be used to use as is without processing of game screen. The game outputs are both spatial/visual and structured elements representing user resources, amount of army etc.

Problem solution and Results

This section provides overview of selected agent architectures and comparable results of work.

The learning algorithm used is the same as in the original paper differing with the method of learning parameters. As an alternative to the asynchronous implementation [2, 77], it was implemented a synchronous, deterministic implementation that waits for each actor to finish its segment of experience before

performing an update, averaging over all of the actors. One advantage of this method is that it can more effectively use of GPUs, which perform best with large batch sizes. This algorithm is called A2C [3, 38], short for advantage actor critic.

Agent architecture

The agent has preprocessing of input data. I embed all feature layers containing categorical values into a continuous space, which is equivalent to using a one-hot encoding in the channel dimension followed by a 1×1 convolution. This is done to reduce the dimensionality of input as some features such as unit type can have up to several thousand value options. I also re-scale numerical features with a logarithmic transformation as some of them such as hit-points or minerals might attain substantially high values.

The agent used is FullyConv agent with some improvements in output layers. The network proposed has no stride and uses padding at every layer, thereby preserving the resolution of the spatial information in the input. For simplicity, we assume the screen and minimap inputs have the same resolution [3, 38]. We pass screen and minimap observations through separate 2-layer convolutional networks with 16, 32 filters of size 5×5 , 3×3 respectively. The state representation is then formed by the concatenation of the screen and minimap network outputs, as well as the flat feature vector, along the channel dimension. Flat feature vector includes observations about user state, such as player_id, minerals, vespene, food used (otherwise known as supply), food cap, food used by army, food used by

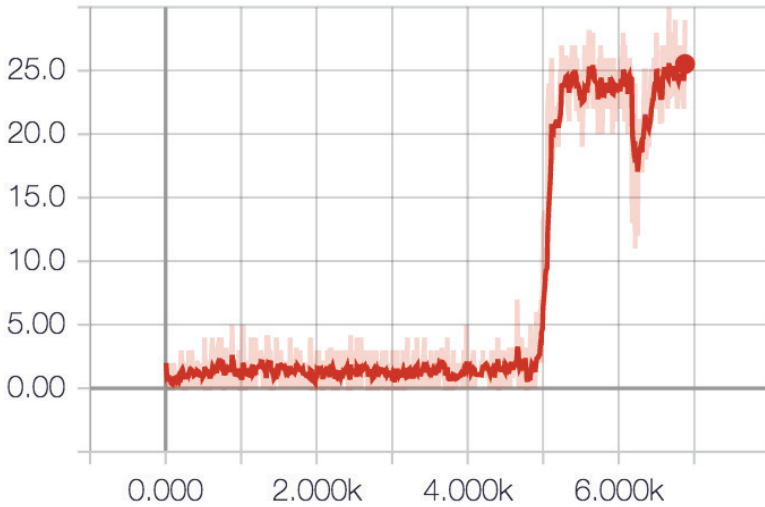


Figure 1. MoveToBeacon mini game rewards over training process. Vertical axis stand for game reward and horizontal one is the number of episodes.

workers. Likewise game steps added to flat features to stimulate agent to perform faster as it takes more time to learn only on sparse rewards. To compute the value and policies over categorical (non-spatial) actions, the state representation is first passed through a fully-connected layer with 256 units and ReLU activations, followed by fully-connected linear layers. These linear layers represent action types and action parameters. For example, `move_camera` action takes minimap point as a parameter while `select_rect` used for army selection requires three parameters: flag showing necessity to add to existing selection and two points on screen. Thus, non-spatial output has multiple layers which are distribution over actions along with distribution over each parameter type options [4, 88]. Finally, a policy over spatial actions, points on map, is obtained using 1×1 convolution of the state representation with a single output

channel.

The approach used to choose action is Boltzmann exploration. Instead of always taking the optimal action, or taking a random action, this approach involves choosing an action with weighted probabilities [4, 90]. This way the agent can ignore actions which it estimates to be largely sub-optimal and give more attention to potentially promising, but not necessarily ideal actions. In line with the human UI, I ensure that unavailable actions are never chosen by an agent. To do so I mask out the function identifier such that only the available subset can be sampled. I implement this by masking out actions and renormalizing the probability distribution available ones.

Training process

DeepMind mean scores are their best individual scores after 100 runs for each game, where the initial learning rate was randomly sampled for each run. I use a

Table 1.
Minimap resulting scores

Map	This agent	Deepmind
MoveToBeacon	26	26
CollectMineralShards	102	103
FindAndDefeatZerglings	43	45
DefeatRoaches	90*	100
DefeatZerglingsAndBanelings	42*	62
CollectMineralsAndGas	3340	3978
BuildMarines	0.55	3

constant initial learning rate for a much smaller number of runs due to limited hardware. All agents use the same Fully-Conv agent. With default settings (32 environments), learning MoveToBeacon well takes between 3K and 8K total episodes. This varies each run depending on random initialization and action sampling. On computer with Nvidia GTX 1060 and Core i7 6700 it takes about 2 hours to train this minimap.

Results

In DefeatRoaches and DefeatZerglingsAndBanelings the result is not stable with standard deviation of 40 and 120 respectively. It took 5 runs to get the score for DefeatRoaches reported here. Also the scores for those are still considerably worse than Deepmind scores. It might be due to no hyperparameter tuning.

Conclusions

To sum up, the work was successful as I managed to reproduce the results of original paper with much less computational resources. Still there is a plenty of

opportunities to improve existing solution. It can be done either with alternative algorithms such as PPO [5, 146], using a specially preprocessed reduced set of features or alternative approaches, such as hierarchical reinforcement learning [7, 67] or Auxiliary Tasks [6, 45].

REFERENCES

1. O. Vinyals, T. Ewalds, S. Bartunov, P. Georgiev, StarCraft II: A New Challenge for Reinforcement Learning, P. 375–397, 2017.
2. V. Mnih. Asynchronous Methods for Deep Reinforcement Learning, P. 76–80, 2016.
3. Y. Wu, E. Mansimov. Scalable trust-region method for deep reinforcement learning using Kronecker-factored approximation, P. 23–56, 2017.
4. L. Metz, J. Ibarz. Discrete Sequential Prediction of Continuous Actions for Deep RL, 2017 P. 88–90.
5. J. Schulman, F. Wolski. Proximal Policy Optimization Algorithms, P. 145–178. 2017.
6. M. Jaderberg, V. Mnih. Reinforcement Learning with Unsupervised Auxiliary Tasks, P. 127–135, 2016.
7. T. D. Kulkarni, K. R. Narasimhan. Hierarchical Deep Reinforcement Learning: Integrating Temporal Abstraction and Intrinsic Motivation, P. 34–87, 2016.

FORECASTING THE MARKET PRICE OF SOFTWARE USING NEURAL NETWORKS

Tykhomyrov Valentyn

Student of the Department of Computer Engineering and Management Kharkiv National University of Radio Electronics

I. Introduction and Problem statement

In today's economic environment, the development and implementation of new technologies is especially important for the successful competition of companies. An important element in making investment decisions for technological projects is to evaluate their effectiveness. As the market of the purchase/sale of new technologies exists and functions, there is a need to determine the value of the development.

Estimating the cost of technology is needed to analyze the profitability of current and future technological projects and the feasibility of investments.

The ability to estimate the potential benefits and losses from the project in the early stages, to analyze possible scenarios of the development of events becomes very important.

Before the formation of valuation models, the cost of the software product was estimated according to the "thumb rule". According to it, the created application was often limited by a certain field of activity, where experts exist [1, 375]. These specialists could tell with some degree of certainty what size of the generated code and its cost will be. Used in most cases, this rule often was not suitable for atypical development projects. Also evaluation by analogy was actively used – the

data was taken from a similar, successfully completed project and shifted to a new project. Of course, the degree of reliability of such assessments was very low. As the number and size of software projects increased, the scope of their valuation also began to expand. Experts working in the field of software project management began to determine the general tendencies and rules of valuation that were actively used in their work. Having accumulated material on a sufficiently large number of projects, they tried to derive formulas for estimating the cost. Of course, the risk of inaccurate calculations remained high, since the limits of the rules could only be determined after a while, with the increase in the number of projects on which these rules were checked [1, 377].

The beginning of intensive studies in the field of cost estimation modeling dates back to 1965-1966, when EA. Nelson, a member of the System Development Corporation (SDC), who carried out a study on calculating the cost of software for the US Air Force, published a work entitled "A Handbook for Managing Cost Estimates for Programming".

Since that time, a lot of models for estimating the cost of software have appeared, including some of the most widely used valuation models – SEER-SEM, SLIM, COCOMO[1, 381].

Nevertheless, it is equally important to be able to estimate the competitive price of a product within the existing software market, that is not allowed in existing evaluation models. For this purpose, the use of assessment methods with the ability to study is the most appropriate – these are updated methods of evaluation by analogy with the subsequent training using artificial intelligence techniques (such as neural networks) [1, 393].

The value of neural network modeling techniques in performing complicated pattern recognition and non-linear estimation tasks has been demonstrated across an impressive spectrum of applications.

This study uses backpropagation artificial neural networks to examine whether they are capable of adequately capturing software cost complexities in their weight space, to enable them to make accurate estimates [1, 396].

II. Problem solution and Results

Data collection

The input for this task is a set of open information about the software of a certain type. Because of the openness criterion, it was decided to choose mobile apps for Android, full information of this type of software is available on an open source Google Play Market. Several API's for access to data were considered:

- Google Play Unofficial Python API (<https://github.com/egirault/googleplay-api>)
- Python Android Market Library (<https://github.com/liato/android-market-api-py>)

- 42matter (<https://42matters.com>)

The third option was the most suitable for this prediction task, because HTTP-request GET that POST of this RESTful API allows to get all the meta-data of any application in a convenient for the next processing format JSON [2, 235].

The following characteristics of the application were chosen as features for the training set: rating, number of ratings, number of downloads, number of reviews, in-app purchases, number of supported languages. The dependent characteristic, which is called the target variable, is the price of the application.

After the data is collected, it must be prepared. This stage is called preprocessing. The main task of preprocessing is to display data in a format suitable for learning the model. For this purpose, unnecessary information was deleted and all data was normalized [2, 26].

Neural network model

For this study, backpropagation artificial neural network models were used. Backpropagation networks are the most generalized neural networks currently in use and this approach was chosen in preference to Hopfield and Kohonen networks. As software development estimation is not a time series problem, approaches such as finite impulse response (FIR) and recurrent networks were not considered [2, 84].

The backpropagation network requires data from which to learn. To learn the network calculates the error, which is the difference between the desired response and the actual response, and a portion of it is propagated backward

through the network [3, 76]. At each neuron in the network the error is used to adjust weights and threshold values of the neuron, so that at the next epoch the error in the network response will be less for the same inputs [3, 78]. This corrective procedure is called backpropagation and is applied continuously for each set of inputs or training data. The training data should consist of as much relevant data as possible. In practice one does not usually have the luxury of a perfect training data set [4, 106].

For this project the data were divided into three sets. The training set, the test set, and the validation set. The data for each category were randomly chosen, except that the data in the test and validation sets was not allowed to be larger or smaller than the largest and smallest features respectively in the training set. This was done so that predictions were not made outside the data range on which the network had been trained.

The inputs were rating, number of ratings, number of downloads, number of reviews, in-app purchases, number of supported languages [4, 108]. The target against which the network was trained was the price of the application in the training set. The accuracy of the price estimate was taken as the Root Mean Square Error (RMSE).

To try and improve the network performance, the learning rate and momentum were varied, as was the network architecture. Models with one through to six hidden layers were developed. Consistently the models with just a single hidden layer performed better, while the

models with multiple hidden layers in many instances did not converge. Various activation functions were tried, and the popular sigmoid function consistently gave the best results [4, 112].

There is no clearly defined theory which allows for the calculation of the ideal parameter settings and as a rule even slight parameter changes can cause major variations in the behavior of almost all networks. It is through a process of trial and error and experience that settings are selected which will result in a reduced average prediction error. The settings of the learning rate and momentum control the way in which the error is used to correct the weights in the neural network for each training case. When the learning rate is set to high values there is the possibility of unstable behavior, as evidenced by widely varying average error values [4, 114]. When the learning rate is set lower, the possibility of unstable behavior is reduced, but training times are increased and there is a greater probability of getting stuck in local error minima. The higher the momentum, the larger the percentage of previous errors that is applied to the weight adjustment in each training case. For example, when the momentum is set at 0.5, then 50 percent of the weight adjustment will be due to the current error and 50 percent will be the weight adjustment applied in the previous case. For this set of data a learning rate of 0.1 and a momentum of 0.7 gave good results [2, 301].

The average training error reduced steadily as the network trains, as is the prediction error. For this network, the

average of the prediction error was obtained at about 2000 iterations. With further training the training error is further reduced, but the network does not generalize well, and from this point the average prediction error increases. The reason for this is that the network tends to curve-fit the training data, giving a low average training error.

Analysis of data

Network models were developed with various combinations of inputs selected from the attributes mentioned above. The results were ambiguous, because prediction errors were erratic. An examination of the results showed that the network overestimate the price of the application with lower rating and worse indicators, as well as underestimate price of the application with higher rating. For the remaining applications, the estimated price was more accurate [2, 328].

III. Conclusions

Within the collected data set, back-propagation artificial neural networks appear to indicate the potential to be developed into good software price estimation models. The model is not difficult to develop and has the flexibility of being able to incorporate additional attributes as input if special circumstances warrant their inclusion [5, 127]. Neural networks have the ability to capture knowledge of the complex interrelationships in their weight matrix to enable them to make predictions. Further research will be conducted

to use larger set of training examples, that covers all possible values of software characteristics. It will allow to make training networks more stable. Also semantic analysis of application reviews is of interest. This type of analysis can provide an opportunity to identify positive and negative references in order to better prioritize applications [5, 127].

1. Dejaeger K., Verbeke W., Martens D., Baesens B. Data Mining Techniques for Software Effort Estimation: A Comparative Study // *IEEE Transactions on Software Engineering*. 2012. Vol. 38, Issue No. 2. P. 375–397. URL: <http://www.computer.org/csdl/trans/ts/2012/02/tts2012020375-abs.html>
2. Mislick G.K., Nussbaum D.A. Cost Estimation: Methods and Tools, 2015. 344 p
3. Subitsha P., Rajan J.K. Artificial Neural Network Models For Software Effort Estimation // *International journal of technology enhancements and emerging engineering research*, 2014. Vol 2, Issue 4. P. 76–80. URL: <http://www.ijteee.org/finalprint/apr2014/Artificial-Neural-Network-Models-For-Software-Effort-Estimation.pdf>
4. Kumar G., Bhatia P.K. Automation of Software Cost Estimation using Neural Network Technique // *International Journal of Computer Applications*. 2014. Vol. 98, No. 20. P. 11–17. URL: <http://research.ijcaonline.org/volume98/number20/pxc3897709.pdf>
5. López-Martín C., Abran A. Neural networks for predicting the duration of new software projects // *The Journal of Systems and Software*. 2015. Vol. 101, Issue, Elsevier Science Inc. New York, NY, USA. P. 127–135.

HIGH LOCAL MICROWAVE DIAGNOSTICS OF MICROELECTRONICS MATERIALS USING SCANNING MICROWAVE MICROSCOPY

Shcherban I.M

PhD student

Kharkiv National University of Radio Electronics

Slusarenko O.A.

student

Kharkiv National University of Radio Electronics

Keywords: Scanning Microwave Microscope, invariance, combined signals, resonance probe, multi-parameter diagnostics.

Diagnostics of the distribution of electrical conductivity over the surface of an object and in depth is one of the primary tasks in material science. One of the most promising tools in this area for determining the profile of σ or $\text{tg}\delta$ is the Scanning Microwave Microscope (SMM). The other more scientifically and technically advanced tools like Scanning probe microscopy, such as the Atomic-Force Microscope (AFM) and the Scanning Tunneling Microscope (STM) do not allow measuring the electrophysical properties of the studied elements. Scanning capacitive microscopy (SCM) also practically does not allow to study the characteristics of semiconductors and dielectrics associated with electrical conductivity, in particular. Since CSM probes the material to the depth of the Debye shielding.

The main distinguishing feature of SMM in comparison with AFM, STM and SCM is multi-parameter diagnostics. Using the aforementioned tool, it becomes possible to obtain information about the surface profile of the studied object, its electrophysical parameters, as well as spe-

cific parameters inherent in materials of different structures, which are derivatives of ϵ and $\text{tg}\delta$ [1, 2].

Diagnostics of the distribution of electrical conductivity or $\text{tg}\delta$ is most in demand in the production of semiconductor microelectronics elements. As it's known, in the manufacture of silicon wafers, the degree of roughness of their surface is quite low. In it's turn, the value of dielectric constant in the volume of this material also does not change. One of the most important parameters of silicon, which can be changed during production, is electrical conductivity. Therefore, the determination of the distribution profile for this parameter is the most necessary one. For these purposes, in the SMM it is preferable to use resonance probes (RP) as the primary measuring transducer, since the Q factor of the resonator or its change strongly depends on the $\text{tg}\delta$ of the material under study [2].

However, as it shown in [2, 3], when measuring fundamental signals in the

SMM $\frac{\Delta Q}{Q_s}$ and $\frac{\Delta f}{f_0}$ depending on σ or $\text{tg}\delta$,

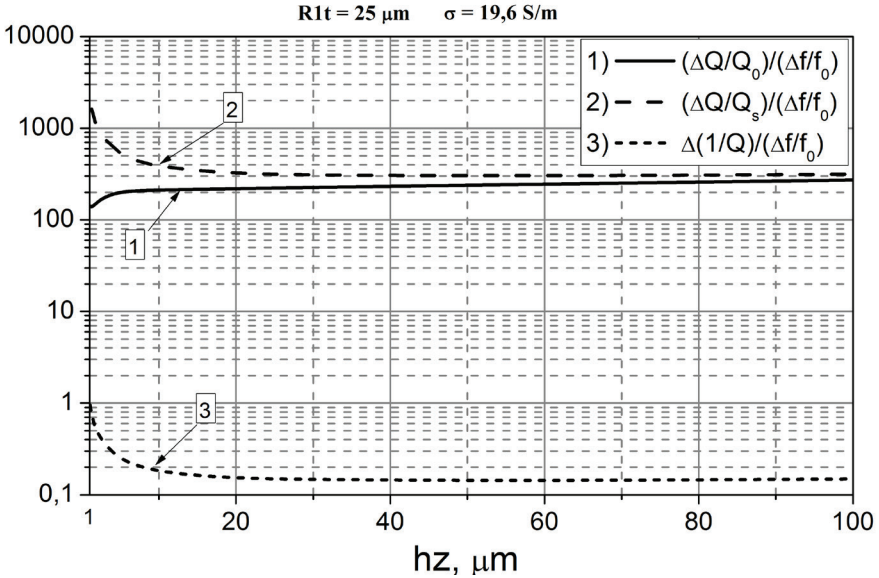


Figure 1 – Invariance of combined signals to the distance between the tip and the sample

the obtained dependences need reconstruction. As it was shown in [3], they are influenced during measurement by such factors as: change in the gap between the probe and the sample hz due to roughness of the surface of the object; the shape of the tip is intermediate between spherical and conical; the error in measuring the radius of the tip $R1t$. All of these factors are interfering. Yet, as shown in [3, 4, 5], taking the ratio of these signals, namely

$$\frac{\Delta Q}{Q_s} / \frac{\Delta f}{f_0},$$

it is possible to obtain a signal that will be invariant to the above mentioned factors.

As shown in Fig. 1, the combined signal may be determined by various rela-

tions. In this case, signals $\frac{\Delta Q}{Q_s} / \frac{\Delta f}{f_0}$ and $\frac{\Delta Q}{Q_0} / \frac{\Delta f}{f_0}$ have a value of the same order, but of a different nature in the region $hz < R1t$.

The amplitude of the combined signal $\Delta(Q_s^{-1}) / \frac{\Delta f}{f_0}$ much lower than $\frac{\Delta Q}{Q_s} / \frac{\Delta f}{f_0}$ and $\frac{\Delta Q}{Q_0} / \frac{\Delta f}{f_0}$. Therefore, the accuracy of

emphasizing this signal in practice will be significantly lower at low $tg\delta$ value of the material. And taking into account that the

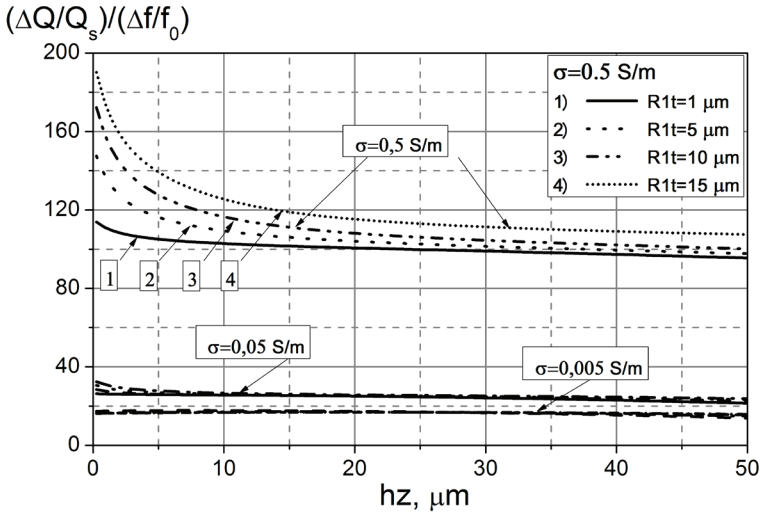


Figure 2 – View of combined signal at spherical shape of the tip

combined signal $\frac{\Delta Q}{Q_0} / \frac{\Delta f}{f_0}$ has a substantially nonlinear character [5] at low $tg\delta$ values, further studies are related only to the signal $\frac{\Delta Q}{Q_s} / \frac{\Delta f}{f_0}$.

In practice, factors of electronic origin are added to interfering factors of an electrodynamic nature. These include changing the parameters of the microwave generator, detector, automatic frequency control system (AFC) or replacing them in case of failure. But, if we take into account that the primary signals of the measurement information Q and $\Delta f/f_0$ in the modulation AFC are emphasized as analog signals at the modulation frequency ($U_{2\Omega M}$) and at the doubled modulation frequency ($U_{\Omega M}$) respectively, then from relation (1) we can obtain a signal that

will be invariant as to interfering factors of electrodynamic origin, and electronic.

$$\Delta\left(\frac{1}{Q}\right) / \frac{\Delta f}{f_0} = \left(\frac{\Delta Q}{Q_0 Q_s} / \frac{\Delta f}{f_0}\right) \times Q_0^2 \quad (1)$$

As can be seen from the figure, the suppression of the influence of such an interfering factor as the gap between the tip and the sample takes place. But, for $tg\delta \geq 0.1$ and $\epsilon \geq 10$, the invariance non-fulfillment region including the zero gap value hz expands with an increase in the tip radius $R1t$.

Thus, we can conclude that only with a tip of radius $R1t = 1 \mu m$ and object parameters $tg\delta \geq 0.1$ and $\epsilon \geq 10$, which ensures a high degree of field evanescence in the object, invariance is observed in the entire studied range of values of hz . In other cases, it can be violated, and will be performed in separate areas of the gap values.

Nevertheless, the implementation of the invariance of combined signals to interfering factors is a rather necessary task

in microwave local diagnostics, since in addition to eliminating the influence of the so-called interfering factors – the prospect of measuring the conductivity profile or $\text{tg}\delta$ of the sample opens up without complicating the task of 2 parameter control.

References

1. Anlage SM Principles of near-field microwave microscopy / Anlage SM, Talanov VV, Schwartz AR // Scanning Probe Microscopy: electrical and electromechanical phenomena at nanoscale – New York: Springer-Verlag, 2007. – Vol.1. – p.215-253.
2. Gordienko Yu. Ye. Scanning microwave microscopy as an intelligent measuring system / Intelligent measuring systems based on new generation microelectronic sensors: a monograph // edited by Ya. I. Lepikh, V.A. Romanov, Odessa: Astroprint, 2011. – P. 176 –246.
3. Gordienko Yu.Ye. Suppressing the effects of interfering factors in local microwave diagnostics / Gordienko Yu.Ye., Shian O.P., Shcherban I.M. // Telecommunication and Radio Engineering, 2016. Vol. 75, № 13. – P. 1221-1229.
4. Shcherban I.M, Gordienko Yu.Ye. Features of the use of microwave microscopy in nanoelectronics technology // X INTERNATIONAL SCIENTIFIC CONFERENCE «Functional Basis of Nanoelectronics», Kharkiv-Odesa, Ukraine, 2019., p. 55-58.
5. Gordienko Yu.Ye. Resonator aperture microwave sensors for small objects properties testing / Gordienko Yu.Ye., Shcheban I.M., Levchenko A.V., Polishchuk A.V., Prokaza A.M. // Telecommunication and Radio Engineering, 2017. Vol. 76, № 18. – P. 1649-1659.

РАЗРАБОТКА ШАГАЮЩЕЙ СЕЯЛКИ ДЛЯ КРУГОВЫХ ЛЕСНЫХ ПИТОМНИКОВ

Бырдина С. С.

магистрант

Братский государственный университет, г. Братск, Россия

Бырдин П. В.

канд. техн. наук, доцент, профессор РАЕ

Братский государственный университет, г. Братск, Россия

Ключевые слова: лесовосстановление, лесной питомник, структура, сеянец, схема выращивания, круговое поле, посадочный радиус.

Keyword: reforestation, forest nursery, structure, seedling, growing scheme, circular field, planting radius.

По способу организации территории лесного питомника и характеру технологического процесса, кроме классических прямоугольных, выделяют круговые и подпологовые лесные питомники. Круговые лесные питомники имеют площадь в виде эллипса или круга, в центре которого и по краям сохраняется естественное насаждение. Такая организация территории позволяет выращивать посадочный материал в условиях более близких к лесной среде и избежать трудоемких работ по отенению посевов.

Подпологовые круговые питомники организуют для выращивания посадочного материала с улучшенной наследственностью, однако имеют не высокий уровень механизации технологических процессов ввиду расположения посевных строк по кругу или эллипсу [1].

Для решения поставленной проблемы и повышения уровня механизации

посева семян в лесных круговых питомниках был проведен анализ существующих сеялок для круговых питомников (Авторское свидетельство СССР №1563612 «Универсальная роторная пропашная широкозахватная сеялка» [2] и Патент РФ на изобретение № 2119278 [3]) и разработана шагающая сеялка для круговых питомников и подана заявка на изобретение.

Согласно Авторскому свидетельству СССР №1563612 рис. 1) универсальная роторная пропашная широкозахватная сеялка состоит из рамы с установленными на ней семенными бункерами, шарнирно соединенных с ними поводков-семяпроводов с семенными камерами, роторов, установленных на полой оси, разделенной на полости повышенного и пониженного давления воздуха.

Роторы снабжены посадочными органами, установленными в цилиндрических втулках с возможностью ради-

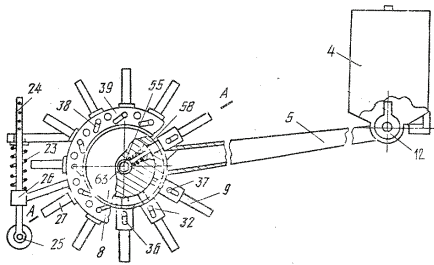


Рисунок 1 – Универсальная роторная пропашная широкозахватная сеялка

ального перемещения. Посадочные органы снабжены подвижными посадочными пальцами. При движении сеялки роторы, взаимодействуя с почвой, перекапываются по ней и посадочные органы заглубляются в почву на заданную глубину заделки семян. Представленная сеялка имеет сложную конструкцию и энергозатратна в эксплуатации.

Шагающая сеялка (Патент РФ №2119278 «Шагающая сеялка», рис. 2) используется для строчно-луночного посева хвойных пород на захламленных порубочными остатками свежих вырубках (лесосеках) без предварительной подготовки почвы во всех лесорастительных зонах.

Она проста в эксплуатации. Ее можно навешивать на все типы тракторов, чтобы она двигалась по следу трактора, что увеличивает ее проходимость. Шагающая сеялка содержит раму, шестеренчатый диск, высевное устройство в виде барабана с семенной камерой, на цилиндрической поверхности которого равномерно в одной вертикальной плоскости закреплены посевные пальцы с радиальным углом между собой 60°. Каждый палец

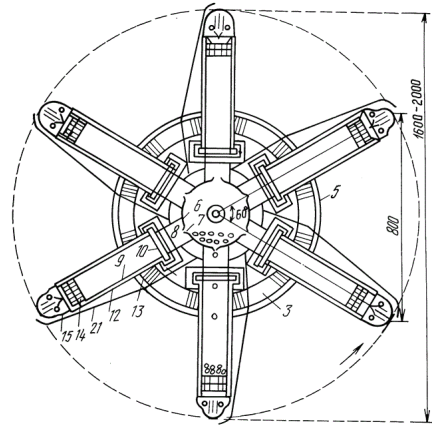


Рисунок 2 – Шагающая сеялка

выполнен в виде двух установленных одна в другую труб. Внутренняя труба закреплена на барабане. Проставленная сеялка не предназначена для работы в организованных лесных питомниках.

Для посева семян в лесных механизированных круговых питомниках была разработана конструкция шагающей сеялки (рис. 3) и подана заявка на патент.

Шагающая сеялка (рис.3) имеет раму 1 с жестко закрепленными на ней стойками 2 в которых на полуосях 3 с возможностью свободного вращения установлены высевные устройства 4.

Высевное устройство 4 (рис. 4) выполнено в виде барабана 5 с семенной камерой, на цилиндрической поверхности которого равномерно в одной вертикальной плоскости через равное расстояние в отверстиях 6 установлены посевные пальцы 7, а на спицах 8 жестко закреплён обод 9 имеющий отверстия 10 под посевные

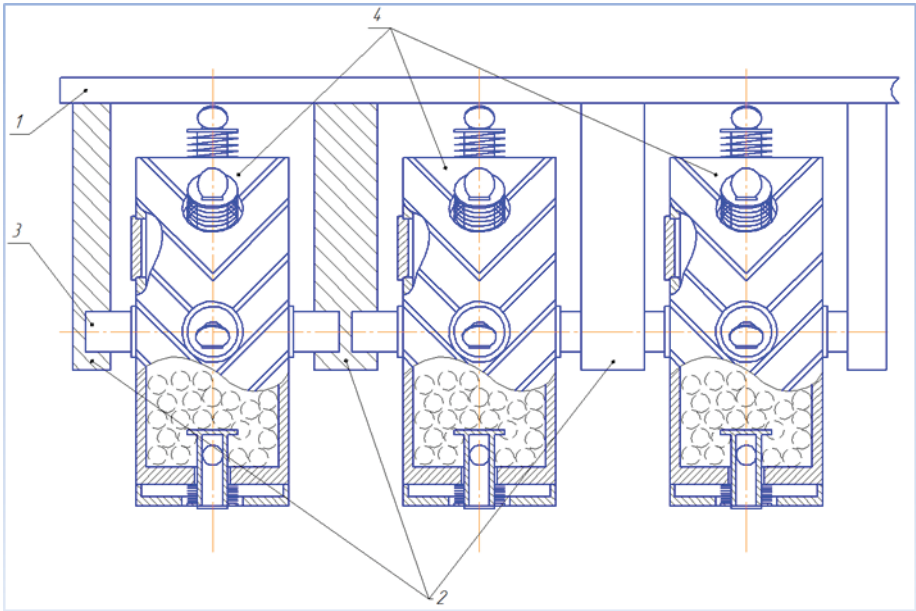


Рисунок 3 – Вид общей шагающей сеялки.

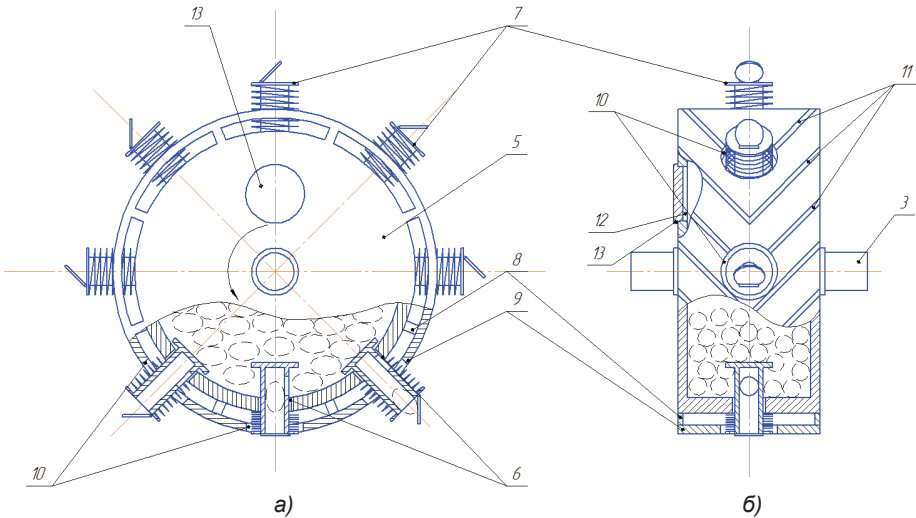


Рисунок 4 – Высевающее устройство: а) главный вид, б) вид сбоку.

пальцы 7 и оснащенный грунтозацепами 11. Высевающее устройство 4 в боковой части оснащено загрузочным отверстием 12 с крышкой 13.

Каждый посевной палец 7 рис. 5) выполнен в виде стакана с упором 14 и отверстием 15 в боковой части и оснащен упорной шайбой 16 с подпружи-

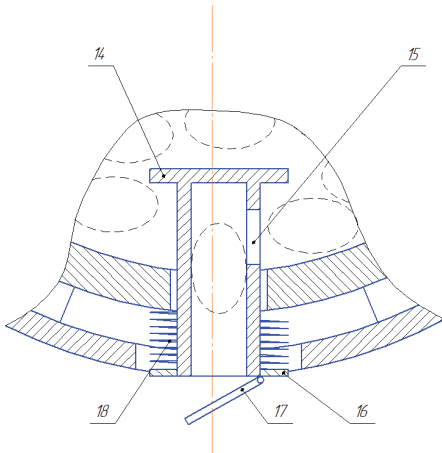


Рисунок 5 – Посевной палец.

ненной крышкой 17. Посевной палец 7 установлен в отверстии 6 барабана 5 по средствам пружин 18.

Шагающая сеялка работает следующим образом.

Перед началом работы на раму трактора 1 рис. 3) по средствам стоек 2 устанавливаю необходимое число высевующих устройств 4 равное количеству строк посева и загружают семена в семенные камеры высевующего устройства 4 через отверстия 12 рис. 4). После загрузки семян отверстия 12 закрывают крышками 13. Далее шагающая сеялка устанавливается на посевную полосу и трактор начинает движение.

В процессе движения трактора высевующее устройство 4 испытывая сопротивление грунта начинает по средствам контакта почвы с грунтозацепами 11 проворачивается на полуосях 3. В процессе вращения высевующего устройства 4 посевной палец 7 встречаясь с поверхностью почвы

начинает входить внутрь семенной камеры барабана 5, при этом подпружиненная крышка 17 рис. 5) начинает закрывать выходное отверстие посевного пальца 7. При вертикальном положении посевного пальца 7 происходит полное закрытие крышки 17 и вхождение посевного пальца 7 в семенную камеру, что приводит к открытию отверстия 15 на боковой стороне посевного пальца 7. Под действием силы тяжести семя через отверстие 15 попадает во внутреннюю полость посевного пальца 7. При дальнейшем вращении высевующего устройства 4 посевной палец 7 под действием пружины 18 выходит из семенной камеры барабана 5 перекрывая отверстие 15 на боковой поверхности, а подпружиненная крышка 17 под действием пружины начинает открывать выходное отверстие через которое происходит выпадение находящегося во внутренней полости посевного пальца 7 семени в посевную строку. Наличие упора 14 препятствует полному выходу посевного пальца 7 из барабана 5.

Предложенная конструкция шагающей сеялки позволит производить посев семян равномерно через одинаковое расстояние независимо от радиуса поворота посевной ленты кругового лесного питомника.

Литература

1. Бырдин П.В., Невзоров В.Н. Обеспечение рационального природопользования на основе круговых лесных питомников // Глобализация и эколого-экономическое развитие регионов: мате-

- риалы науч.-практ. конф. Москва, 2015. С. 112-116.
2. Чухматый В.Е., Чухматый В.В. Универсальная роторная пропашная широкозахватная сеялка: автор. свид. 1563612 СССР. № заявки 4244794/30-15; заявл. 18.05.1987; опубл. 15.05.1990; Бюл. № 18.
3. Шадрин А.П. Шагающая сеялка: пат. 2119278 Рос. Федерация. № заявки 97117869/13; заявл. 29.10.1997; опубл. 27.09.1998; Бюл. 15.

APPROXIMATING OF SIGNALS. THE FINITE DIFFERENCE METHOD

Saichyshyna N

Student of Kharkiv National University of Radio Electronics

The article considers the finite difference method, which can be derived by replacing the derivatives with difference schemes. The main problem of the method is the construction of the correct difference scheme, which will converge to the solution. The construction of the circuit is based on the properties of the original differential operator.

Keywords: approximation, difference scheme, differential equation, finite difference method, Laplace formula, method, signals.

The finite difference method is a numerical method for solving differential equations based on the replacement of derivatives by difference schemes. It is a grid method.

It is necessary to introduce the concept of difference schemes. Any equation of the form (S) for finding the grid function x will be called a difference scheme, and its (equation) solution, which we will always denote $\varphi\tau$, will be called a difference or grid solution (equation (S)). The operator $F\tau$ will be called the difference operator. Of course, in this general form, the definition of a difference scheme is in no way connected with the original Cauchy problem (E) – (C) (equation (O)). At the same time, if the operator $F\tau$ in (S) is defined by formula (2), then the difference scheme (S) has the most direct relation to problem (E) – (C) (to explain what these words mean is the purpose of the rest of paragraph).

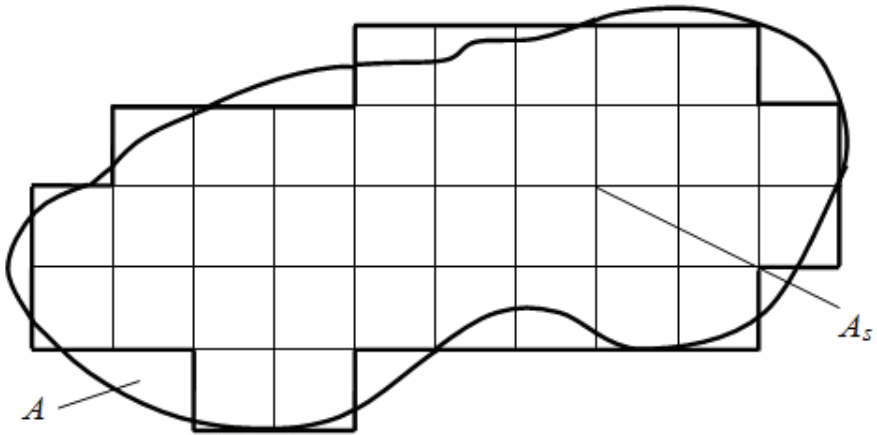
In the latter case, the difference scheme (S) is called the explicit (difference) Euler scheme. It is called explicit because its solution can be written out explicitly using recurrence relations:

$$(\varphi\tau)_0 = x_0,$$

$$(\varphi\tau)_i = (\varphi\tau)_{i-1} + \tau f[t_{i-1}, (\varphi\tau)_{i-1}], \quad i = 1, \dots, n.$$

Recall that recurrence relations are the same type of formulas that bind successive elements of a sequence (it can be a sequence of numbers, functions, etc.).

The idea of the finite difference method (the grid method) has been known for a long time, from the corresponding works of Euler. However, the practical application of this method was then very limited due to the huge amount of manual calculations related to the dimension of the resulting systems of algebraic equations, which took years to solve. Currently, with the advent of high-speed computers, the situation has fundamentally changed. This method has become convenient for practical use and is one of the most effective in solving various problems of mathematical physics.



Picture 1 – Grid area

The main idea of the finite difference method (grid method) for an approximate numerical solution of the boundary value problem for a two-dimensional partial differential equation is that:

1) on the plane in region A, in which the solution is sought, a grid region A_s is constructed (Picture 1), consisting of identical cells of size s (s is the grid step) and is an approximation of this region A;

2) the given partial differential equation is replaced at the nodes of the grid A_s by the corresponding finite-difference equation;

3) taking into account the boundary conditions, the values of the desired solution are established at the boundary nodes of the region A_s .

Solving the resulting system of finite-difference algebraic equations, we obtain the values of the desired function at the nodes of the grid A_s , i.e. approximate numerical solution of a boundary value problem. The choice of the grid domain A_s depends on the specific problem, but you should always strive to ensure that the contour of the grid region A_s best approximates the contour of region A.

Consider the Laplace equation

$$\frac{\partial^2 p}{\partial x^2} + \frac{\partial^2 p}{\partial y^2} = 0, \quad (1)$$

where $p(x, y)$ is the desired function, x, y are the rectangular coordinates of the flat region and we obtain the corresponding finite-difference equation.

We replace partial derivatives $\frac{\partial^2 p}{\partial x^2}$ and $\frac{\partial^2 p}{\partial y^2}$ in equation (1) by finite-difference relations:

$$\frac{\partial^2 p}{\partial x^2} \approx \frac{p(x+s, y) - 2p(x, y) + p(x-s, y)}{s^2},$$

$$\frac{\partial^2 p}{\partial y^2} \approx \frac{p(x, y+s) - 2p(x, y) + p(x, y-s)}{s^2}.$$

Then solving equation (1) with respect to $p(x, y)$, we obtain:

$$p(x, y) = [p(x+s, y) + p(x-s, y) + p(x, y+s) + p(x, y-s)] / 4. \quad (2)$$

Having set the values of the function $p(x, y)$ at the boundary nodes of the contour of the grid region A in accordance with the boundary conditions and solving the resulting system of equations (2) for each node of the grid, we obtain a numerical solution of the boundary value problem (1) in a given region A .

References

1. Christian Grossmann; Hans-G. Roos; Martin Stynes (2007). Numerical Treatment of Partial Differential Equations. Springer Science & Business Media. p. 23.
2. Majumdar P (2005). Computational methods for heat and mass transfer (1st ed.). Taylor and Francis, New York.
3. Hoffman JD; Frankel S (2001). Numerical methods for engineers and scientists. CRC Press, Boca Raton.

РОЛЬ ПРОСТОРУ У ВІРТУАЛЬНІЙ РЕАЛЬНОСТІ

Храменок Л. О.

викладач кафедри соціально-гуманітарної освіти

КЗ Сумського обласного інституту післядипломної педагогічної освіти

Ключові слова: віртуальна реальність, кіберпростір, медіапростір, фізичний світ.

На даний момент ми спостерігаємо інтенсивні і глибокі зміни в середині віртуальної реальності, які пов'язані з появою комп'ютерів і процесом математизації знань, який постійно прискорюється в час науково-технічного прогресу.

Важливе значення в дослідженні комп'ютерних віртуальних технологій відіграють поняття «кіберпростору», «медіапростору» та «фізичного світу». Осмислення місця та ролі комп'ютерної віртуальної реальності в розвитку новітніх технологій, зокрема мережі Інтернет та її впливу на процеси формування людських комунікацій розглянуто в роботах О. Астаф'євої, І. Девтерова, О. Дзьобаня, Д. Іванова, М. Кастельса, В. Саяпіна, А. Севальникова, Д. Шапіро та інших.

Мета статті – дослідити окремі аспекти кіберпростору, медіапростору та фізичного світу та їх значення у віртуальній реальності.

За допомогою технічних можливостей, суб'єкт може потрапити у нову дійсність, якій характерні свої відносини, критерії оцінки, своя система вимірювань, свій спосіб передачі інформації і т. д. Але незважаючи на технічні можливості кожна людина в певний момент часу буде сприймати

свій простір, свій можливий світ по різному.

Віртуальний простір, на думку В. Межуєва, «...ця нова реальність, подібно до інших уможливлених категорій, не піддається простому сприйняттю. Філософія культури ще тільки починає розкривати «віртуальне середовище» як феномен сучасної культури, а можливість існування цієї реальності в багатоманітних технологічних проявах зростає з кожним днем...» [9, с. 24].

Якщо поглянути хронологічно то поняття «медіапростору» з'явилося дещо пізніше ніж комунікаційний та інформаційний простори. Але все ж таки встигло посісти важливе місце в комп'ютерній віртуальній реальності та нелінійній методології. В системі концептуальних підходів осмислення медіапростору, один з них розглядає медіапростір через призму соціального простору та їх взаємовідношення. Так, Н. Кирилова наводить таке визначення медіа, що «це не простий засіб для передачі інформації, це ціле середовище, у якому проводяться, естетизуються і транслюються культурні коди» [7, с. 22]

Термін «медіапростір» досить поширений у дослідженнях інформаційних процесів та комунікаційних техно-

логіях. Необхідно зазначити, що особливу роль у створенні медіапростору на комп'ютерній основі належить саме Інтернету. Взаємодія технічної та соціальної сфери і утворює сучасну мережу Інтернет. В. В. Буряк, так характеризує переваги технологічної глобалізованої комунікації: «мобільний зв'язок та Інтернет докорінним чином перетворили повсякденне життя та професійне середовище перебування. Перш за все, це стосується сфери комунікації. Культурний і соціальний час істотно ущільнився, й індивідууми стали виконувати набагато більше вправ, ніж до початку інтернетизованої мобільної революції» [5, с. 62].

Розглядаючи комп'ютерну віртуальну реальність необхідно відмітити, що вона з однієї сторони пов'язана з кіберпростором, медіапростором, а з іншої – фізичним світом. О. М. Астаф'єва звертає увагу на те, що «інтерпретувати і пояснювати процеси саморозвитку віртуальних просторів – медіа-просторів, інформаційно-комунікативного середовища Інтернету, комп'ютерної віртуальної реальності – в рамках певних дисциплінарних меж складно» [3, с. 422]. Саме Інтернет виступає однією з головних тенденцій розвитку медіапростору, та щоб зрозуміти його вплив на індивідуальну та суспільну свідомість необхідно одночасно бути і спостерігачем і піддослідним.

Наука має стати головним важелем по збереженню життя людини. У створенні глобальних катастроф зазвичай винна людина, дії якої призвели до

певних катаклізмів у природі. Але і спасти людство в змозі лише людина з своїм унікальним мозком. М. В. Ковальчук, О. С. Нарайкін, Е. Б. Яцішина наголошують, що «сьогодні людство знаходиться в точці біфуркації. Ми підійшли до усвідомлення того, що повинні стати частиною природи, жити за рахунок принципово нових ресурсів і технологій, створених за зразком живої природи. А це можливо тільки з використанням найдосконаліших технологічних досягнень» [8, с. 11].

Л. А. Ороховська орієнтується на вплив мас-медіа на конструювання соціальної реальності, визначаючи, що «мас-медіа, здійснюючи вплив на свідомість індивіда, можуть спричинити віртуалізацію реального життя, адже досить часто ми співвідносимо реальні події та вчинки з стандартами, які нам нав'язують мас-медіа. Мас-медіа здійснюють на нас не тільки значний вплив, але і значний тиск. Ті, хто володіє ЗМІ, визначають, що потрібно та корисно аудиторії, а що – ні» [10, с. 110].

С. Грицай визначаючи структуру медіапростору як складну суб'єктивно-об'єктивну систему, що знаходиться під впливом соціальної дійсності пише «медіапростір є динамічною системою, що розширюється в міру розвитку суспільства, – це ключова властивість медіапростору. Форми і засоби надання інформації історично змінювалися. Медіапростір – не просто соціальна система, а система, компоненти якої об'єднані в соціальну мережу. Загалом його архітектоніка складна, кожен еле-

мент у структурі може функціонувати в рамках відкритої системи, одержувати і передавати інформацію. У медіа-просторі ядром, що організовує систему, виступає масова інформація, яку потрібно переробити для споживання аудиторією. Тому всі об'єкти: соціальні інститути, організації, групи та окремі особи, зайняті її виробництвом і споживанням, мають органічно входити в цю систему» [6, с. 29].

Таким чином, процес комп'ютеризації неможливий без засобів комунікації та безпосередньо пов'язаний зі складними технологіями. Так, В. Г. Буданов, елементи комунікації вбачає в метафорі випадкового нелінійного середовища. Він зазначає, «випадкове нелінійне середовище – це мовне середовище, елементи середовища – об'єкти комунікації, самодія – когнітивний процес, взаємодія – комунікативний канал. Оскільки інформація може народжуватися, генеруватися, то це моделі не обов'язково фізичних сил або полів. У цій метафорі N – елементний сектор відповідає колективу, двох-елементний діалогу. Виключаючи в парній комунікації у одного з елементів самодію, ми приходимо як суб'єкт об'єктної дихотомії» [4, с. 81–82]. За допомогою комп'ютерних віртуальних технологій людина може вільно входити в кіберпростір, та виходити з нього змінюючи його на фізичний простір та навпаки, адже вона функціонує за власними законами. Саме комп'ютерні віртуальні технології є границею між кіберпростором і фізичним світом. Взаємодія кіберпростору з фізичним

світом і утворюють комп'ютерну віртуальну реальність, яка і є тією межею, що їх об'єднує. Як відмічають Н. З. Алієва, С. А. Гармаш, А. С. Якубенко, А. А. Прігородова «результатами процесу віртуалізації, зобов'язані комп'ютерній техніці, є кіберпростір і медіа-простір. Однак уявну, віртуальну реальність створює також мистецтво в процесі творчості. Відмінність дійсності полягає в тому, що в мистецтві вона носить символічний характер і відображає ідейний світ, а дійсність, створена технікою, позбавлена символіки, в ній світ відображений безпосередньо» [1, с. 195].

Створення комп'ютерних віртуальних технологій значно розширює спектр можливостей людини. Вплив її на комп'ютерні віртуальні технології дозволяє вирішувати надскладні задачі, які не може вирішити просто людський мозок, або запрограмований нею комп'ютер. Як пише М. Хайм «кіберпростір може містити багато світів, що чергуються, але альтернативність іншого світу зосереджена в його здатності збуджувати в нас інші думки і відчуття» [11, с. 77].

Так, комп'ютерні мережеві технології відкривають нам нові можливості, але тим самим і спонукають до переосмислення ролі віртуальності, де В. І. Аршинов, М. В. Лебедев влучно відмічають, що «віртуальність (кіберпростір) як у вузькому, так і в широкому сенсі – це специфічна форма інтерактивної комунікативної реальності, що являє собою по суті альтернативу матеріальному світу. Оскільки кіберп-

ростір передбачає можливість безпосередніх контактів з іншими віртуальними суб'єктами, проблема комунікації підлягає переосмисленню в цьому контексті» [2, с. 40].

Дослідження комп'ютерних віртуальних технологій може допомогти в реалізації варіантів подій які ще не відбулися. На нашу думку, технологія віртуальної реальності є однією з можливостей відшукування раціональних рішень.

Література

1. Алиева Н. З., Гармаш С. А, Якубенко А. С., Пригородова А. А. Концепции виртуальной реальности в мировой философской мысли. Успехи современного естествознания. 2012. №6. С. 195–196.
2. Аршинов В.И., Лебедев М.В. Европейская субъективность и виртуальный субъект постнеклассической науки. Вопросы искусственного интеллекта. 2008. № 1. С. 27-41.
3. Астафьева О. Н. Синергетический дискурс современных информационно-коммуникативных процессов. Синергетическая парадигма. Когнитивно-коммуникативные стратегии современного научного познания. Москва: Прогресс-традиция, 2004. С. 419–443.
4. Буданов В. Г. Методология синергетики в постнеклассической науке и в образовании. Изд. 2-е, испр. Москва: Изд. ЛКИ, 2008. 232с.
5. Буряк В. В. Глобальна мобілізація та соціальні медіа. Наукові записки Київського університету туризму, економіки і права. Серія: філософські науки. / Гол.ред. В. С. Пазенок. Київ: КУТЕП, 2011. Випуск 11. С. 56–65.
6. Грицай С. Архітектоніка сучасного медіапростору. Вісник Книжкової палати, 2012. № 5. С. 26-29.
7. Кириллова Н. Б. Медиакультура: от модерна к постмодерну. 2-е изд. Москва: Академический проект, 2006. 362 с.
8. Ковальчук М. В., Нарайкин О. С., Яцишина Е. Б. Конвергенция наук и технологий – новый этап научно-технического развития. Вопросы философии. 2013. № 3. С. 3-11.
9. Межуев В. М. Модернизация и глобализация – два проекта «эпохи модерна». Глобализация и перспективы современной цивилизации / [отв. Редактор К. Х. Делокаров]. Москва: КМК, 2005. С. 3-26.
10. Ороховська Л. А. Віртуалізація мас-медіа і соціальної реальності в контексті конструктивістського підходу. Наукові записки Київського університету туризму, економіки і права. Серія: філософські науки. / Гол.ред. В. С. Пазенок. Київ: КУТЕП, 2012. Випуск 13. С. 107–117.
11. Heim M. The Metaphysics of Virtual Reality. New York, Oxford: Oxford University Press, 1993. 175p.

СИНТЕЗ И АНАЛИЗ КООРДИНАЦИОННЫХ СОЕДИНЕНИЙ ИОДИДОВ НИКЕЛЯ (II), ЦИНКА И КАДМИЯ С ИЗОНИКОТИНОИЛГИДРАЗОНОМ ПАРА-АЦЕТИЛАМИНОБЕНЗАЛЬДЕГИДОМ (ИНГПААБА-L4)

Чантурия М.М.

Ассоциированный профессор, кандидат химических наук
Сухумский Государственный Университет

Цинцадзе М.Г.

Профессор, доктор химических наук
Грузинский Технический Университет

Гулбани Д.В.

кандидат химических наук
Сухумский Государственный Университет

Аннотация

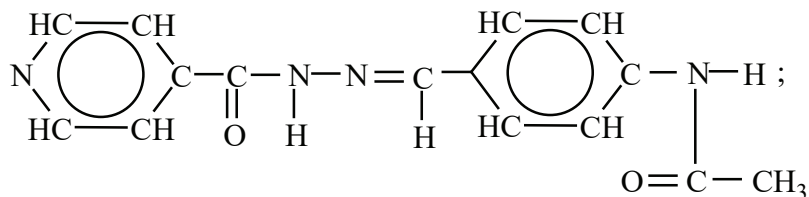
Разработаны методы синтеза координационных йодидных соединений никеля(II), цинка и кадмия с изоникотиноилгидразоном пара-ацетиламинобензальдегидом (ИНГПААБА-L⁴). Синтезированные комплексы йодидных соединений: NiI₂L⁴(H₂O), ZnI₂L₃⁴, CdI₂L₂⁴(H₂O). Определены их состав, строение, охарактеризованы их свойства, установлен способ координирования. Указанные комплексы являются удобными моделями для интерпретации структурно-химических эффектов в области координационной химии.

Ключевые слова: координационные соединения, изоникотиноилгидразон пара-ацетиламинобензальдегид, синтез, методы.

Введение: В качестве лигандов применялись азометины (гидразоны изоникотингидразида).

Изоникотиноилгидразон пара-ацетиламинобензальдегида (ИНГПААБА) был получен при сливании этанольных растворов, содержащих 0,01 моля изоникотингидразида и 0,01 моля пара-ацетиламинобензальдегида (рН≈6). После смешивания этих растворов (рН 6) и смеси, для удаления воды и поддержания кислой среды, добавляли несколько капель концентрированной серной кислоты. Через несколько минут выпадал осадок, который отфильтровывали, промывали на фильтре этанолом, сушили в вакуум-эксикаторе над CaCl₂.

ИНГПААБА – изоникотиноилгидразон пара-ацетиламинобензальдегида (ИНГА-17) (L⁴):



ИНГПААБА порошок желтого цвета, нерастворимый в этаноле, хорошо растворимый в ДМФА.

Найдено, %: С – 56,91; N – 18,69; H – 4,69.

Для ИНГПААБА – $C_{15}H_{14}N_4O_2$;

Вычислено, %: С – 60,40; N – 18,79; H – 4,22;

Синтез комплексных йодидных соединений никеля(II), цинка и кадмия с изоникотиноилгидразоном пара-ацетиламинобензальдегидом (ИНГПААБА- L^4) – с лигандом осуществляется путем сливания при комнатной температуре этанольных растворов очищенных солей металлов с диметилформамидными растворами гидразонов в молярных соотношениях м:гидразон = 1:1; 1:2 и 1:3. При этом растворы соответствующих солей (0,0025 моля в 10-20 мл этанола) перемешивались с растворами гидразонов (0,0025 моля в 15 мл, 0,005 моля в 20 мл и 0,0075 моля в 25 мл ДМФА). Осадки в некоторых случаях выделялись сразу же, а некоторые – через неделю и даже через месяц при сливании раствором исходных компонентов после кристаллизации реакционных смесей в вакууме-эксикаторе над $CaCl_2$. В некоторых случаях фильтраты после отделения компонентов выдерживались до кристаллизации на более длительное время, в результате чего выделялись комплексы.

Цель работы. Задачей нашего исследования являлась разработка методов получения координационных йодидных соединений никеля(II), цинка и кадмия с изоникотиноилгидразоном пара-ацетиламинобензальдегидом (ИНГПААБА- L^4), изучение их состава, строения современными инструментальными методами.

Практическая ценность. Результаты исследования йодидных соединений никеля(II), цинка и кадмия с изоникотиноилгидразоном пара-ацетиламинобензальдегидом (ИНГПААБА- L^4), могут быть использованы в лабораторной практике для прогнозирования синтеза координационных соединений с родственными по своей природе органическими лигандами.

Объекты и методы исследований.

В качестве исходного вещества использовались йодиды никеля(II), цинка и кадмия (квалификации «чистый для анализа», «чистый», которые предварительно очищались перед использованием по специальной методике, перекристаллизацией). В качестве органического лиганда использовали изоникотиноилгидразо-

ном пара-ацетиламинобензальдегидом (ИНГПААБА- L^4). Комплексные соединения никеля(II), цинка и кадмия с изоникотиноилгидразоном пара-ацетиламинобензальдегида получались в результате взаимодействия йодидов никеля(II), цинка и кадмия и лиганда в водной среде при комнатной температуре и при молярном соотношении исходных компонентов 1:2. Часть комплексов выпадала сразу же при сливании исходных растворов, другая же часть выделялась после непродолжительной кристаллизации в вакуум-эксикаторе. Выпавшие осадки комплексов отделялись от маточника фильтрованием, промывались на фильтре небольшими порциями воды и сушились в вакуум-эксикаторе над CaCl_2 .

Для проверки растворимости использовали воду, этанол, ацетон, диоксан, ДМСО (диметилсульфоксид), ДМФА (диметилформамид). Этанол и ацетон перед употреблением очищали. ДМФА и ДМСО марки «ч» высушивали над прокаленным поташем в течение нескольких суток и перегоняли под вакуумом, собирая фракции соответственно при 150-151°C и 140-141°C. Состав полученных комплексов устанавливали элементным анализом. Физико-химические свойства и строение полученных комплексов изучены методами рентгенографии, ИК спектров поглощения, термографии, проведены квантово-химические расчеты лиганда.

Методы исследования комплексных соединений

Элементный анализ

Для установления элементного анализа синтезированных соединений они анализировались на содержание металлов трилонометрическим методом, а галогенид-ионы – аргентометрическим методом, на углерод и водород – по модифицированным методом микроанализа, а на азот – по методу Дюма.

Инфракрасные спектры поглощения

ИК-спектры поглощения комплексных соединений и исходных лигандов записывались на спектрофотометрах UR-20 в области 400-4000 см^{-1} . Применялась методика растирания поликристаллических образцов с вазелиновым и фторированными маслами. Для некоторых соединений записывались спектры этанольных растворов.

Определение молярной электропроводности

Молярная электропроводность определялась с помощью кондуктометра КЭЛ-1М при 25°C. Удельная электропроводность вычислялась по формуле:

$$\chi = \frac{1}{R_x} \alpha,$$

где χ – удельная электропроводность ($\text{ом}^{-1} \text{см}^2$),

α – постоянная измерительная ячейка (для наших условий

$\alpha = 0,037451 \text{ ом}^{-1} \text{ см}^2$),

R_x – удельное сопротивление (ом).

Молярную электропроводность определяли по формуле:

$$\mu = 1000 \text{ ом}^{-1} \text{ см}^2 \text{ мол}^{-1}$$

C – концентрация раствора (г-моль/л).

Разработаны методы синтеза координационных соединений йодидов никеля(II), цинка и кадмия с изоникотиноилгидразона пара-ацетиламинобензальдегидом.

Координационные соединения йодидов никеля(II), цинка и кадмия с изоникотиноилгидразоном пара-ацетиламинобензальдегида

$\text{NiI}_2\text{L}^4(\text{H}_2\text{O})$. Навеска с 0,7813 г (0,0025 моль) NiI_2 растворялась в 20 мл этанола, после чего к приготовленному раствору приливался раствор, с 0,705 г (0,0025 моль) L^4 в 15 мл ДМФА раствор желтого цвета. Сразу получен осадок зеленого цвета, который растворяется в воде, ДМСО (при нагревании), мало растворяется в ДМФА, не растворяется в воде, этаноле, ацетоне, толуоле, бензоле, эфире. При 178°C начинается изменение окраски, при 184°C обугливается.

ZnI_2 . Для синтеза этого соединения смешивалось 20 мл этанольного раствора, содержащего 0,7979 г (0,0025 моль) ZnI_2 рН≈5,5) и 25 мл ДМФА раствора, содержащего 0,705 г (0,0025 моль) L^4 рН≈6). После смешивания растворов, смесь давали отстоять и выпавшие кристаллы отделяли от маточника фильтрованием. ZnI_2 – кристаллы желтого цвета, мало растворяются в ДМСО, не растворяется в воде, ацетоне, эфире, этаноле, ДМФА, бензоле, толуоле. $T_{\text{пл.}} = 310\text{-}312^\circ\text{C}$.

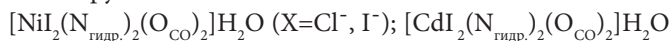
$\text{CdI}_2(\text{H}_2\text{O})$. К 20 мл этанольного раствора, содержащего 0,9155 г (0,0025 моль) CdI_2 растворяли при нагревании, рН≈5,5), приливался раствор с 1,4115 г (0,005 моль) L^4 в 25 мл ДМФА (рН≈6). Выпадали кристаллы желтого цвета через три дня после смешивания растворов (рН≈5,5), которые при нагревании растворялись в ДМФА, ДМСО, не растворялись в воде, этаноле, ацетоне, хлороформе, толуоле, бензоле. (Полученное координационное соединение имеет рН≈5,5),

Полученные соединения устойчивы на воздухе, не выветривались даже при длительном хранении.

Результаты элементного анализа координационных йодидных соединений никеля (II), цинка и кадмия с ИНГПААБА (L^4)

Соединение	Найдено, %					Вычислено, %					$T_{\text{пл.}}$ °C (±2)
	М	Н	С	Н		М	Н	С	Н		
$\text{NiI}_2\text{L}^4(\text{H}_2\text{O})$	10,38	9,42	30,29	42,67	2,37	9,92	9,41	30,25	42,69	2,35	184
ZnI_2	12,26	11,97	39,11	27,21		12,04	12,04	38,70	27,31	3,01	312
$\text{CdI}_2\text{L}^4(\text{H}_2\text{O})$	5,90	14,12	46,01	21,76		5,58	14,42	46,35	21,80	3,60	–

Спектральные данные показали, что по характеру координации с металлом соединения молекулы L^4 бидентатны, образуют пятичленные металлоциклы и координируются с металлами, атомами азота гидразидных и кислорода карбонильных групп:



Литература:

1. Ed. Mak Oliff K. Per., Methods and achievements of bioorganic chemistry, With the English. Moscow: Mir. 1978. 411
2. Henkin R.I., Craziadei P.P. G., Bradley D.F. The Molecular Basis of Taste and Its Disorders. Ann Internal Med. 1969. V. 71. P. 791-821.
3. Цинцадзе Майя Гивиевна. Координационные соединения с азот- и кислородосодержащими лигандами – производными алифатического, ароматического и гетероциклического ряда. Национальная академия Наук Грузии, Тбилиси, 2009, 248.
4. Инфракрасные спектры неорганических и координационных соединений. Накамото К. М.: Мир. 1966. 411 С.
5. Chemistry of hydrazones, Ed. Kitaeva Yu.P. Moscow: Nauka, 1977, 204 p.
6. Цинцадзе Г.В., Цинцадзе М.Г., М.М. Чантурия, Нариманидзе А.П. Исследования в области координационных соединений металлов с гидразонами изоникотингидразида, Proceedings of Tskhum-Abkhazian Academy of Sciences. Dedicated to the 70th Anniversary of Georgian National Academy of Sciences, Publishing House “Meridiani” Tbilisi, Tom I, 2011, 351-364. ©ISBN 978-9941-10-421-3.

Resume

SYNTHESIS AND ANALYSIS OF COORDINATION COMPOUNDS OF NICKEL(II), ZINC AND CADMIUM IODIDES WITH ISONICOTINOYL HYDRAZONE OF PARA-ACETYLAMINOBENZALDEHYDE (INGPAABA- L4)

Mineda Chanturia, Maia Tsintsadze, Daredjan Gulbani
Sokhumi State University, Georgia Technical University

Methods have been developed for the synthesis of coordination iodide compounds of nickel (II), zinc and cadmium with isonicotinoyl hydrarazone para-acetylaminobenzaldehyde (INGPAABA- L⁴). Synthesized complexes of iodide compounds $NiI_2L^4(H_2O)$, ZnI_2 , $CdI_2(H_2O)$. Their composition and structure are determined, their properties are characterized, and a method of coordination is established. These complexes are convenient models for the interpretation of structural and chemical effects in the field of coordination chemistry.

A DESIGN OF A NEURAL NETWORK FOR IMAGES CLASSIFICATION USING MATLAB AND JAVA

Dudka. D. A.

Student of the Department of Computer Engineering and Management
Kharkov National University of Radio Electronics

I. INTRODUCTION AND PROBLEM STATEMENT

In the past few years, face recognition and classification has received a significant attention and regarded as one of the most successful applications in the field of image analysis [1]. Developing a computational model for face recognition and classification is difficult [1, 4]. Artificial neural networks (ANT) were used largely in the recent years in the fields of image processing and pattern recognition. In the recent years, different architectures and models of ANT were used for face detection, recognition and classification. In paper [2] was proposed approaches for face recognition based on combined Gabor wavelet faces with ANT feature classifier [1, 7]. The Gabor wavelets used to represent face image. It was proved the representation of face images using Gabor wavelets was effective for facial action recognition and face identification. In reference [3] was proposed approach based on ANT and Gabor wavelets to detect desirable number of faces in fixed photo with gray background [2, 552]. They used correlation of a window with a face with photo. Then they estimated areas of candidate of face presence. After that, they used step algorithm and referred these areas and

around them to section of extraction of Gabor wavelets characteristics and neural network classifier. [2, 554]. In work [4] was presented a back propagation neural network (BPNN) for face recognition. The BPNN input is feature vector based on Fourier Gabor filters. They used an algorithm for detecting face regions in images using the color of skin which presents overlooked in different background, accessory and clothing [3, 68]. After that, they introduced Gabor filters with 8 orientations and 5 resolutions to get maximum information by varying the resolution and orientation. This is done to generate and extract the features vector of the whole face in image. BPNN is then applied to perform the recognition task [3, 70]. This solution was implemented using Java environment [3, 74]. In reference [5] was considered an efficient face recognition method where enhanced local Gabor binary pattern histogram sequence has been used for efficient face feature extraction and generalized neural network with wavelet as activation function is being used for classification. In papers [6, 7] was compared the effectiveness of Daubechies and Haar wavelet transforms while images classifying with the use of ANT.

Although, many researches have worked

on the problems of face recognition and classification for many years, still several challenges need to be solved. This work is attempt to downsize the computational complexity of an artificial neural network due to the original algorithm with the application of Daubechies wavelet transform [4, 15].

In many modern web services, for example [work.com](#), a visual identification possibility of users is very desirable. New user must upload his real portrait as his avatar during registration on that site. This portrait needs to be a full face focused image which made often with a simple gray or monochromatic background. In such case the portraits with a head rotation are not acceptable. Clearly, images verifying functionality must be realized programmatically [4, 17].

The purpose of this paper is ANT design for portrait images classification in two categories:

- 1). Full face portraits;
- 2). Portraits with a head rotation.

The practical outcome of the work is a web-application capable to verify uploading images with the use of ANT [4, 19].

In this work a capability of Matlab tools for preliminary images handling, ANT creating, training and simulating is demonstrated. What's more, Matlab provided Java-code generating from Matlab-function. It's significantly simplifies the programmatic realization of neural networks as parts of Java web-applications. [4, 21].

II. PROBLEM SOLUTION AND RESULTS

For ANT training were used 128 portrait RGB-images that made mainly with a light

or dark monochromatic background and also with a gradient background. There were people with various ages, ethnicities, skin and hair colors. Half of the images were male portraits and half of the images were female portraits. Half of the images were full face portraits and half of the images were portraits with a head rotation.

For ANT simulating and classification performance assessing were used other 32 portrait RGB-images that were choosed using similar prerequisites, in such proportions as mentioned before [5, 1557].

First of all portrait images were framed to size 256×256 px and uploaded to Matlab. With the use of Matlab-functions `rgb2gray` and `im2double` color images were converted to black and white images with range their pixels brightness from 0 to 1. That means we should use the input of ANT that consists from 65536 values to each of images. That means far too much complexity while designing and using the neural network [5, 1560].

In the paper the solution is proposed for resolving this problem. We could reduce a computational complexity due to application of Daubechies wavelet transform with 3 steps to each of images.

The 1-st step scheme for a discrete two-dimensional (2-D) Daubechies wavelet decomposing is shown in Fig. 1.

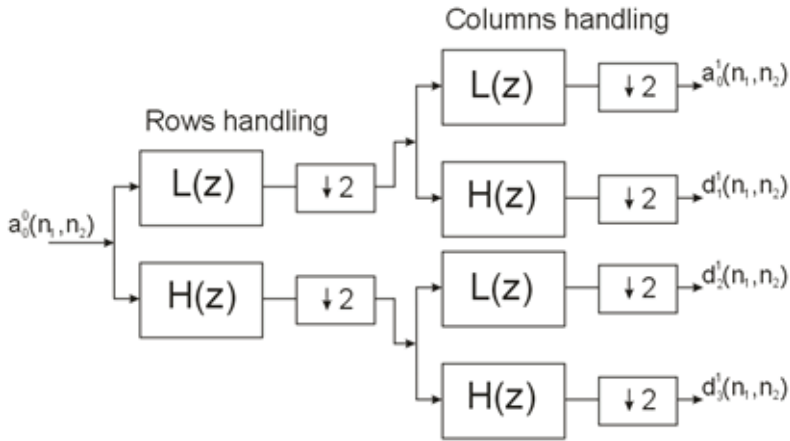


Figure 1. Image decomposition using 2-D wavelet transform

For matrices of images decomposing were used lowpass $L(z)$ and highpass $H(z)$ first-order Daubechies filters. Matlab-function `dwt2` computes the approximation coefficients matrix $a_0^1(n_1, n_2)$ and details coefficients matrices $d_1^1(n_1, n_2)$, $d_2^1(n_1, n_2)$, $d_3^1(n_1, n_2)$ (horizontal, vertical, and diagonal, respectively), obtained by wavelet decomposition of the input matrix $a_0^0(n_1, n_2)$. The result of the 1-st step of Daubechies wavelet transform for one of images is shown in Fig. 2.

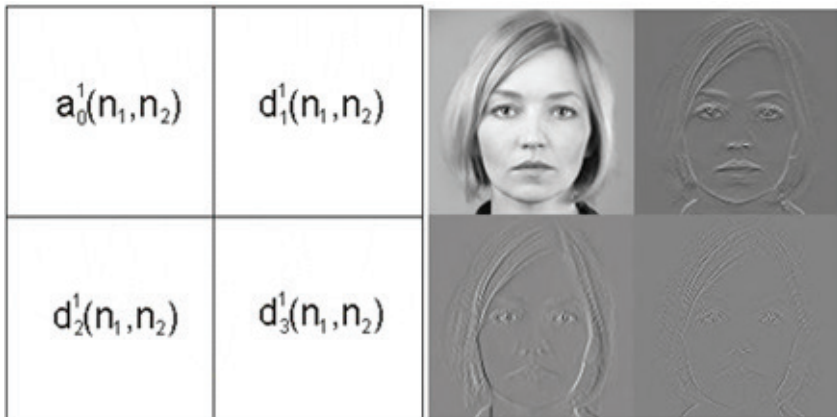


Figure 2. The result of the 1-st step of 2-D wavelet transform On the 2-nd step of 2-D wavelet decomposition each of these matrices were transforms separately, replacing the input matrix in Fig. 1. And from four result matrices was remained only a corresponding

one. The 3-rd step of 2-D wavelet decomposition was executed in the same way. This has given to reduce the image matrix to size 64×64 px and downside the input of ANT to 4096 values to each of images [6, 105].

The design a feed-forward back propagation neural network with 3 layers was performed using Matlab-function newff [8]. Matlab-function trainrp that updates weight and bias values according to the resilient backpropagation algorithm (RPROP) was used for training. After training with the use of 128 portrait images the simulation using 32 control images was performed. Simulation results for various number of neurons in 1-st layer of ANT is shown in Fig. 3.

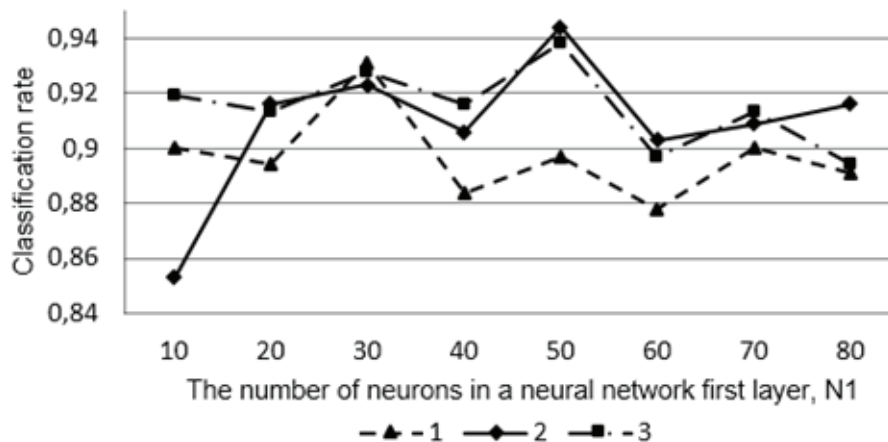


Figure 3. Classification rate curves

Here the number of neurons in 2-nd layer of ANT was equal $N_2=50$ and the number of neurons in 3-rd layer ANT was equal $N_3=2$. Curves 1-3 in Fig. 3 were acquired for sum square errors (SSE) values 1, 0.1 and 0.01 (during training procedure) respectively [6, 104].

The resulting fully trained neural network with parameters $N_1=N_2=50$ and $N_3=2$ was used for converting (using Matlab-function getFunction) into a Matlab-function and, further, for Java-code generating with the application of Matlab Library Compiler.

III. CONCLUSIONS

This paper presents an efficient face classification method, using artificial neural network. The original algorithm for reducing the computational complexity of ANT is proposed. Experimental results of the proposed method on the open portrait images set have evidently illustrated its effectiveness and robustness. It's significantly simpli-

fies the programmatic realization of Java web-applications with neural network functionality. The optimal configuration of the artificial neural network, that found by simulating, provides the classification rate 0.94 [7, 61].

The web-application based on technologies Java 8, Maven, Spring WebMVC, Apache Commons File Upload Library, JSP, CSS3, Tomcat, including Java-code for the designed neural network, was developed. The source code this web-application and full set of images is available at a link github.com/bozonhiggsa/images-Recognizing. Note that for a stand-alone application that execute on PC without Matlab R2016a you should install Matlab Compiler Runtime 9.0.1.

Future efforts will be focused on images recognition and classification using Python and deep learning algorithms [8, 46].

[1] Al-Allaf, Omaima N.A. Review of face detection systems based artificial neural networks algorithms, *The International Journal of Multimedia & Its Applications (IJMA)*, Vol. 6, No. 1, February 2014, pp. 1–16.

[2] Sahoolizadeh, Sarikhanimoghdam and Dehghani “Face Detection using Gabor Wavelets and Neural Networks”, *World Academy of Science, Engineering and Technology*, Vol. 45, 2008, pp. 552–554.

[3] Mohammad Abadi, et al, “Face Detection with the Help of Gabor Wavelets Characteristics and Neural Network Classifier”, *American Journal of Scientific Research*, Issue.36, 2011, pp. 67–76.

[4] Anissa Bouzalmat, et al “Face Detection And Recognition Using Back Propagation Neural Network And Fourier Gabor Filters”, *Signal & Image Processing: An International Journal (SIPIJ)*, Vol. 2, No. 3, September 2011, pp. 15–21.

[5] Sharma P, Arya K.V., Yadav R.N. Efficient face recognition using wavelet-based generalized neural network. *Signal Processing*, Vol. 93, Issue 6, June 2013, pp. 1557–1565.

[6] Буй Тхи Тху Чанг, Фан Нгок Хоанг, Спицын В.Г. Алгоритмическое и программное обеспечение для классификации цифровых изображений с помощью вейвлет-преобразования Хаара и нейронных сетей // *Известия Томского политехнического университета. Инжиниринг георесурсов – Томск: НИТПУ*, 2011. – Т.319. – № 5. – С. 103–106.

[7] Фан Нгок Хоанг, Спицын В.Г. Алгоритмы для классификации отпечатков пальцев на основе применения фильтра Габора, вейвлет-преобразования и многослойной нейронной сети // *Известия Томского политехнического университета. Инжиниринг георесурсов – Томск: НИТПУ*, 2012. – Т. 320. – № 5. – С. 60–64.

[8] Hudson, Hagan and Demuth, *Neural Network Toolbox™ User’s Guide R2012a*, The MathWorks, Inc., 3 Apple Hill Drive Natick, MA 01760–2098, 2012, www.mathworks.com.